

Development of a Draping Algorithm for Non-Structural Aerospace Composites

by

Jacob Hoffer

Thesis submitted to the University of Ottawa

in partial fulfillment of the requirements

for the M.A.Sc. Degree in

Advanced Materials and Manufacturing Engineering

Department of Mechanical Engineering

Faculty of Engineering

University of Ottawa

© Jacob Hoffer, Ottawa, Canada, 2020

Abstract

Fibre reinforced polymer matrix composites are used frequently in aerospace applications. Manufacturers of aerospace components favour composites over traditional metallic alloys due to their light weight, high modulus, corrosion resistance and fatigue resistance. Advantages of composites for non-structural interior components over metallic include: ease of manufacturing for single parts of complex geometry as opposed to assemblies, cheaper manufacturing of a limited series of parts and composites greatly reduced noise, vibration and harshness. However, manufacturing interior composite components requires critical attention to detail during the preforming stages and handling of dry fabric textiles. Since these components are handmade they often yield lower profits and therefore efficient preforming is critical. Designing draping strategies for industrial liquid composite moulding processes requires a significant amount of time and testing, in simulation and also working on physical moulds. Mould and part surfaces are often defined by a number of geometric features, labelled base surfaces in the context of this thesis, which can be used to quickly probe multiple draping strategies and identify the best one. Traditionally, trial and error work is performed over a full mould surface until a working or acceptable draping strategy is found, rarely identifying the best strategy.

The work in this thesis presents the initial development stages for a draping predictive tool aimed at quickly probing multiple draping scenarios in simulation prior to receiving moulds and identifying the best draping strategy for industrial non-structural aerospace composites. A multi-parameter remodelling tool – the conical frustum – was developed for uniformly identifying base surfaces through 12 geometric parameters linked into a database of in-plane shear and yarn orientations results. The development of the database is discussed, detailing Taguchi methods of

experimental design used for developing linear functions from the database results, which allow interpolation of results on base surfaces that do not directly exist within the database.

This thesis also includes major developments for the core draping algorithm used for linking individual base surface results together when probing draping strategies. Further investigations were performed on unique elements of in-plane shear behaviour that are encountered during draping, so that these could ultimately be considered during the development of this version of the draping algorithm whilst others may be included in future developments.

Acknowledgements

I would like to express my sincere gratitude for my supervisor Dr. François Robitaille for his continuous support, wealth of knowledge, patience and guidance throughout my thesis. I am incredibly thankful for the opportunity to work with him and could not imagine a better adviser and mentor.

I would like to thank the entire team working on our project: my colleagues in the Colonel By office for the stimulating discussions and engaging cooperation throughout my time at the University of Ottawa; and the Hutchinson team members for providing access to their facilities and direct links to industrial practices.

I would like to extend thanks to my friends, especially those in computer engineering who came to my aid in times of struggle and frustration. I would still be lost in lines of code without you.

Lastly, I would like to thank my family for their unyielding support from words of encouragement and care packages, to irregular visits and regular breakfasts. Thank you for always believing in me.

“Persistence, persistence, persistence. Just don’t give up. The power can be created and maintained through daily practice, through continuous effort.” – Bruce Lee

Table of Contents

1	Introduction.....	1
1.1	Background on composite materials	1
1.2	Project overview and contributions	3
1.3	Section summary.....	7
2	Literature Review	10
2.1	Introduction.....	10
2.2	2D textile reinforcements.....	10
2.3	In-plane shear of woven dry fibre reinforcements	14
2.4	Draping algorithms	17
2.4.1	Kinematic draping algorithm	17
2.4.2	Energy drape algorithm.....	18
2.4.3	Nottingham DrapeIt	19
2.4.4	Nottingham kinematic draping algorithm.....	19
2.5	Manufacturing composites.....	21
2.5.1	Resin transfer moulding	22
2.5.2	Vacuum assisted resin transfer moulding	22
2.6	Statistical experimental design	23
2.6.1	Plackett-Burman screening designs	24
3	Evaluating Geometric Features	29
3.1	Base surfaces.....	29
3.2	Defining the conical frustum	31
3.2.1	Introduction.....	31
3.2.2	12 parameter conical frustum.....	32
3.2.3	Remodelling base surfaces using the conical frustum	35
3.2.3.1	Calculating centre points C1 and C2	37
3.2.4	Calculating the remaining conical frustum parameters	44
3.2.5	Calculating the conical frustum parameters using Method 1.....	46
3.2.6	Calculating the conical frustum parameters using Method 2.....	49
3.2.6.1	Sample calculations performed by Script 2 from example 2.....	52
4	Hierarchy of Parameters.....	54
4.1	Mould decomposition	54
4.2	Measurement tools	55
4.3	FR mould decomposition.....	57

4.4	<i>BD100</i> mould decomposition	62
4.5	Hierarchy of parameters: Five parameter conical frustum	63
5	Basic conical frustum draping simulations: Taguchi study	67
5.1	Draping parameters	67
5.2	Design of experiments	69
5.2.1	Input Factors	71
5.3	Draping simulations	74
5.3.1	Runs #1 and #2.....	76
5.3.2	Runs #3 and #4.....	77
5.3.3	Runs #5 and #6.....	78
5.3.4	Runs #7 and #8.....	78
5.3.5	Runs #9 and #10.....	79
5.3.6	Runs #11 and #12.....	80
5.3.7	Run #13 and #14	80
5.3.8	Runs #15 and #16.....	81
5.4	Discussion	82
5.5	Follow up study: Reflected runs	85
5.5.1	Runs #R1 and #R2	86
5.5.2	Runs #R3 and #R4	87
5.5.3	Runs #R5 and #R6	87
5.5.4	Runs #R7 and #R8	88
5.5.5	Runs #R9 and #R10	88
5.5.6	Runs #R11 and #R12	89
5.5.7	Runs #R13 and #R14	89
5.5.8	Runs #R15 and #R16	90
5.6	Discussion	91
6	Database population using the basic conical frustum	95
6.1	Introduction.....	95
6.2	Database population.....	96
6.3	Developing linear functions of results	97
7	Unique in-plane shear occurrences	105
7.1	Shear progression.....	105
7.1.1	Defining shear progression	105
7.1.2	Vector projection	107

7.1.2.1	Alternative head-to-tail summation	112
7.1.3	Head-to-Tail Summation Example	113
7.2	Effects of double curvature configurations	117
7.3	Angle between surfaces	119
7.3.1	Case 1: Corner.....	119
7.3.2	Case 2: Chamfers along a single edge at a corner	123
7.3.3	Case 3: Chamfers along multiple edges at a corner.....	124
7.3.4	Preliminary shear angle calculations for Case 1 and 3	126
8	Linking base surface results: The draping algorithm	129
8.1	Draping algorithm overview	129
8.2	Definition of a draping strategy	130
8.3	Organization of surface connections.....	133
8.4	Consolidation of boundary conditions	135
9	Draping algorithm demonstration.....	139
9.1	Partial draping demonstrations	140
9.1.1	Draping starting direction	140
9.1.2	Demonstration 1 and 2	141
9.2	Validating consolidation of boundary conditions	145
9.3	Draping algorithm sample.....	147
10	Conclusions and recommendations	152
11	References.....	154
Appendix A Remodelling Base Surfaces using ANSYS		161
A.1	Basic essential ANSYS commands	162
A.2	Five parameter base surface remodelling example.....	164
A.3	12 parameter base surface remodelling example	167
A.4	Calculating $THXY$, $THYZ$, $THZX$ for a new local coordinate system	170
A.5	12 parameter frustum ANSYS command textfile	175
Appendix B ANSYS to Nottingham Drape It.....		178
B.1	Nottingham DrapeIt input geometry file overview.....	179
B.2	Node Extraction and Organization Steps.....	181
B.3	Element Extraction and Organization Steps	185
Appendix C Running Simulations and Extracting Data		188
C.1	Running draping simulations in Drape It.....	189
C.2	Edge vector calculation and specific element data extraction	194

C.3	MATLAB script for edge element extraction	196
C.4	Warp and weft orientation angle calculations.....	199
Appendix D	<i>BD100</i> Mould Points	202
Appendix E	MATLAB Scripts.....	207
Appendix F	SolidWorks Macro	215

List of Figures

Figure 1-1: Composite reinforcement types [3].....	2
Figure 2-1: Different stitching patterns.....	11
Figure 2-2: Warp knit without yarns (left); warp knit with yarns (right) [13].....	12
Figure 2-3: Plain weft knited fabric [14]	12
Figure 2-4 Weave patterns: (a) plain weave (one warp over one weft), (b) 4 harness satin (one warp over three weft) [16]	13
Figure 2-5: 2D braided pattern [17]	13
Figure 2-6: (a) Initial angle between warp and weft, (b) weave shears (c) weave locking angle is reached [21] [22].....	15
Figure 2-7: Fabric during shear: (a) unsheared state (b) locking shear angle, for different types of fabric [27].....	16
Figure 2-8: Kinematic drape algorithm [32]	18
Figure 2-9: Determination of node position [34]	19
Figure 2-10: RTM process [38]	22
Figure 2-11: VARTM process [39].....	23
Figure 2-12: Scree plot [48]	27
Figure 3-1: Flat, single curvature and double curvature base surfaces	29
Figure 3-2: Tapered single curvature (left) and tapered double curvature (right): $R1 < R2$	30
Figure 3-3: Twisted flat (left), single curvature (centre), double curvature (right) base surfaces	31
Figure 3-4: Traditional conical frustum	32
Figure 3-5: Conical frustum geometric parameters	34
Figure 3-6: Base surface arc lengths and midpoints	36

Figure 3-7: Base surface coordinate system translation	37
Figure 3-8: Arc radius	39
Figure 3-9: Portion of MATLAB Script 1 for calculating centre points (Method 1)	40
Figure 3-10: Plane vectors and normal vector	41
Figure 3-11: MATLAB Script 2: Portion for calculating centre point (Method 2)	43
Figure 3-12: Script 2 solutions (left); correct solution identification (right)	44
Figure 3-13: Rotation angle α	45
Figure 3-14: Base surface for example 1	46
Figure 3-15: Base surface for example 2	49
Figure 4-1: <i>FR</i> (left) and <i>BD100</i> (right) moulds.....	55
Figure 4-2: Metallic and <i>INSIZE</i> digital radius gauges	55
Figure 4-3: Zeroing radius gauge (left) and radius measurement (right).....	56
Figure 4-4: MicroScribe G2 component diagram [50]	57
Figure 4-5: <i>FR</i> mould base surface (left) and vertex identification (right).....	58
Figure 4-6: Surface splitting and additional points.....	59
Figure 4-7: <i>BD100</i> symmetric half-mould decomposition	62
Figure 4-8: Uniform flat base surface (left); “titled” flat base surface (right).....	65
Figure 4-9: Five parameter core conical frustum.....	66
Figure 5-1: Draping parameters: Starting position and orientation	68
Figure 5-2: Draping parameters: Initial angle between the warp and weft, and fibre spacing	69
Figure 5-3: Double curvature base surfaces.....	72
Figure 5-4: Near-flat and single curvature base surfaces.....	73
Figure 5-5: Increasing base surfaces surface curvature	73

Figure 5-6: Runs #1 and #2 draping simulation results	77
Figure 5-7: Runs #3 and #4 draping simulation results	77
Figure 5-8: Runs #5 and #6 draping simulation results	78
Figure 5-9: Run #7 and #8 draping simulation results.....	79
Figure 5-10: Runs #9 and #10 draping simulation results	79
Figure 5-11: Runs #11 and #12 draping simulation results	80
Figure 5-12: Runs #13 and #14 draping simulation results	81
Figure 5-13: Runs #15 and #16 draping simulation results	81
Figure 5-14: Basic 16-run shear angle Scree plots	84
Figure 5-15: Basic 16-run shear angle Scree plots, M removed.....	85
Figure 5-16: Runs #R1 and #R2 draping simulation results.....	86
Figure 5-17: Runs #R3 and #R4 draping simulation results.....	87
Figure 5-18: Runs #R5 and #R6 draping simulation results.....	88
Figure 5-19: Runs #R7 and #R8 draping simulation results.....	88
Figure 5-20: Runs #R9 and #R10 draping simulation results.....	89
Figure 5-21: Runs #R11 and R12 draping simulation results.....	89
Figure 5-22: Runs #R13 and #R14 draping simulation results.....	90
Figure 5-23: Runs #R15 and #R16 draping simulation results.....	91
Figure 5-24: Fully reflected 16-run shear angle Scree plots.....	93
Figure 5-25: Fully reflected 16-run shear angle Scree plots, M removed	94
Figure 6-1: Four-edge base surface and three-edge base surface	98
Figure 6-2: Average shear angle database results for three edge base surface	99
Figure 6-3: Average shear angle results and linear function (main effects only).....	101

Figure 6-4: Line plot comparison: a) interactions of 2; b) interactions of 3; c) interactions of 4 d) interactions of 5; e) interactions of 6 f) interactions of 7 parameters	103
Figure 6-5: Line plot comparison: a) Main effects; b) interactions of 2; c) interactions of 3 d) interactions of 4; e) interactions of 5 f) interactions of 6 parameters	104
Figure 7-1: Shear progression.....	106
Figure 7-2: Projected orientation change on a flat base surface	108
Figure 7-3: Shear progression: changing yarn orientation (left) vector progression paths (right)	109
Figure 7-4: Max shear angle	110
Figure 7-5: Head-to-tail vector summation.....	111
Figure 7-6: Progressing shear angles (left), varying shear angle along common edge (right) ...	113
Figure 7-7: Single and double curvature model (left) shear progression (right)	114
Figure 7-8: Maximum shear angle vectors	115
Figure 7-9: Draping double curvature configurations simulation (left), physical surface (right)	118
Figure 7-10: Box corner.....	120
Figure 7-11: Angle between surfaces model (left) and physical draping results (right)	121
Figure 7-12: Symmetrical preliminary shear (left), unsymmetrical preliminary shear (right) ...	122
Figure 7-13: Symmetric (left) and non-symmetric (right) preliminary shear results	122
Figure 7-14: Single edge chamfer.....	124
Figure 7-15: Single edge chamfer (a) yarn misalignment (b) preliminary shear drape test	124
Figure 7-16: Uniform corner chamfer full body (left) and AC plane (right)	125

Figure 7-17: Chamfer analysis: (a) cube; (b) AC' triangle projection; (c) BC' triangle projection	126
Figure 7-18: Double edge chamfer (a) yarn misalignment (b) yarn alignment (c) preliminary shear test	126
Figure 8-1: (a) Connections between two base surfaces (b) Connections between three base surfaces	131
Figure 8-2: Impractical draping strategy	132
Figure 8-3: Base surface links	133
Figure 8-4: Mould with 7 base surfaces	135
Figure 8-5: Draping sequence boundary conditions	136
Figure 8-6: Consolidation of boundary conditions a) Boundary conditions; b) Chosen boundary condition input; c) Results from both boundary conditions; d) Average of results	138
Figure 8-7: Orientation angles along base surface edges	138
Figure 9-1: Surface decomposition and vertex labelling	139
Figure 9-2: Warp and weft starting orientations (a) aligned and (b) misaligned with edges	141
Figure 9-3: Pattern 1 (left); Pattern 2 (right)	143
Figure 9-4: Surface 3 boundary conditions	144
Figure 9-5: Pattern 1 (left) and pattern 2 (right) partial demonstration results	144
Figure 9-6: Draping strategy 1 (left) and draping strategy 2 (right) demonstrations	147
Figure 9-7: Physical demonstrator results	151

List of Tables

Table 1-1: Engineering properties for select materials	2
Table 2-1: Example of a 12-run Plackett-Burman design [45]	26
Table 3-1: Example vertices and parameters for Method 1	47
Table 3-2: Translated coordinates for example 1	48
Table 3-3: Conical frustum parameters for example 1	48
Table 3-4: Example vertices and parameters for Method 2	50
Table 3-5: Translated coordaintes for example 2	51
Table 3-6: Conical frustum parameters for example 2	51
Table 4-1: FR mould base surface vertices	60
Table 4-2: <i>FR</i> mould conical frustum parameters	61
Table 4-3: <i>BD100</i> mould conical frustum parameters	63
Table 5-1: 16-run design experiment for the analysis of geometric and draping parameters.....	71
Table 5-2: Taguchi experimental design including parameter modalities	71
Table 5-3: 16-run draping simulations.....	76
Table 5-4: Basic 16-run Taguchi simulation results	82
Table 5-5: 16-run design experiment for fully reflected study	86
Table 5-6: Fully reflected 16-run Taguchi experiment results	92
Table 6-1: Edge <i>DC</i> simulation results, modalities and responses	100
Table 7-1: Shear progression example: Core conical frustum parameters	114
Table 7-2: Warp and weft yarn orientation vectors	115
Table 7-3: Increasing shear angle into single curvature surface	116
Table 7-4: Shear progression difference	117

Table 7-5: Maximum shear angle and head-to-tail vector summation results.....	117
Table 8-1: Draping strategies starting from base surface 1	134
Table 9-1: Partial draping strategies	142
Table 9-2: Linear function results for surface 3 draping from surface 2	145
Table 9-3: Linear function results for surface 3 draping from surface 4	145
Table 9-4: Base surface 2 edge boundary conditions	148
Table 9-5: Base surface 4 edge boundary conditions	149
Table 9-6: Final draping results along edge <i>DC</i> for draping strategy 3	150

1 Introduction

1.1 Background on composite materials

A composite material is a material made of two or more distinct materials with vastly different properties which, when combined, produce a material with properties that differ from those of the individual component materials. The individual materials remain distinct from one another within the finished material thus differentiating composite materials from solid solutions or mixtures [1]. For a composite featuring two main phases, these are often termed the reinforcement and the matrix.

Composite reinforcements can take a variety of different forms including fine particles, discontinuous chopped fibres, long continuous fibres or woven fabrics, as well as other forms, Figure 1-1. In this thesis, reinforcements in the form of fabrics are the main focus. Long continuous fibres feature high modulus, high tensile strength, and low density when compared to other engineering materials, Table 1-1. Such fibres often have very small diameters, typically $10\text{ }\mu\text{m}$ or less. Smaller diameter fibres offer an increase in surface area per unit mass for fibre-matrix interfacing, resulting in better load transfer between matrix and fibres, a reduction in bending radius causing less fibre breakage during handling, a decrease in the amount of defects on the surface of a fibre and a higher strength [2]. Dry fibres and fabrics are capable of withstanding large tensile loads but buckle under very low compressive loads.

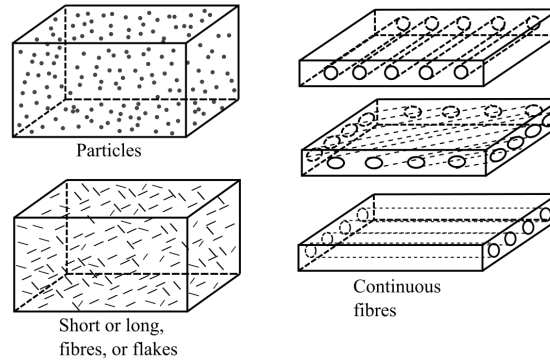


Figure 1-1: Composite reinforcement types [3]

Table 1-1: Engineering properties for select materials

Materials	Density (g/cm^3)	Tensile Strength (MPa)	Young's Modulus (GPa)
Ti-6Al-4V Titanium [4]	4.41	1200	119
4132 Alloy Steel [5]	7.85	1210	210
Carbon Fibre [6]	1.77	4205	239
Glass Fibre [7]	1.54	3310	68.9

Fibre reinforcements are infused with a liquid matrix that cures into a solid polymer holding, the fibre in place. The matrix restricts fibre movement, transfers the loads to the fibers and allows for parts to be constructed with directional properties [8]. The matrix also protects the fibres from environmental hazards. A polymer matrix is commonly an epoxy or another thermosetting or thermoplastic resin thus these composites are often referred to as polymer matrix composites (PMCs).

Early applications of PMCs in aircrafts used boron fibre and carbon fibre reinforced epoxies for many wings, fuselage components, and other structural components. This reduced the weight by up to 30% for these components when compared to the metallic components used and allowed flexibility in manufacturing [9]. High specific stiffness and strength, along with relative ease of

manufacturing and ability to adjust final part properties to specific applications, has led to a major increase in the use of composites. Major examples of composite applications include commercial airplanes such as Boeing 787 Dreamliner and Airbus A350. These aircrafts feature 53% [10] and 50% [11] of composite materials by structural mass, respectively. Other advantages of PMCs over common engineering materials include relatively low tooling costs, corrosion resistance, and exceptional design flexibility.

1.2 Project overview and contributions

This section explains the project overview, outlines the main objectives for this work and highlights terminology used throughout this thesis.

The overall focus of the project is the manufacturing of non-structural composite aerospace components made by liquid composite moulding (LCM) processes. The parts are all thin shells with varying degree of complexity. As part of these processes, a significant amount of time is dedicated to the design of methods used for cutting and forming 2D textiles into 3D components, defined as preforms. One or more textile patterns consisting of multiple, stacked textile plies with the same outlines and laid with the same yarn orientations are draped on a mould as the first step to the manufacturing of composite components. To position and conform these plies onto a mould, a manual method known as draping is used. Ideally, a single pattern would be used for making an entire preform, but often this is not possible due to the complexity, geometric features and size of the corresponding part. Properties of the fabrics that determine its propensity to drape include resistance to in-plane shear and to bending, as well as stability, which all influence results to different degrees depending on the geometric features found on the mould. Some element of

behaviour can become problematic in complex cases. Details of the industrial process can be found in the Master's thesis of Sébastien Gagné [12].

By designing a robust draping strategy, which consists of a series of steps to be followed by technicians in order to drape patterns on a mould, parts can be manufactured more quickly. The draping strategy must also enable low variance so that reproducible, high quality parts can be manufactured. Other considerations include the choice of the fabric to be used, which is also dependant on mechanical requirements on the final component.

Industrial partner Hutchinson Aéronautique et Industrie operates in a competitive industrial space where predicting design, manufacturing and preform quality is crucial to building competitive quotes for new clients. A client submits a CAD model of a part and Hutchinson must create a quote very quickly, outlining key preforming characteristics as well as costs, to successfully and profitably secure the client. The production of a quote requires that draping strategies be designed; this can be difficult in the case complex components, without a physical mould being available. The goal of the project, which is the context that defines this thesis, is to develop a predictive software tool that can design draping strategies quickly for new and existing parts [12]. The predictive draping tool will:

- Identify a preform design that will streamline the manufacturing process and make it reproducible;
- Identify the steps necessary for manufacturing the part, and communicate the procedure to technicians clearly through plybooks;
- Choose a fabric that is most suitable for the draping challenges induced by the geometric features specific to the part - this point is not addressed in this thesis;

- Identify a preforming strategy with low variability and high final part quality.

Draping software already exists that is capable of assisting towards defining draping strategies. However, once a part is received these simulations are no better than trial and error on a physical mould. The number of possible draping scenarios for an individual mould is very high; attempting to test all of them and identify the best one would be extremely time consuming. A draping strategy that stands out after a few trials with such software is most likely to be chosen because it works and, not necessarily because it is the best. The approach developed for this project and developed in this thesis moves all the simulation work ahead of receiving the geometry. The surfaces are expressed so that an algorithm can probe very quickly through a very large number of scenarios, identifying the best one amongst them.

In this work, complex mould surfaces are split into individual geometric features labelled base surfaces. These base surfaces are readily identified in CAD models by lines and edges drawing distinct differences between features. Edges common to neighbouring surfaces are used as direct links between the base surfaces, so they can be used in writing explicit expressions of draping strategies. Draping strategies are designed by sequencing transitions between individual base surfaces through common edges. Transitions occur sequentially until all the mould is draped or a pattern reaches its drapability limit. Each base surface induces unique draping properties quantifiable through draping simulations. Each base surface is also identified through vertices, lengths and radii which can be characterized into a set number of geometric parameters.

Base surfaces are labelled in this work as flat, single curvature and double curvature, each capable of featuring a level of twist, as highlighted in Section 3.1. Base surfaces are parameterized using the conical frustum, a parameterization tool devised as a part of this thesis work and used for structuring a database of shear and yarn orientation results (Database A) that is part of the

predictive tool. Database A contains results from draping simulations for individual base surfaces generated using varying conical frustum geometric parameters and as well as draping parameters such as drape starting position, drape starting orientation, fibre spacing and initial angle between warp and weft yarns. The conical frustum parameters act as direct links between geometric features on the defined as real geometric parameters, and results contained in the database.

Database A will contain a very large number of results, even for a low number of geometric and draping parameters. Any draping strategy would involve many different combinations of these parameters and require to a large amount of information to be considered, hence readily available. To compile a complete database, an unrestricted array of shear results would have to be generated for any set of geometric and draping parameters, even if any parameter was documented at only two modalities. In order to address efficiency, Taguchi methods were used, allowing a drastic reduction in the number of simulations required in order to populate the database and provide a clear framework as to what information needs to be quantified, and what information is less important. Furthermore, Taguchi methods lead naturally towards linear functions used for interpolations between database results, which are required since it is extremely unlikely that all surfaces encountered in a specific industrial case will be represented exactly in database results. Database population is a continuous process, and filling the database will eventually require a scripted, automated process. However, a minimally populated version is very useful especially if the modalities of parameters are selected, once all the above elements are assembled. The predictive tool uses the database and geometric information gathered from the mould surface to probe all draping strategies quickly, identifying the best one.

This thesis details solutions for a first phase of the predictive tool development featuring:

- Proposing a formalism specifying geometric and draping parameters used for building Database A; the conical frustum, along with a hierarchy for its geometric parameters;
- Identifying specific modalities used in Database A;
- Selecting and quantifying output responses from draping simulations performed on base surfaces;
- Performing draping simulations following Taguchi plans, beginning the construction of Database A and developing linear functions for the interpolation of results;
- Establishing precise procedures for quantifying the link between base surfaces in the mould.

Rather than assessing draping scenarios using the full mould, investigating draping behaviour over individual base surfaces and linking their results allows numerous draping strategies to be probed quickly. This predictive tool will drastically reduce the time required for designing draping strategies, assisting Hutchinson in producing quotes for better parts more quickly and more reliably.

1.3 Section summary

This thesis is organized into 10 chapters and 6 appendices. Chapter 1 provides background on composite materials, detailing some industrial use.

Chapter 2 is a literature review, focusing mainly on manufacturing methods for composites, specifically prior to resin injection, deformation modes of dry fabrics, factorial design plans and a comprehensive review of the principles behind Nottingham Drapelt, the draping software used extensively in this thesis.

Chapter 3 evaluates geometric features commonly observed on industrial mould surfaces. The development of a multi-geometric parameter characterization tool, the conical frustum, is presented in this chapter. Details of the steps required for identifying and calculating geometric parameters of the conical frustum from geometric features on mould surfaces are presented.

Chapter 4 presents a hierarchy of geometric parameters defining the conical frustum. Two mould surfaces are decomposed into individual base surfaces, and calculations for their conical frustum parameters are performed. Through these calculations, specific conical frustum parameters are identified that change more frequently than others, thus establishing a core conical frustum featuring a smaller number of parameters which can be used for modelling most base surfaces using a reduced number of geometric conical frustum parameters.

Chapter 5 details draping simulations performed, which were planned following Taguchi methods for experimental design to evaluate the effects of geometric and draping parameters on in-plane shear.

Chapter 6 uses the results from Chapter 5 towards building Database A, a database of in-plane shear and yarn orientation angles extracted from draping simulations. Results put into the database lead to the development of linear functions for in-plane shear, warp orientation and weft orientation angles at different locations on base surfaces as a function of conical frustum parameters and draping parameters.

Chapter 7 discusses the linking of base surfaces and phenomena found in specific cases. Topics include shear progression, effects of subsequent double curvature configurations and the effect of angles between surfaces.

Chapter 8 presents the draping algorithm developed for this thesis. It outlines the organization and definitions of draping strategies, how conical frustum parameters are used to probe the database, how connections are made between base surfaces, and presents a method for consolidating multiple boundary conditions.

Chapter 9 presents a demonstration of the draping algorithm uses a physical mould surface. The demonstrator shows how base surfaces are used to calculate conical frustum geometric parameters and how those parameters are used with linear functions and draping parameters to probe draping strategies.

Finally, Chapter 10 provides an overview and conclusion on the contributions this thesis has towards the project, details current work being performing, and suggests directions of future work.

2 Literature Review

2.1 Introduction

Draping composites using the fishnet algorithm is a well-established method for predicting composite deformation, thus new methods in literature have not been thoroughly investigated but have consistently used the fishnet algorithm since its conception. The methods presented later in this thesis use the kinematic draping algorithm as it exists following a new technique for quantifying their results, using individual geometric features found on mould surfaces to link together a full draping strategy rather than attempting to cover a full surface with a single pattern. As such, this section is used to present background information on preform architectures, fabric deformation, draping algorithms and composite manufacturing, as well as highlight the main features of factorial experimental designs used in this work.

2.2 2D textile reinforcements

For composite components to be manufacturing using LCM processes, fibre reinforcements often need to be joined into sheets labelled preforms. Performs are classified into different types based on their manufacturing process and final 2D architecture including stitched textiles, knits, weaves, and braids. Such reinforcements are labelled as fabrics and are commonly supplied in large rolls and manufactured out of various grades of glass and carbon fibres.

To achieve a fabric architecture, individual fibres are first assembled into tows, untwisted bundles of fibres, or yarns, twisted bundles of fibres. Stitched textile reinforcements feature warp and weft yarns that do not overlap one another but are held together with non-structural stitched threads. These reinforcement types are identified as non-crimp fabrics (NCFs) since the yarns do not

interlace with one another, therefore remaining essentially flat. Layers of unidirectional fabrics are often stacked and stitched together using a lockstitch; an upper and lower thread entwined, holding layers of reinforcements together, as seen in a standard sewing machine. Different stitching patterns can be used influence the deformation of the stitched fabrics, Figure 2-1.

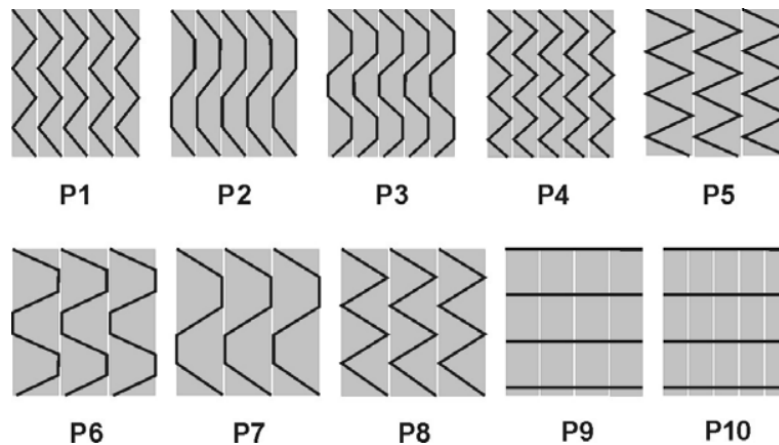


Figure 2-1: Different stitching patterns

Knit reinforcements are identified as warp knits and weft knits. Warp knits are differentiated by the stitching pattern used to hold the layers of fabric together and are classified as NCFs. Similar to stitched reinforcements, warp knits use non-structural threads to hold layers of fabric together. The major difference between stitched reinforcements and knits is that knits interlace around yarns following a specific pattern that avoids piercing through the yarns and is dependent on the orientations of the layers being stitched together. Figure 2-2 shows an example of a tricot stitch warp knit without and with layers of yarns.

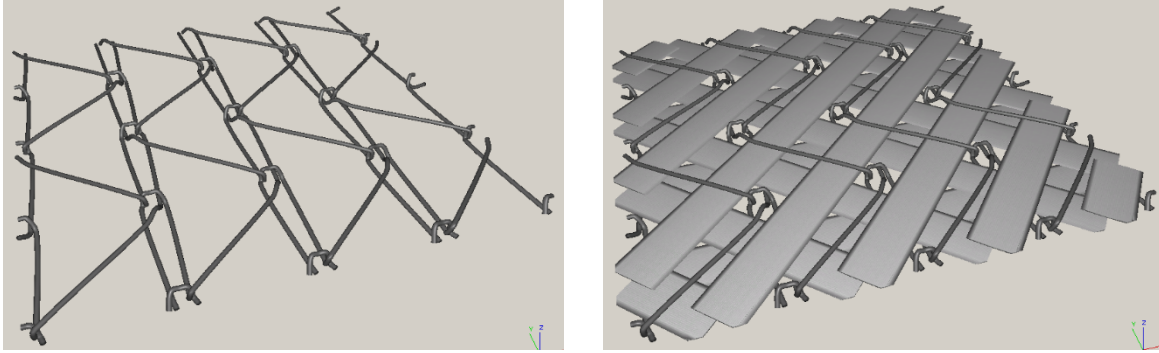


Figure 2-2: Warp knit without yarns (left); warp knit with yarns (right) [13]

Weft knits do not use non-structural stitched reinforcements but interlock yarns into a specific pattern. This technique is frequently used in knitting wool or cotton but has been applied to fibre reinforcements in composites. Weft knits do not feature any warp yarns but follow a repeating pattern of weft yarns horizontally and vertically, producing a fabric from a single yarn, Figure 2-3. Weft knits are not commonly used in manufacturing composites since they typically lead to a lower fibre volume fraction in the final component. NCFs and weft knits are not discussed further in this thesis.

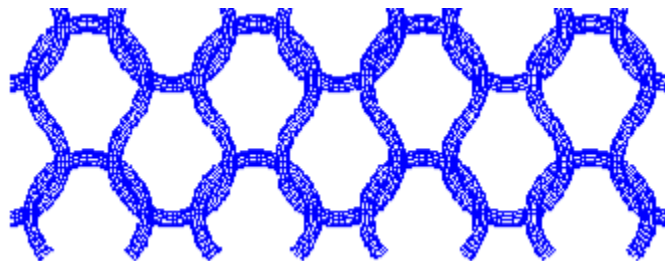


Figure 2-3: Plain weft knited fabric [14]

Woven bidirectional fabrics feature warp yarns that run the entire length of the fabric, and weft yarns that are interlaced under and over the warp yarns. The overlaps in the reinforcements creates crimp in the individual yarns since they are forced to bend around one another thus differentiating these reinforcement types from NCFs, [15]. Several different weave patterns exist including plain

weaves, satins, and twills, which are classified by the interlacing pattern of the warp and weft yarns, Figure 2-4. It is also possible for triaxial weaves to exist in applications where textiles are required to be close to isotropic. These weaves deform differently from bidirectional weaves and therefore are not discussed further in this thesis.

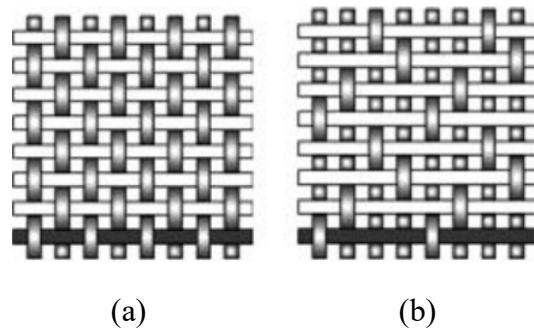


Figure 2-4 Weave patterns: (a) plain weave (one warp over one weft), (b) 4 harness satin (one warp over three weft) [16]

Braided textiles consist of three or more intertwined yarns that form a continuous tubular structure, Figure 2-5. To create a braided textile, the tubular section is typically slipped over a mould of the desired shape or braiding is performed around the mould. Since braided textiles feature multi-axial yarns, deformation modes are different than those found in stitched fabrics, knits and weaves hence braids are not discussed further. The focus in this thesis is woven bidirectional fabrics.

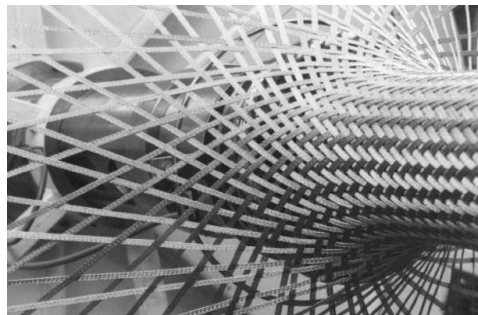


Figure 2-5: 2D braided pattern [17]

2.3 In-plane shear of woven dry fibre reinforcements

When draping fibre reinforcements onto mould surfaces, any change in curvature will cause changes in fibre orientations, fibre volume fraction and permeability, leading to non-homogenous mechanical properties in the final component [18]. Forming over complex geometries requires a degree of in-plane shear so the fabric can conform to the surface. However, this can quickly lead to wrinkling and resin flow issues. Manufacturers are not only concerned with the mechanical performance of the final composite but also with how easily it is to process and drape the fabrics, and with the overall cost of the entire manufacturing process [19]. When considering draping, in-plane shear is the primary mode of fabric deformation.

The analysis by Mack and Taylor [20] provides a more in-depth understanding on draping woven fabric over complex geometries. The proposed pin-jointed net model assumes a pinning effect at the cross over points of warp and weft yarns, much like pin joints with straight, rigid bars whilst yarns are under axial tension. These pin joints allow the warp and weft yarns to rotate around these joints, which effectively correspond to yarn crossovers inducing in-plane shear onto the fabric, Figure 2-6. This mechanism is what allows 2D textiles to be draped over 3D surfaces such as a hemisphere, solely by yarn rotation at the crossovers. Shear angle β can easily be calculated using the spacing between the individual yarns.

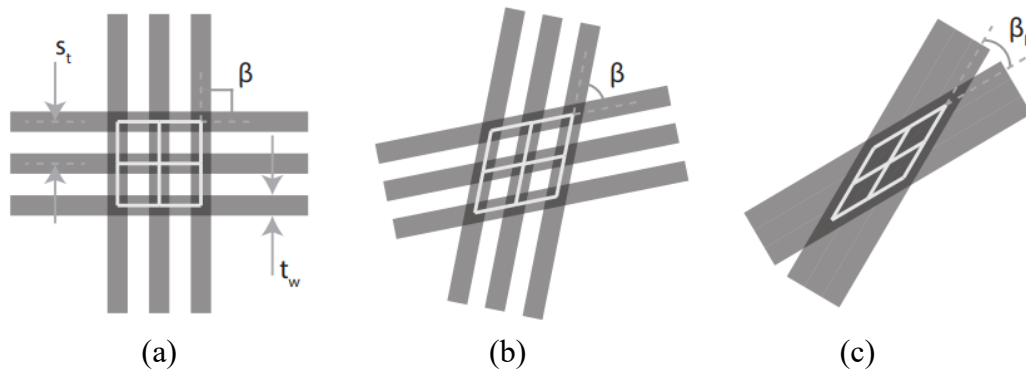


Figure 2-6: (a) Initial angle between warp and weft, (b) weave shears (c) weave locking angle is reached [21] [22]

In plane shear has a limit known as the locking angle, denoted by β_L in Figure 2-6 c). When the locking angle is reached, the interaction between yarns causes an increase in load between the yarns. Without room for deformation in-plane, the build up of compressive forces causes fibre buckling, ultimately leading to wrinkling [23]. The locking angle of woven fabrics is dependent on the weave pattern and mechanical properties of the fibres. Fabrics can tolerate relatively high amounts of in-plane shear before reaching their locking angle. This fabric dependent property is typically measured using biased extension and picture frame tests, between 20° and 65° [24] [25] [26]. In general fabrics with more loose weaves such as satins and sateens tend to reach higher levels of shear prior to reaching their locking angle. Figure 2-7 shows a series of different woven fabric types, along with differences between an unsheared state and the point at which the fabrics reach their locking angle.

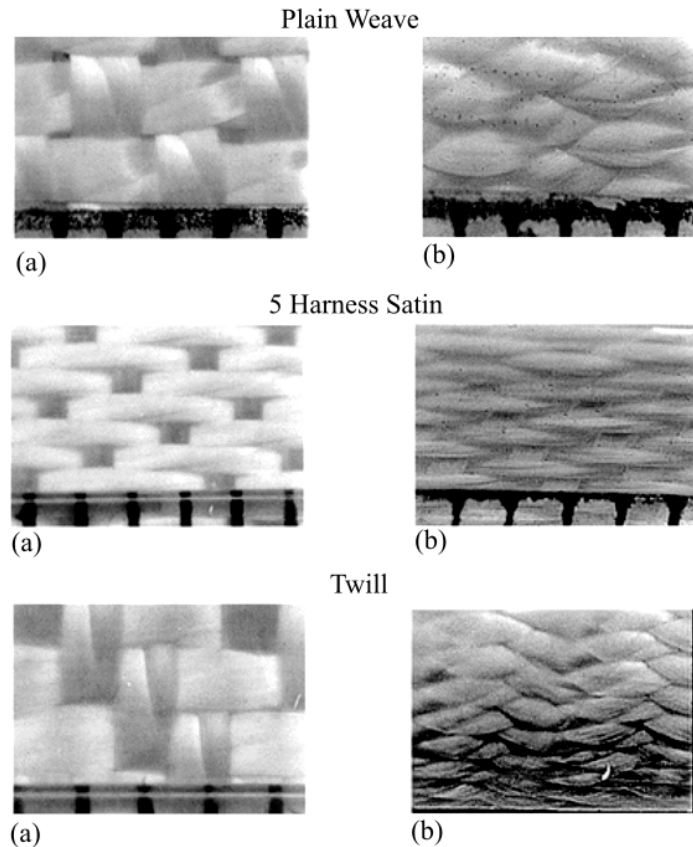


Figure 2-7: Fabric during shear: (a) unsheared state (b) locking shear angle, for different types of fabric [27]

High amounts of shear and wrinkling lead to defects in composite parts such as unwetted fibres, due to resin flow issues, or resin rich areas where fibres did not conform to the mould surface. To ensure complete fibre wetting and no wrinkling, draping strategies need to be designed to ensure excellent coverage with minimal shearing. However, in order to produce composite components quickly, this stage of the manufacturing process needs to be performed before receiving a physical mould.

2.4 Draping algorithms

To assist in manufacturing composite components, in-plane shear mechanics over mould surfaces have been used to develop draping algorithms. These draping algorithms assist in predicting shear behaviour of ideal woven structures.

2.4.1 Kinematic draping algorithm

For draping textiles, the main assumption is that the woven textiles acts much like a fishnet; a square mesh made of inextensible segments. These segments correspond to yarns running in two directions, warp and weft, which are originally orthogonal [28]. The kinematic algorithm is dependent on creating orthogonal geodesics, which cross each other at the start of draping. Geodesics, defined as a local length-minimizing curves that preserve a direction on a surface [29], project constrained warp and weft yarn paths over the mould. Stemming from the fishnet algorithm the kinematic draping algorithm was developed by Robertson *et al.* [30] using a hemisphere and follows five assumptions laid by Mack and Taylor:

1. Fibres are inextensible;
2. Fibre crossovers act as pin joints with no slip;
3. Fibre segments are straight between joints;
4. Uniform surface contact is maintained between yarns;
5. The geometry and fabric layer are infinitely thin.

As the geodesic moves over the surface, individual node locations are calculated and represent the fabric weave geometry using bilinear patches attributed to an ideal fabric behaviour over the surface [31]. The result from the draping algorithm is a shear pattern that can be used for predicting

the outcome of draping fabrics over geometric features. Applications for this kinematic algorithm extended into industries where forming fabric over complex surfaces was needed. Further details on the kinematic drape algorithm specifically for Nottingham DrapeIt software is presented in Section 2.4.4

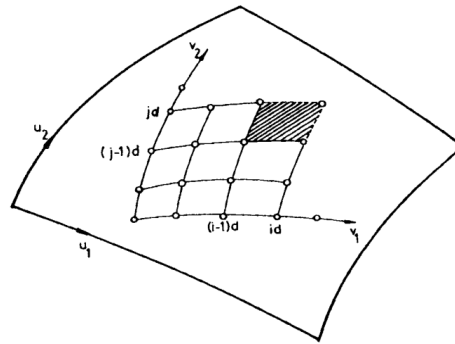


Figure 2-8: Kinematic drape algorithm [32]

2.4.2 Energy drape algorithm

Kinematic draping algorithms work very well for symmetric shapes however for non-symmetric geometry, yarn path generation can be limited. The energy drape method uses finite element simulations as an approach to overcome the problem above [33]. Still using the kinematic draping method, the strain energy required to produce each mapping result is calculated. The mapping that results in the lowest strain energy is assumed to represent the actual behaviour of the fabric. This method may overcome the unsymmetrical geometric problems but increases the computation time. Unlike the kinematic method, this method does not rely on initially determined geodesics to extend over the surface. Instead, these lines are determined as the coverage progresses allowing the simulated fabric to move more freely.

2.4.3 Nottingham DrapeIt

Developed at the University of Nottingham, DrapeIt is a software package that allows draping simulations to be performed on surface meshes. A kinematic draping algorithm [34] and a energy draping algorithm [33] are used to simulate deformation of woven fabrics. Fibre rotation at the warp and weft crossover points (nodes) is the only mode of deformation assumed for these algorithms.

2.4.4 Nottingham kinematic draping algorithm

This section outlines the kinematic draping algorithm that Nottingham DrapeIt uses to project shear patterns over geometric surfaces. The algorithm uses initial fibre spacing S and known node positions along constrained fibre geodesics to calculate the coordinates for subsequent nodes over the entire surface. Nodes are identified with indices (m, n) starting at $(0,0)$, the point of intersection of the two constrained fibres. A point of intersection of two spheres with the geometry surface is used to calculate (m, n) repeatedly until the entire surface is covered. Each sphere has a radius of the initial fibre spacing S , which are centred at adjacent nodes $(m-1, n)$ and $(m, n-1)$, Figure 2-9.

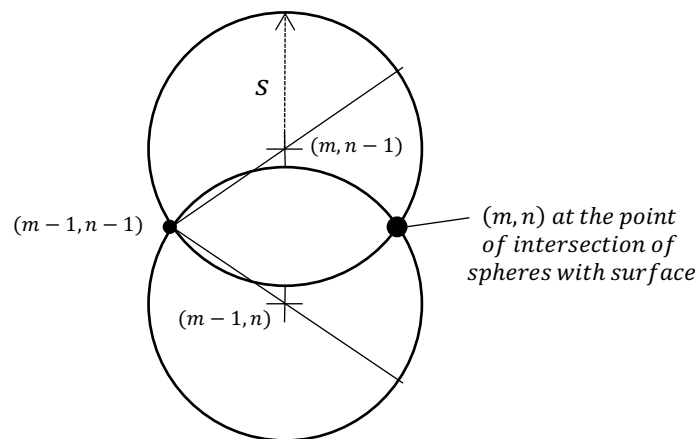


Figure 2-9: Determination of node position [34]

The adjacent nodes begin along the constrained fibres and therefore are always known. The portions of the surface identified by the four nodes are represented by bilinear patches. A reduction in initial fibre spacing yields refined results of the shear pattern for the surface. The equations for the spheres, the parametric equation of the bilinear patch with corner points P_{mn} , and the equation of the plane containing the patch are as follows:

$$(x_{m,n} - x_{m-1,n})^2 + (y_{m,n} - y_{m-1,n})^2 + (z_{m,n} - z_{m-1,n})^2 = S^2 \quad (1)$$

$$(x_{m,n} - x_{m,n-1})^2 + (y_{m,n} - y_{m,n-1})^2 + (z_{m,n} - z_{m,n-1})^2 = S^2 \quad (2)$$

$$P[u, v] = (1 - u)(1 - v)P_{00} + u(1 - v)P_{10} + (1 - u)vP_{01} + uvP_{11}, \quad (3)$$

$$u, v \in [0, 1]$$

$$ax_{m,n} + by_{m,n} + cz_{m,n} = d \quad (4)$$

where coefficients a , b and c can be calculated using and d is calculated by substituting P_{11} into Equation 4.

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = (P_{10} - P_{00}) \times (P_{01} - P_{00}) \quad (5)$$

Combining equations 1, 2 and 4, the given node (m, n) locations calculated for the known positions of nodes $(m-1, n)$ and $(m, n-1)$. As mentioned, the calculation begins at the intersection of the constrained fibre paths thus nodes $(0, 0)$, $(0, 1)$, and $(1, 0)$ are known, therefore coordinates for node $(1, 1)$ may be calculated. The process is repeated until the surface is completely covered producing a shear pattern for the specified orientation and starting position of the constrained fibres. Once a shear pattern is generated, Nottingham DrapelIt allows shear angles and warp and

weft orientation vectors at every element on the surface to be extracted which is used extensively throughout this thesis.

2.5 Manufacturing composites

Production of composites can be distinguished into two major types, starting with the reinforcement type selected for draping. Using fibre reinforcements with pre-impregnated resin (prepregs) that is allowed to partially cure can be draped onto a mould surface benefiting from low void content, uniform fibre alignment, and ease of handling [35]. This identifies one major type of composite manufacturing processes. The other classification for composite manufacturing uses dry fabric layup is identified as liquid composite moulding (LCM) and is the focus in this thesis.

LCM benefits from higher manufacturing rate, extended fabric shelf life, lower manufacturing costs and higher flexibility during draping when compared to prepreg. LCM is also more economical than prepreg based manufacturing processes and offers a more practical option for manufacturing small series of components quickly without loss of component quality.

PMCs are manufactured using LCM in two major steps: fibre reinforcement positioning and resin infusion. The first step involves arranging reinforcements into patterns that are capable of conforming to the geometric features of open or closed moulds. The geometric features of a mould are dictated by the final component design. Once the reinforcements are positioned, resin is infused into the reinforcement, most often using a liquid thermosetting resin which bonds to the reinforcements, solidifying into a single rigid component [36]. Two major LCM processes, resin transfer moulding (RTM) and vacuum assisted resin transfer mould (VARTM), are detailed in the following sections.

2.5.1 Resin transfer moulding

Resin transfer moulding is a low-pressure injection process. Dry fibre reinforcements, single layer or multilayer fabric stacks are draped into an open mould. Once the fabric is conformed to the surface, the mould is closed, and resin is commonly injected at a constant flow rate with a volumetric pump. This closed mould process yields good surface finishes on both sides of the final component. However, issue may arise during resin injection since the mould is closed. After the mould is filled, resin flow is halted and cure initiates. The final step in RTM is demoulding, which can be performed once the resin is sufficiently cured [37].

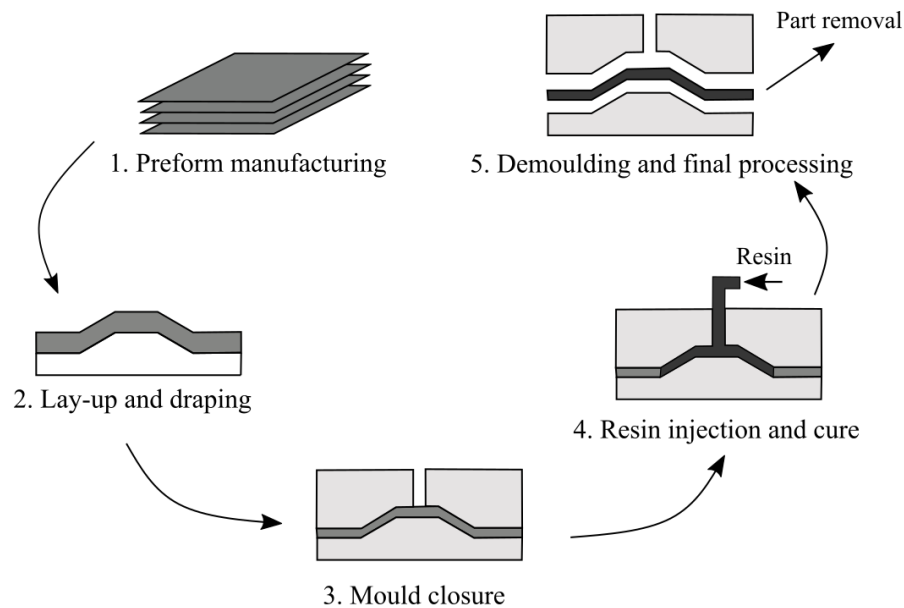


Figure 2-10: RTM process [38]

2.5.2 Vacuum assisted resin transfer moulding

Vacuum assisted resin transfer moulding (VARTM) is widely used for composite manufacturing, due to its flexibility and scalability. The process is capable of manufacturing complex components with outstanding quality at a much lower cost than RTM. VARTM uses a single-sided mould

surface with the following components: fibre reinforcement, peel ply, flow distribution medium and helical tubes. All these layers are sealed under a plastic film or silicon vacuum bag as shown in Figure 2-11. A vacuum is applied under the bag, drawing resin into the reinforcement. A maximum pressure difference of 1 bar is reached from the applied vacuum causing resin flow to be slow, hence the flow distribution medium is need. The flow distribution medium has high in-plane permeability and promotes resin flow through the fibres, allowing the resin to gradually infuse into the reinforcements. Once the resin has flowed through the length of the assembly, it is left to cure under vacuum. Afterwards the parts are demoulded, producing a final composite component.

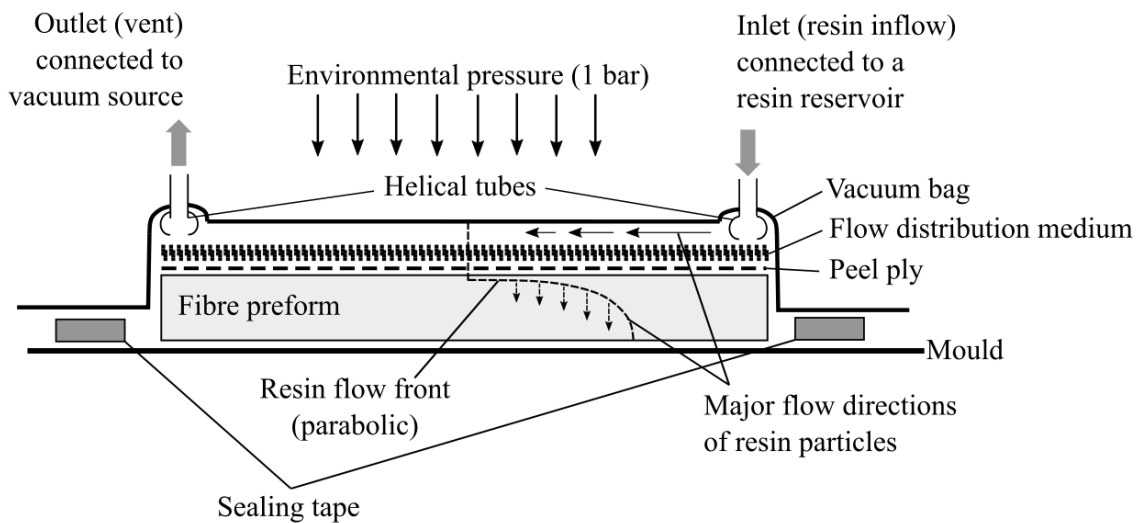


Figure 2-11: VARTM process [39]

2.6 Statistical experimental design

Traditionally, design of experiments (DOE) is a statistical approach to investigate a system or a process that allows conclusions to be drawn on the significance of input variables to out, either as their own influences or the interactions between variables or factors [40]. This idea originated in

the 1920s in England when R.A. Fisher decided to vary all factors simultaneously rather than individually, calling it factorial design [41]. A factorial design involves a fixed number of levels or values for a number of factors run in all possible combinations. The factors for the experiments can be qualitative or quantitative allowing in-depth studies to be performed. Two level factorial designs are especially useful since relatively few runs need to be performed per factor, observations are easily interpreted and additional results can easily be added to the experimental space [42]. As the number of factors increases the process of studying the effect of each factor becomes time consuming thus people like R.L. Plackett, J.P. Burman [43] and G.Taguchi have developed methods that assist in minimizing the work required in a multifactorial experiment. For example, if an experiment containing two levels and has 15 different factors, $2^{15} = 32,768$ experiments would need to be performed to have a complete set of data.

2.6.1 Plackett-Burman screening designs

Screening designs refer to experimental designs that can be applied when a large number of input factors need to be examined, identifying which have the greatest effect or are the most important to the results or responses of interest. Determining which factors play a larger role can assist in reducing the factors used in further experiments. The goal is to minimize the number of experiments whilst maximizing the information gained from those experiments. Another important feature of screening designs is of structure of information. Experimental sample space is well defined so that additional experiments can be easily added and conclusions are well defined.

R. L. Plackett and J.P. Burman developed a popular class of screening designs that allows for factors with larger or small effects on the final results to be separated. Characteristics outlined in [43] for these designs include:

- a) Each factor in a particular experimental design will have a prime number of levels, L ;
- b) The number of runs for a design must be a multiple of L^2 ;
- c) All main effects factors are estimated equally – no predictions on which factor will have the greater effect needs to be performed;
- d) All main effects factors are estimated to be independent of each other. Although interactions between factors may have some effect, the main factors do not influence one another.

A critical step in building a screening design is selecting the factors and defining levels and responses to be measured in the experiment. Selection is based on the experience of the individual or group performing the study, and is guided by the response that is desired for the experimental space [44]. More information is gained from the experiment when the experimental space is explicitly understood, and the desired outcome of the screening designs is well defined. Factor levels need to also be selected so that experiments have a good amount of variation. Too small of a factor range results in small response variation whereas too large of a range may result in identifying all factors as important. Both qualitative and quantitative results are obtainable from these types of experiments. After factors and their levels are chosen, a design matrix is generated for the Plackett-Burman screening design. Table 2-1 shows a 12-run two level design with 11 factors. Columns represent each of the factors, with the elements of the column represent the high (+1) and low (-1) levels or modalities set for each factor. Each row in the table represents a single

run. Each run leads to a result that is used for calculating responses identifying which factors are and are not main effects.

Upon calculating all the results from each run in the Plackett-Burman design table, contrasts for each factor are calculated. Contrasts are used for making comparisons between runs. The design tables are created so that the modalities in each column sum to zero, allowing contrasts to be calculated for each factor using Equation 6.

Table 2-1: Example of a 12-run Plackett-Burman design [45]

Trial	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
1	+	-	+	-	-	-	+	+	+	-	+
2	+	+	-	+	-	-	-	+	+	+	-
3	-	+	+	-	+	-	-	-	+	+	+
4	+	-	+	+	-	+	-	-	-	+	+
5	+	+	-	+	+	-	+	-	-	-	+
6	+	+	+	-	+	+	-	+	-	-	-
7	-	+	+	+	-	+	+	-	+	-	-
8	-	-	+	+	+	-	+	+	-	+	-
9	-	-	-	+	+	+	-	+	+	-	+
10	+	-	-	-	+	+	+	-	+	+	-
11	-	+	-	-	-	+	+	+	-	+	+
12	-	-	-	-	-	-	-	-	-	-	-

$$C = \sum_{j=1}^k c_j \mu_j \quad (6)$$

where c_j is the level for run j and μ_j is the results for run j . After the contrasts are found, the determination of significant factors is completed using multifactorial analyses of variance (ANOVA).

ANOVA analyzes the variation in experimental outcome in order to determine the contributions of the given factors to the variance [46]. Used in conjunction with Plackett-Burman designs ANOVA assists with distinguishing which factors are main effects and which are interactions of

main effects. Using the sum of squares for the contrasts, Equations 7 and 8, Scree plots can be developed for the sum of squares allowing a comparison of the results to be performed.

$$SS(C) = \frac{(\hat{C})^2}{\text{Adjustment term}} \quad (7)$$

$$\text{Adjustment term} = \frac{1}{n} \sum_{j=1}^k c_j^2 \quad (8)$$

Scree plots are graphical tools used for analyzing sums of squares displaying which factors have more effects on the results than others. “Scree” is defined as the rubble or loose stone lying on the slope at the base of a cliff [47], accurately describing the visual representation of this plot. Scree plot display contrast results in descending order of magnitude. The cliff peak represents any of the contrasts that have a large effect on the results, where the cliff height is representative of how large the effect is. The rubble at the base of the cliff represents contrasts that do not present a large effect on the contrasts, Figure 2-12.

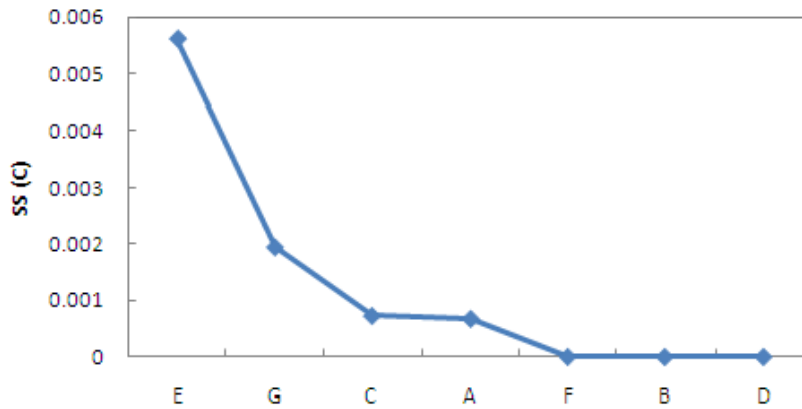


Figure 2-12: Scree plot [48]

The sum of squares is the estimation of variance for the contrasts therefore inflated results presented within the Scree plots represent contrasts that exhibit more variance and can be confirmed to have a larger effect on the results. The only way to confirm that the contrast are main effects is to perform a subsequent experiment with the same factors and different modalities. If the contrasts remain high across both Scree plots, then it is likely that it represents a main effect.

3 Evaluating Geometric Features

This section describes the procedure used for the identification of geometric features found on industrial mould surfaces and the process in which these features are remodelled for draping simulations. A remodelling tool is introduced in Section 3.2 that is used as a direct link between CAD model parameters and a database of in-plane shear results built from draping simulations performed on parameterized geometric features.

3.1 Base surfaces

Industrial moulds are made of a number of geometric features quantified through various geometric parameters that have design specific values. Lengths and radii define either tight corners, flat segments, or long, smooth arcs that make up the complex geometry of moulds and final components. These individual geometric features are labelled as base surfaces in the approach developed in this work, and they are identified as either flat, single curvature, or double curvature, Figure 3-1.

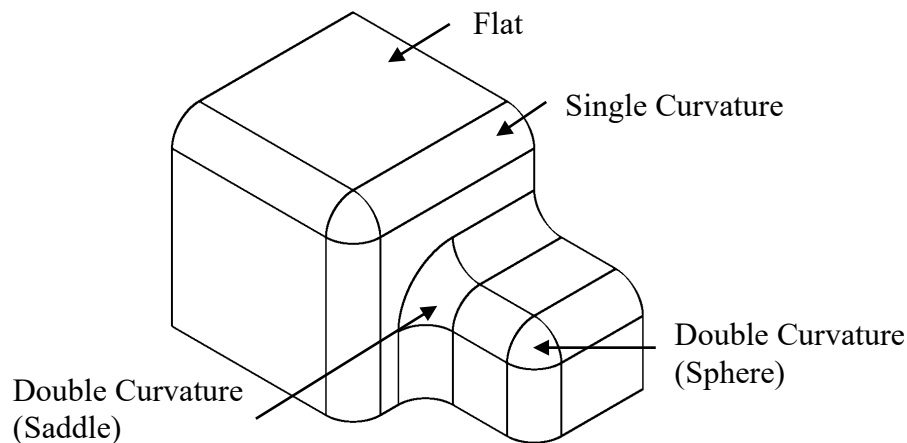


Figure 3-1: Flat, single curvature and double curvature base surfaces

Flat surfaces are defined by straight edges containing an area that resides on a single plane. At any single point on a flat surface, there is no curvature. Single curvature surfaces are defined as having curvature in a single plane at any single point on the surface. Observing a segment of single curvature between two planes connected by two straight edges, the radius of curvature can increase or decrease and is still labelled single curvature, Figure 3-2. At any point on single curvature, one direction devoid of curvature can always be identified. Finally, double curvature surfaces are defined as having curvature in two planes at any single point on the surface. Double curvature surfaces are observed in two configurations labelled as, spheroid or saddle as illustrated in Figure 3-1, the main difference being the sign or direction of the two radii that are defined at any point on that surface. In similar fashion to single curvature surfaces, double curvature surfaces can have a varying radius between two planes found on the surface, Figure 3-2.

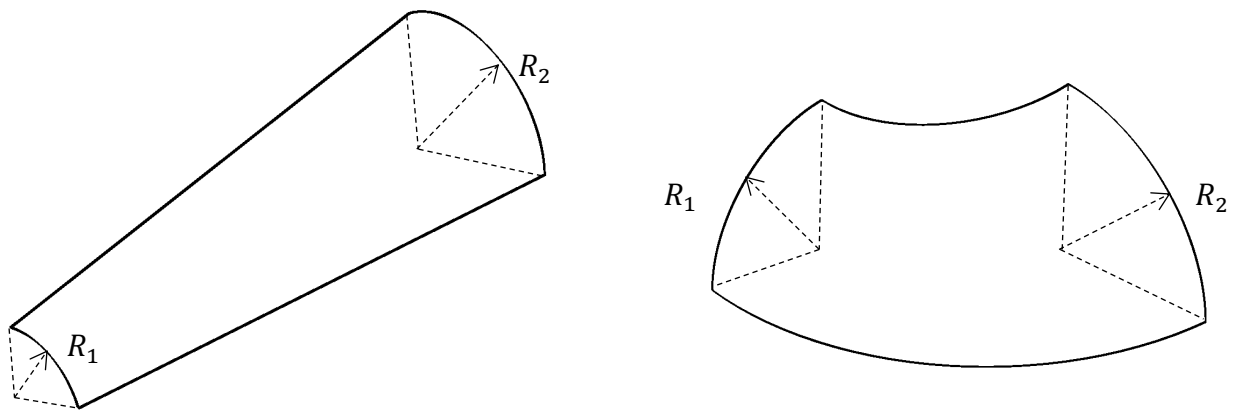


Figure 3-2: Tapered single curvature (left) and tapered double curvature (right): $R_1 < R_2$

The three base surface types identified can all have a degree of twist associated with them. Twist can be defined as a degree of angular rotation between opposing ends of an individual base surface and can occur in conjunction with each base surface type. These are defined with the prefix twisted before each base surface label, Figure 3-3.

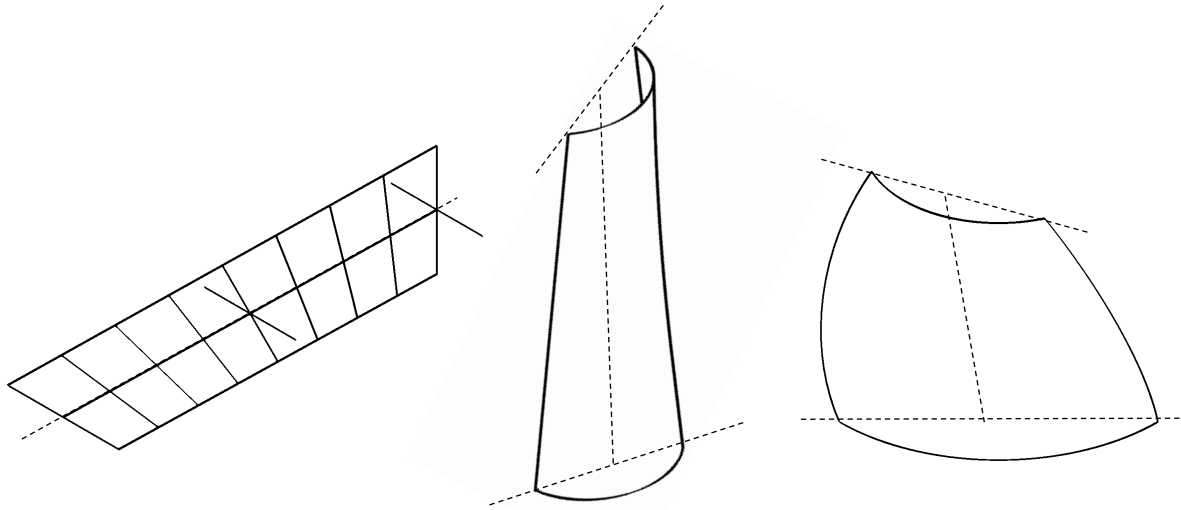


Figure 3-3: Twisted flat (left), single curvature (centre), double curvature (right) base surfaces

3.2 Defining the conical frustum

3.2.1 Introduction

Individual base surfaces control the extent of in-plane shear generated when draping a fabric. In order to determine the extent of shear in each case, draping simulations are performed on parameterized base surfaces with results compiled in a database (Database A) and used when building draping strategies. In order to build the database, models of base surfaces must be built allowing a direct link between base surface results and parameters obtained from CAD models of parts or measurable on physical moulds. In order to structure and populate the database whilst accurately building base surfaces, a multi-parameter base surface remodelling tool – the conical frustum – was devised. In an initial phase, prior to any specific cases being tackled, the conical frustum is used for modelling base surfaces, allowing draping simulations to be performed and a database populated. The parameters by which the conical frustum is defined enable base surfaces

on moulds to be directly linked to database results and allow specific base surface results to be probed.

3.2.2 12 parameter conical frustum

A breakdown of the 12 parameters used by the conical frustum is presented in this section, followed by a description of the method in which the conical frustum is used for building base geometries.

A traditional conical frustum is defined as a portion of a cone that lies between two parallel planes, removing the tip, Figure 3-4. For present use, alterations were made and a number of parameters were added so that flat, single curvature and double curvature surfaces may be generated, with and without a degree of twist.

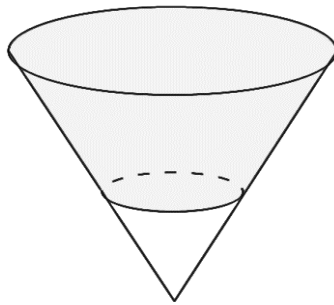


Figure 3-4: Traditional conical frustum

Rather than using parallel circles, the conical frustum can be generalized to using ellipses 1 and 2 to define two planes with centres C_1 and C_2 respectively, Figure 3-5. These ellipses can be out of alignment with one another, both in terms of the planes on which they are defined and in terms of their major and minor axes. The distance between the centres of the two ellipses is defined by $|\vec{L}|$,

where vector $\vec{L}$ always points from C_1 to C_2 . Tilt angles θ_1 and θ_2 are specified by unit vectors $\vec{N}_1$ and $\vec{N}_2$ normal to ellipses 1 and 2, which always point outward from the respective ellipses and away from $\vec{L}$, Figure 3-5. In industrial cases, this almost always causes θ_1 to be greater than θ_2 since $\vec{N}_2$ is nearly generally parallel to $\vec{L}$. $\vec{N}_1$ and $\vec{N}_2$ are defined as independent variables for each of the two ellipses with the tilt angles defined by the angle between $\vec{L}$ and $\vec{N}_1$, and $\vec{L}$ and $\vec{N}_2$, respectively.

Four points are defined on the conical frustum: points A and B located on the edge of ellipse 1, and points C and D located on the edge of ellipse 2. The major radii are defined by $|\vec{R}_1|$ and $|\vec{R}_2|$, where $\vec{R}_1$ points from C_1 to A , and $\vec{R}_2$ points from C_2 to C . Similarly, the minor radii are defined by $|\vec{r}_1|$ and $|\vec{r}_2|$, where $\vec{r}_1$ points from C_1 to B , and $\vec{r}_2$ points from C_2 to D . $\vec{R}_1$ and $\vec{r}_1$ are orthogonal, as are $\vec{R}_2$ and $\vec{r}_2$. In practice, it is decided by the whomever is performing the calculation whether a base surface has measurable elliptical radii $|\vec{R}_1|$ and $|\vec{r}_1|$. The radii of the ellipses must be known explicitly prior to proceeding with ellipses in the model; otherwise constant arcs are assumed between points A and B on ellipse 1, and between points C and D on ellipse 2. In general, these radii are not explicitly defined as the major and minor radii of their respective ellipses, but rather the radius to each point on the curved edge. This means that $|\vec{R}_1|$ is the radius to A , $|\vec{r}_1|$ is the radius to point B , $|\vec{R}_2|$ is the radius to point C and $|\vec{r}_2|$ is the radius to point D . To determine which points on the base surface are assigned A , B , C and D , A and B , and C and D , if possible, will always be assigned along arcs. Straight edges will always connect between points A and C , and B and D .

The straight edges of the cone connecting the ellipses in the classic frustum are replaced with arcs of radii C_R and C_r connecting points A and C , and B and D , respectively. Connecting radius C_R is defined on a plane including points A , C and the midpoint of $\vec{L}$, and connecting radius C_r is defined on a plane including points B , D and the midpoint of $\vec{L}$.

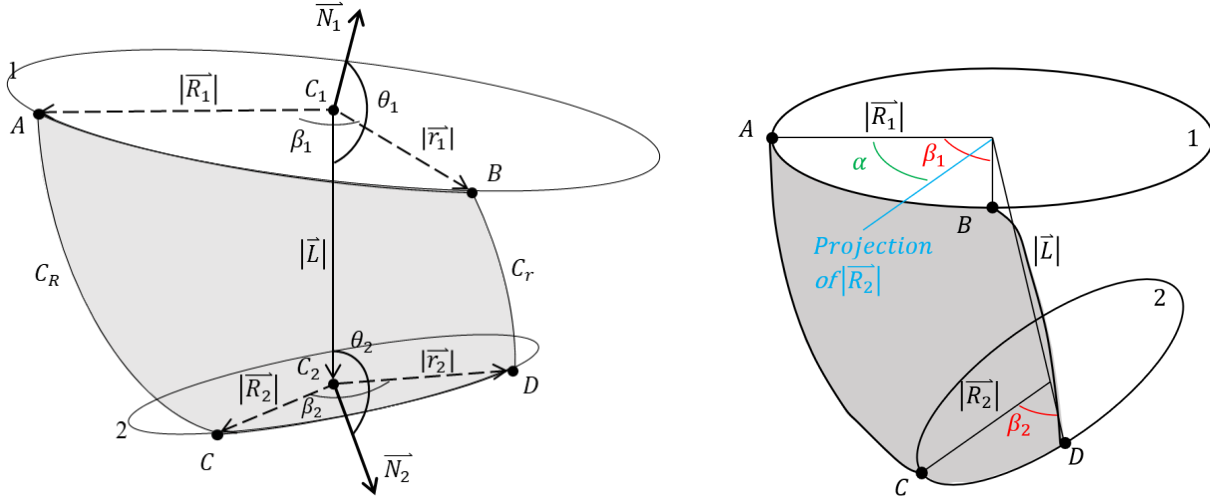


Figure 3-5: Conical frustum geometric parameters

Angles β_1 and β_2 are used for denoting the angular area of the conical frustum. If ellipses are decided upon for modelling, β_1 and β_2 are initially assumed to equal 90° and the angles are initially defined between the major and minor radii of each ellipse. These angles can be redefined and recalculated based on the portions of the ellipses used for modelling the base surfaces, if necessary. Calculations for the ellipse sections is found in Section 3.2.3. When a constant radius of an arc is assumed for connecting points A and B , on ellipse 1, and C and D on ellipse 2, β_1 and β_2 can be calculated since they are defined by the points A and B , and C and D . This calculation is presented in Section 4.2.3 as well. Finally, α is the twist angle defining the rotation of twist between the major radii on each plane.

The 12 geometric parameters of the conical frustum are summed up as:

- Radii for the points A, B, C and D for ellipses 1 and 2 $|\vec{R}_1|, |\vec{R}_2|, |\vec{r}_1|$ and $|\vec{r}_2|$
- Connecting radii C_R and C_r
- Distance between ellipses $|\vec{L}|$
- Sectioning angles β_1 and β_2
- Tilt angles θ_1 and θ_2 (defined by $\vec{N}_1$ and $\vec{N}_2$)
- Rotation angle α

3.2.3 Remodelling base surfaces using the conical frustum

The conical frustum parameters were defined in Section 4.2.2. This section outlines how each of these conical frustum parameters is calculated in order to remodel base geometries. Vectors $\vec{R}_1$, $\vec{r}_1$, $\vec{R}_2$, $\vec{r}_2$, $\vec{L}$, $\vec{N}_1$ and $\vec{N}_2$ are mainly used for calculating conical frustum parameters and base surface specific variables such as radii, lengths, angles and point positions. Other important features of base surfaces that are used in the calculations are the arc lengths labelled S_{ij} along the edges of a base surface, that extend between points i and j . Furthermore, the coordinates of the midpoints of these arcs labelled M_{ij} are also needed in the upcoming calculations, Figure 3-6.

A macro in the CAD software SolidWorks, developed by Fabien Salles at Hutchinson Aéronautique et Industrie, is capable of quickly probing a mould surface, extracting all vertex coordinates, arc lengths, and arc midpoint locations used in the calculations that follow. The macro, presented in Appendix F, always extracts A, B, C, D , all M_{ij} s and all S_{ij} s, and it is capable of extracting $|\vec{R}_1|, |\vec{r}_1|, |\vec{R}_2|, |\vec{r}_2|, C_R$, and C_r . On industrial surfaces, explicit radii may not be

available in the CAD model and therefore the radii need to be calculated in such cases. The point and arc length extraction step is crucial as all of these variables are used for calculating the conical frustum parameters.

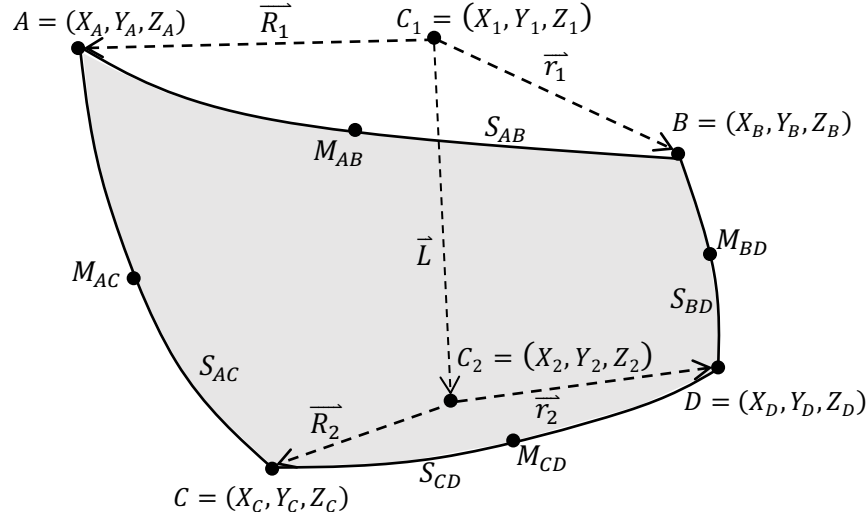


Figure 3-6: Base surface arc lengths and midpoints

Standard vector Equations 9 and 10 are used for defining the vector direction and magnitude for any vectors calculated in this section, substituting the appropriate points where necessary.

$$\vec{R}_1 = (R_{1x}, R_{1y}, R_{1z}) = (X_A - X_1, Y_A - Y_1, Z_A - Z_1) \quad (9)$$

$$|\vec{R}_1| = \sqrt{(X_A - X_1)^2 + (Y_A - Y_1)^2 + (Z_A - Z_1)^2} \quad (10)$$

where (X_A, Y_A, Z_A) and (X_B, Y_B, Z_B) are the coordinates of vertices A and B , and (X_1, Y_1, Z_1) are the coordinates of centre point C_1 . The coordinates of centre point C_2 are identified by substituting vertex coordinates for points A and B , with vertex coordinates for points C and D and by using $|\vec{R}_1|$ and $|\vec{r}_1|$. At this stage the locations of the centre points are unknown and need to be

calculated. Two methods are outlined in section 3.2.3.1 of this thesis for calculating these centre points.

From a CAD model or from a physical mould, vertex coordinates for A , B , C and D are extracted and translated into a local coordinate system where D is the origin, prior to any other calculations. This is done by taking the coordinates for the other points and subtracting the location of point D , effectively translating point D to the origin and the other points to their relative distances from the new origin without changing the surface orientation, Figure 3-7. Moving into a local coordinate system assists in simplifying the calculations. Figure 3-7 labels these new points as A' , B' , C' and D' only to show that the translation occurs; however, for the remainder of calculation development the points are still referred to by their original labels, knowing that they have been translated to a different position.

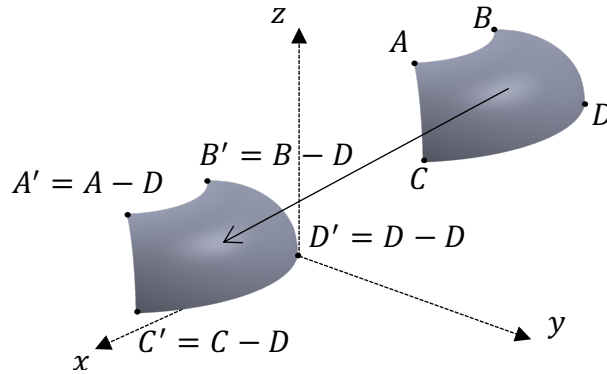


Figure 3-7: Base surface coordinate system translation

3.2.3.1 Calculating centre points C_1 and C_2

This section presents two methods for calculating centre points C_1 and C_2 and sectioning angles β_1 and β_2 . The remaining conical frustum parameters $|\vec{L}|$, θ_1 , θ_2 and α are all calculated in the

same manner following either Method 1 or Method 2. Centre point calculations constitute the groundwork in the process in which β_1 and β_2 are calculated.

Method 1 uses an assumption that the arc between two points, S_{AB} for example, has a constant radius, therefore setting $|\vec{R_1}| = |\vec{r_1}|$. Method 2 uses an assumption that the arc is defined by an ellipse where the maximum and minimum radii for that ellipse are known.

Method 1: Constant arc radius

As mentioned above, method 1 assumes a constant radius for any arc between two conical frustum points, Figure 3-8. This assumption sets $|\vec{R_1}| = |\vec{r_1}|$ for ellipse 1, and $|\vec{R_2}| = |\vec{r_2}|$ for ellipse 2, effectively making them circles. In order to remain consistent in this text, the ellipses will preserve their ellipse label. A number of arc specific variables are calculated so that the radius of the arc can be calculated. First, vector $\vec{AB}$ pointing from A to B along with its magnitude $|\vec{AB}|$ are calculated using Equations 9 and 10. In a similar way, vector $\vec{MB}$ pointing from the end point of the arc B to the midpoint of the arc M_{AB} , along with its magnitude $|\vec{MB}|$ are calculated. Using Equation 11, distance $|\vec{MW_{AB}}|$ from M_{AB} to the midpoint of $\vec{AB}$ labelled W_{AB} can be calculated. For a uniform arc radius, the intersection of $|\vec{MW_{AB}}|$ with W_{AB} will be perpendicular, allowing the remaining calculations in this method to yield a single centre point for the solution. Finally, the radius of the arc is calculated using Equation 12. This process is then repeated for each edge of the base surface interchanging the respective points, arc lengths and arc midpoints, calculating $|\vec{R_1}|$, $|\vec{R_2}|$, C_R and C_r . Since a constant arc is assumed at the start of method 1, $|\vec{R_1}| = |\vec{r_1}|$ and $|\vec{R_2}| = |\vec{r_2}|$.

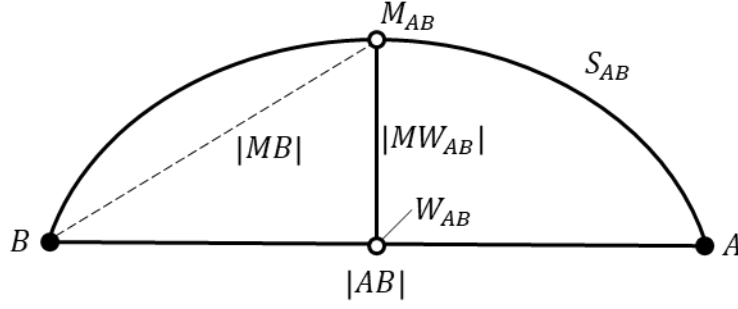


Figure 3-8: Arc radius

$$|MW_{AB}| = \sqrt{|MB|^2 - \left(\frac{|AB|}{2}\right)^2} \quad (11)$$

$$|\vec{R}_1| = \frac{|\vec{MW}_{AB}|}{2} + \frac{|\vec{AB}|^2}{8|MW_{AB}|} \quad (12)$$

Now that the radii for the arcs are known, the centre points C_1 and C_2 can be calculated. The calculation for the centre points is typically performed with a portion of MATLAB Script 1, Figure 3-9. Script 1, found in Appendix E, performs all the calculations required for determining $|\vec{R}_1|$ and $|\vec{R}_2|$; however, it is especially needed for the calculation of the centre points. The script uses three sets of Equation 10, the vector magnitude equation, pointing from the centre point to each point on the arc. Since the coordinates for each point are known, and the radius of the arc is known, MATLAB can solve the three equations simultaneous, producing a uniquely defined centre point of the arc. If radii $|\vec{R}_1|$, $|\vec{r}_1|$, $|\vec{R}_2|$, $|\vec{r}_2|$, C_R , C_r are extracted using the SolidWorks macro the radius calculations do not need to be performed; only the centre point calculation portion of Script 1 are needed.

```

%Points on Arc
A = [43.5;26.18;18.93];
M_AB = [44.08;25.6;17.01];
B = [46.36;25.78;15.88];
R = 5;

%Calculate centre of arc
syms x1 y1 z1
eq1 = (A(1) - C1(1))^2 + (A(2) - C1(2))^2 + (A(3) - C1(3))^2 - magR1^2
eq2 = (B(1) - C1(1))^2 + (B(2) - C1(2))^2 + (B(3) - C1(3))^2 - magR1^2
eq3 = (M_AB(1) - C1(1))^2 + (M_AB(2) - C1(2))^2 + (M_AB(3) - C1(3))^2 - magR1^2
sol1 = solve(eq1,eq2,eq3);

x1 = round( sol1.X1 );
y1 = round ( sol1.Y1 );
z1 = round ( sol1.Z1);

```

Figure 3-9: Portion of MATLAB Script 1 for calculating centre points (Method 1)

Since arc lengths are extracted from the base surface, section angles β_1 and β_2 can be calculated using Equation 13.

$$\beta_1 = 360 \frac{S_{AB}}{2\pi |\overline{R_1}|} \quad (13)$$

In practice Method 1 is almost always used except on base surfaces that are explicitly defined using ellipses.

Method 2: Elliptical arc radius

For a base surface that is defined by elliptical arcs, $|\overline{R_1}|$ and $|\overline{R_2}|$, and $|\overline{r_1}|$ and $|\overline{r_2}|$ must be known prior to proceeding. For these cases they may be the major and minor radii for each ellipse however, calculations for section angles β_1 and β_2 will confirm this. For ellipse 1, points A , B and M_{AB} would be extracted using the same macro in SolidWorks. Since the distance from centre, C_1 to M_{AB} is unknown, an alternative approach is used for calculating C_1 .

To begin, vectors $\overrightarrow{AM_{AB}}$ and $\overrightarrow{AB}$, pointing from A to M_{AB} and from A to B respectively, are calculated using Equation 9, Figure 3-10. These vectors will be used for defining the plane in which points A , B and M_{AB} are on. To determine the equation for a plane, Equation 14, the vector $\overrightarrow{N_1}$ normal to the plane needs to be calculated. This is achieved taking the cross product, Equation 15, of vectors $\overrightarrow{AM_{AB}}$ and $\overrightarrow{AB}$:

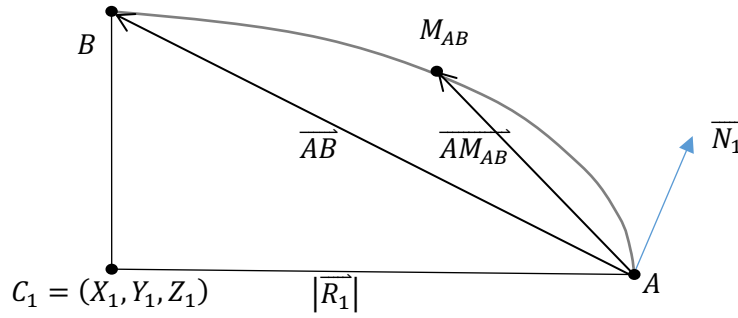


Figure 3-10: Plane vectors and normal vector

$$a(x - x_0) + b(y - y_0) + c(z - z_0) = 0 \quad (14)$$

where a , b and c are the respective x , y and z components of the normal vector $\overrightarrow{N_1} = (N_{1x}, N_{1y}, N_{1z})$, x , y and z are the coordinates of any point in the plane, and x_0 , y_0 and z_0 are the coordinates of any known point in the plane, in this case either A , B or M_{AB} .

$$\overrightarrow{N_1} = \overrightarrow{AM_{AB}} \times \overrightarrow{AB} = (N_{1x}, N_{1y}, N_{1z}) \quad (15)$$

$$N_{1x} = (AM_{AB,y} * AB_z - AM_{AB,z} * AB_y) \quad (7a)$$

$$N_{1y} = (AM_{AB,z} * AB_x - AM_{AB,x} * AB_z) \quad (7b)$$

$$N_{1z} = (AM_{AB,x} * AB_y - AM_{AB,y} * AB_x) \quad (7c)$$

Once $\overline{N_1}$ is known, the equation for the plane can be finalized by inputting $(N_{1x}, N_{1y}, N_{1z}) = (a, b, c)$, and $(x_0, y_0, z_0) = (A_x, A_y, A_z)$. If either of the other points are used for (x_0, y_0, z_0) , the same equation for the plane is defined.

Now that the plane equation is known MATLAB Script 2 found in Appendix E calculates the centre point. Similar to Script 1, Script 2 calculates all the conical frustum parameters. It uses three equations to solve for the three coordinates of C_1 . Vector magnitudes for points A and B , Equation 10, are used since $|\overline{R_1}|$ and $|\overline{r_1}|$ are known. The equation for the plane is defined by $\overline{N_1}$ and A . Centre point C_1 coordinates (X_1, Y_1, Z_1) are substituted into the plane equation as (x, y, z) . The plane equation ensures that the solution produces points that lie on the plane while the vector magnitude equations ensure that the point is located at the centre of the ellipse.

Script 2 initially yields two possible solutions for the centre point; one solution is the actual centre of the ellipse whilst the other is the centre of another ellipse with the same major and minor radii. Solution 1 and solution 2, S_1 and S_2 respectively, are observed on the same plane with the same points A and B , Figure 3-12. Drawing a line between points A and B shows that solutions 1 and 2 are mirrored about this line, validating the possibility of two solutions. To determine which solution is correct, known point M_{AB} defined on the desired ellipse is used. For the correct solution, the straight line distance from actual centre point C_1 to midpoint M_{AB} , labelled $|\overline{C_m}|$, will always be greater than the distance from the mirrored solution centre point S_2 labelled $|\overline{S_m}|$, Figure 3-12. Distances $|\overline{R_m}|$ and $|\overline{S_m}|$ are calculated in Script 2 using the vector magnitudes, Equation 10. The mirror line causes the mirrored centre point/solution to move closer to midpoint of the actual ellipse. Script 2 performs a check, comparing the distances from both solutions and chooses the centre point with the greater distance to M_{AB} . This entire process is repeated for ellipse 2.

```

syms X1 Y1 Z1

A = [-1;11;3];
M_AB = [0.136;10.691;1.659];
B = [0;10;0];

C1 = [X1;Y1;Z1]

magR1 = sqrt(6);
magr1 = sqrt(5);

R1 = A - C1;
r1 = B - C1;
vM = M_AB - C1;

AM = M_AB - A
AB = B - A

BM = M_AB - B
BA = A - B

N = cross (AM,AB)

eq1 = N(1)*(X1-A(1)) + N(2)*(Y1-A(2)) + N(3)*(Z1-A(3))
eq2 = R1(1)^2 + R1(2)^2 + R1(3)^2 - magR1^2;
eq3 = r1(1)^2 + r1(2)^2 + r1(3)^2 - magr1^2;

sol1 = solve(eq1,eq2,eq3);
x1 = (sol1.X1)
y1 = (sol1.Y1)
z1 = (sol1.Z1)

%Check Location of Mab

magMAB1 = sqrt((M_AB(1)- x1(1))^2 + (M_AB(2)- y1(1))^2 + (M_AB(3)-z1(1))^2);
magMAB2 = sqrt((M_AB(1)- x1(2))^2 + (M_AB(2)- y1(2))^2 + (M_AB(3)-z1(2))^2);

if gt(magMAB1,magMAB2)
    C1 = [x1(1);y1(1);z1(1)]
end

```

Figure 3-11: MATLAB Script 2: Portion for calculating centre point (Method 2)

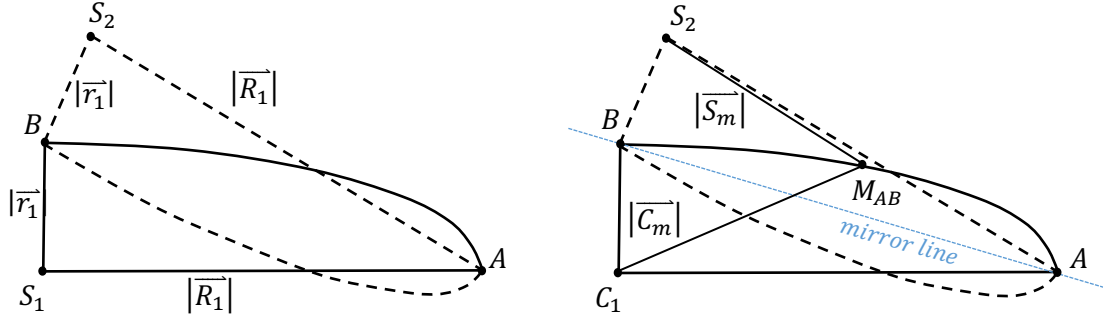


Figure 3-12: Script 2 solutions (left); correct solution identification (right)

3.2.4 Calculating the remaining conical frustum parameters

The main difference between method 1 and method 2 for calculating the centre point of ellipses is that in method 1, no radii are extracted for the base surface - only arc lengths. This leads to the mandatory assumption that the arc that passes through the three points is of constant radius and calculations for the centre point can proceed. If the radii $|\vec{R}_1|$, $|\vec{r}_1|$, $|\vec{R}_2|$ and $|\vec{r}_2|$ are all known, Method 2 is used.

Once the centre points are known for each ellipse, calculations for section angles β_1 and β_2 can be calculated. As mentioned in Section 4.2.2, the radii $|\vec{R}_1|$, $|\vec{r}_1|$, $|\vec{R}_2|$ and $|\vec{r}_2|$ as the radius to points A, B, C and D respectively. Values of β_1 and β_2 will confirm, for an elliptical curvature, if $|\vec{R}_1|$ and $|\vec{r}_1|$, or $|\vec{R}_2|$ and $|\vec{r}_2|$ are orthogonal and will be equal to 90° . β_1 and β_2 are calculated using Equations 16 and 17, after applying either method 1 or method 2 to calculate the centre points C_1 and C_2 .

$$\beta_1 = \cos^{-1} \left(\frac{\vec{R}_1 \cdot \vec{r}_1}{|\vec{R}_1| |\vec{r}_1|} \right) \quad (16)$$

$$\vec{R}_1 \cdot \vec{r}_1 = R_{1x} * r_{1x} + R_{1y} * r_{1y} + R_{1z} * r_{1z} \quad (17)$$

Normal vectors $\vec{N}_1$ and $\vec{N}_2$ are needed for calculating tilt angles θ_1 and θ_2 and twist angle α . If Method 2 was used for calculating the centre points, these vectors are already available. If Method 1 was used, $\vec{N}_1$ and $\vec{N}_2$ are calculated using the cross product, Equation 15, substituting vectors $\vec{AM}_{AB}$ and $\vec{AB}$ for $\vec{R}_1$ and $\vec{r}_1$, or $\vec{R}_2$ and $\vec{r}_2$. The reader may recall that these normal vectors point out from the surface and therefore, $\vec{N}_2 = \vec{R}_1 \times \vec{r}_1$, $\vec{N}_2 = \vec{r}_2 \times \vec{R}_2$. Once $\vec{N}_1$ and $\vec{N}_2$ are calculated, Equation 2 is used for calculating θ_1 and θ_2 with $\vec{N}_1$ and $\vec{L}$ for ellipse 1, and $\vec{N}_2$ and $\vec{L}$ for ellipse 2. $\vec{N}_1$ and $\vec{N}_2$ are explicitly defined for the planes of the base surface, whilst θ_1 and θ_2 are defined using these normal vectors. Finally, the twist angle α is calculated using a set of extra vectors defined with respect to $\vec{L}$. These vectors are found by taking the cross products $\vec{R}_1 \times \vec{L}$ and $\vec{R}_2 \times \vec{L}$, Equation 15. Angle α is defined between these two vectors, and is calculated using Equations 16 and 17. Since these vectors are both perpendicular to $\vec{L}$ as result of the cross products. a comparable relative rotation angle between these two vectors is directly linked to the rotation angle about $\vec{L}$ between $\vec{R}_1$ and $\vec{R}_2$, Figure 3-13.

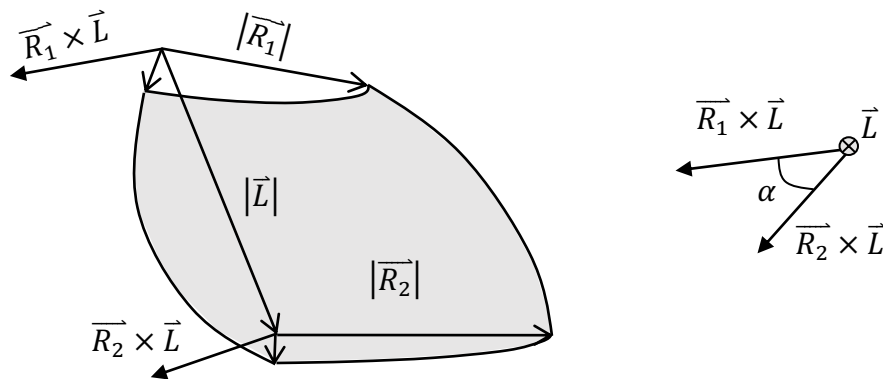


Figure 3-13: Rotation angle α

3.2.5 Calculating the conical frustum parameters using Method 1

The following is an example for how the conical frustum parameters are calculated using Method 1 for a base surface, Figure 3-14. Table 3-1 lists the vertex coordinates and arc lengths that have been extracted from a CAD model for this base surface using the SolidWorks macro found in Appendix F. Points appear twice on the table to represent what is extracted after the macro is run. Since points are extracted along each edge and point A , for example, is shared by two edges, it appears twice in the table. For this base surface, $|\vec{R}_1|$, $|\vec{r}_1|$, $|\vec{R}_2|$, $|\vec{r}_2|$, C_R and C_r were not extracted, therefore a constant radius is assumed. This assumption sets $|\vec{R}_1| = |\vec{r}_1|$ and $|\vec{R}_2| = |\vec{r}_2|$. All conical frustum parameters are calculated by running MATLAB Script 1. The full script used for this example can be found in Appendix E.

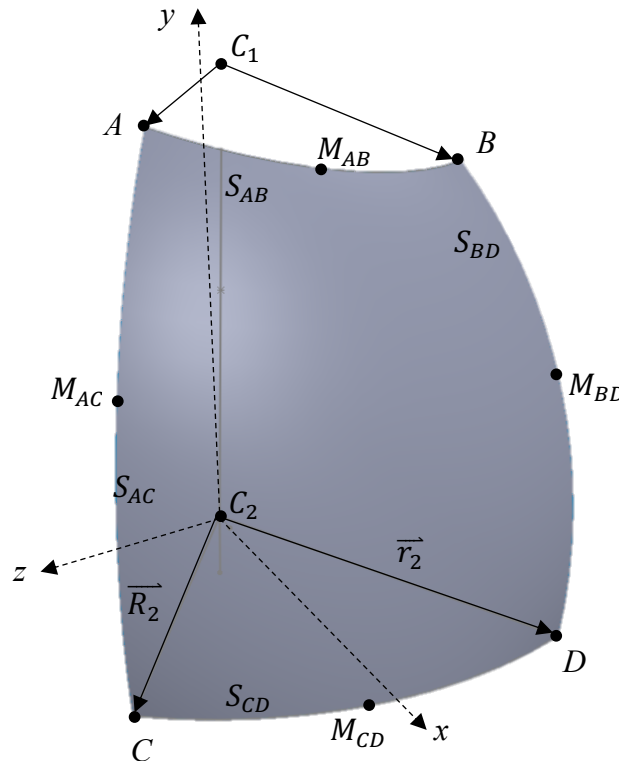


Figure 3-14: Base surface for example 1

The first step of Method 1 is to determine the radii for each of the edges. All the extracted points are first translated into a coordinate system where D is at the origin. The new points are found in Table 3-2. Next, vectors $\overrightarrow{AB}$, $\overrightarrow{BD}$, $\overrightarrow{CD}$, and $\overrightarrow{AC}$ and their magnitudes, $|\overrightarrow{AB}|$, $|\overrightarrow{BD}|$, $|\overrightarrow{CD}|$, and $|\overrightarrow{AC}|$ between all the conical frustum vertices are calculated using Equations 9 and 10. For each edge, Equations 11 and 12 are used for calculating each radius along their respective base surface edge; $|\overrightarrow{R_1}|$ for edge AB , C_r for edge BD , $|\overrightarrow{R_2}|$ for edge CD and C_R for edge AC .

Table 3-1: Example vertices and parameters for Method 1

Extracted Parameter	Coordinates			Conical Frustum Vertex Labels
	x	y	z	
Vertex 1	3	10	-5	B
Vertex 2	4.468	11.4	-0.987	M_{AB}
Vertex 3	2.823	12.608	3.019	A
Vertex 4	8.192	0	5.736	C
Vertex 5	7.096	7.253	5.643	M_{AC}
Vertex 6	2.823	12.608	3.019	A
Vertex 7	8.192	0	5.736	C
Vertex 8	9.99	0	0.436	M_{CD}
Vertex 9	8.66	0	-5	D
Vertex 10	8.66	0	-5	D
Vertex 11	6.83	5.566	-5.908	M_{BD}
Vertex 12	3	10	-5	B
S_{AB}	9.184			
S_{AC}	14.911			
S_{CD}	11.345			
S_{BD}	11.982			

Now that the radii are known, centre points C_1 and C_2 are calculated: $C_1 = (-11, 11, 4)$ and $C_2 = (-9, 0, 5)$. β_1 and β_2 are calculated using the arc lengths extracted from the surface and Equation 13, $\beta_1 = 81^\circ$ and $\beta_2 = 65^\circ$. $\vec{L}$ is calculated between the centre points: $\vec{L} = (2, -11, 1)$. Normal vectors

$\overrightarrow{N_1}$ and $\overrightarrow{N_2}$ are obtained using the cross product, Equation 15, between $\overrightarrow{R_1}$ and $\overrightarrow{r_1}$ and between $\overrightarrow{r_2}$ and $\overrightarrow{R_2}$, respectively. $\overrightarrow{N_1}$ and $\overrightarrow{N_2}$ are used for calculating tilt angles θ_1 and θ_2 using Equations 16 and 17. Rotation angle α is calculated using Equations 15, 16, and 17. $\theta_1 = 165^\circ$, $\theta_2 = 11^\circ$, and $\alpha = 4^\circ$. Table 3-3 lists all the conical frustum parameters for this base surface.

Table 3-2: Translated coordinates for example 1

Translated Coordinates			Conical Frustum Vertex Labels
x	y	z	
-5.66	10	0	B
-4.192	11.4	4.013	M_{AB}
-5.837	12.608	8.019	A
-0.468	0	10.736	C
-1.564	7.253	10.643	M_{AC}
-5.837	12.608	8.019	A
-0.468	0	10.736	C
1.33	0	5.436	M_{CD}
0	0	0	D
0	0	0	D
-1.83	5.566	-0.908	M_{BD}
-5.66	10	0	B

Table 3-3: Conical frustum parameters for example 1

Conical Frustum Parameter	Value
$ \overrightarrow{R_1} $	6.48 mm
$ \overrightarrow{R_1} $	6.48 mm
$ \overrightarrow{R_1} $	10.01 mm
$ \overrightarrow{R_1} $	10.01 mm
C_R	12 mm
C_R	12 mm
$ \overrightarrow{L} $	11.22 mm
β_1	81°
β_2	65°
θ_1	165°
θ_2	11°
α	4°

3.2.6 Calculating the conical frustum parameters using Method 2

The following is an example for how the conical frustum parameters are calculated using Method 2 for a base surface, Figure 3-15. Table 3-4 lists the vertex coordinates, arc midpoints and radii that have been extracted from this particular base surface. Data extraction typically includes arc length. Since Method 2 does not use arc length, they are not listed. All conical frustum parameters were calculated by running MATLAB Script 2. The full script used for this example can be found in Appendix E.

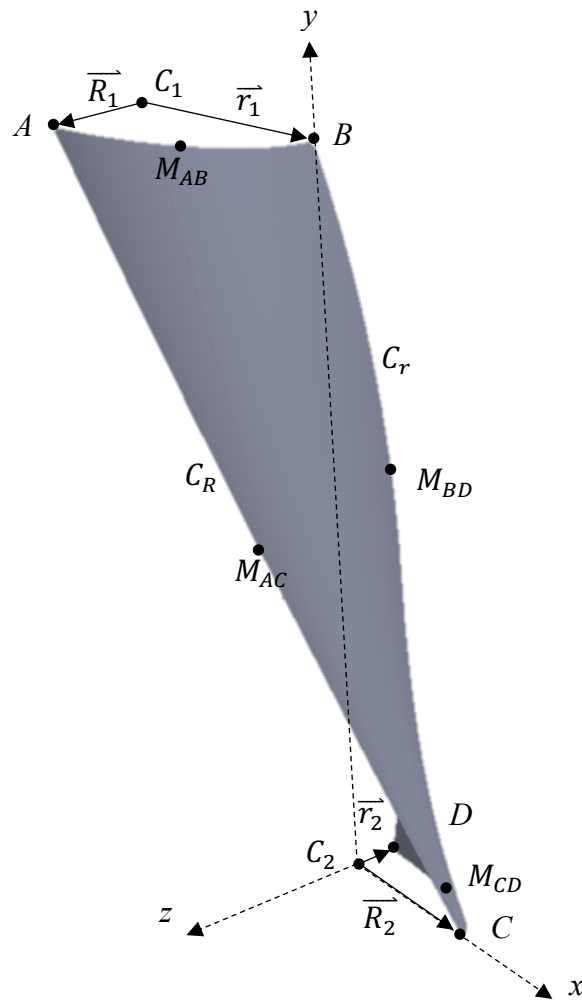


Figure 3-15: Base surface for example 2

Table 3-4: Example vertices and parameters for Method 2

Extracted Parameter	Coordinates			Conical Frustum Vertex Labels
	x	y	z	
Vertex 1	0	0	-0.5	D
Vertex 2	1.069	0	-0.423	M_{CD}
Vertex 3	2	0	0	C
Vertex 4	0	0	-0.5	D
Vertex 5	0.405	5.015	-0.553	M_{BD}
Vertex 6	0	10	0	B
Vertex 7	0	10	0	B
Vertex 8	0.136	10.691	1.659	M_{AB}
Vertex 9	-1	11	3	A
Vertex 10	2	0	0	C
Vertex 11	1.108	5.57	1.85	M_{AC}
Vertex 12	-1	11	3	A
$ \vec{R}_1 $	2.45 cm			
$ \vec{r}_1 $	2.24 cm			
$ \vec{R}_2 $	2 cm			
$ \vec{r}_2 $	0.5 cm			
C_R	25 cm			
C_r	25 cm			

Both planes are defined by elliptical curves by the user in this example, meaning that β_1 and β_2 are 90° , which will be confirmed later, and the major and minor radii for both ellipses are explicitly known. Vertices are labelled with the appropriate conical frustum labels A , B , C , D , M_{AB} , M_{BD} , M_{CD} , and M_{AC} . This is achieved by ensuring that the largest arc length or straight length distance between vertices is associated with $|\vec{L}|$. This sets vertices 1 and 3 as A and B , and 2 and 4 as C and D . Next, the vertices are translated to a local coordinate system where vertex D is the origin. The new coordinates are in Table 3-5.

Table 3-5: Translated coordaintes for example 2

Translated Coordinates			Conical Frustum Vertex Labels
x	y	z	
0	0	0	D
1.069	0	-0.423	M_{CD}
2	0	0	C
0	0	-0.5	D
0.405	5.015	-0.553	M_{BD}
0	10	0	B
0	10	0	B
0.136	10.691	1.659	M_{AB}
-1	11	3	A
2	0	0	C
1.108	5.57	1.85	M_{AC}
-1	11	3	A

Table 3-6: Conical frustum parameters for example 2

Conical Frustum Parameter	Value
$ \vec{R}_1 $	2.45 mm
$ \vec{R}_1 $	2.24 mm
$ \vec{R}_1 $	2 mm
$ \vec{R}_1 $	0.5 mm
C_R	25 mm
C_R	25 mm
$ \vec{L} $	10.18 mm
β_1	90°
β_2	90°
θ_1	152°
θ_2	13°
α	57°

The values for A , B , M_{AB} and C , D , M_{CD} are used in MATLAB Script 2 for calculating the centre points of the two ellipses: $C_1 = (-1.98, 9.93, 1.53)$ and $C_2 = (0, 0, 0.5)$. During this stage, normal vectors $\vec{N}_1$ and $\vec{N}_2$ are calculated to define the plane equations. Once the centres of both ellipses are known, vectors $\vec{R}_1$, $\vec{r}_1$, $\vec{R}_2$, $\vec{r}_2$ and $\vec{L}$ are calculated using Equation 9: $\vec{R}_1 = (1, 0.5, 2.5)$, $\vec{r}_1 = (2, -0.5, -0.5)$, $\vec{R}_2 = (2, 0, 0)$, $\vec{r}_2 = (0, 0, -0.5)$ and $\vec{L} = (2, -10.5, -0.5)$. Normal vectors $\vec{N}_1$ and $\vec{N}_2$ are used for calculating tilt angles θ_1 and θ_2 using Equations 16 and 17. Rotation angle α is calculated using Equations 15, 16, and 17. $\theta_1 = 152^\circ$, $\theta_2 = 13^\circ$, and $\alpha = 57^\circ$. β_1 and β_2 are checked using Equations 16 and 17, and confirmed to be equal to 90° . Table 3-6 lists all the conical frustum parameters for this base surface.

3.2.6.1 Sample calculations performed by Script 2 from example 2

For calculating the equation of the plane that A , B , M_{AB} lay on:

$$\overrightarrow{AB} = (AB_x, AB_y, AB_z) = (B_x - A_x, B_y - A_y, B_z - A_z)$$

$$\overrightarrow{AB} = (1, -1, -3)$$

$$\overrightarrow{AM_{AB}} = (1, -0.31, -1.34)$$

$$\overrightarrow{N_1} = \overrightarrow{AM_{AB}} \times \overrightarrow{AB} = (N_{1x}, N_{1y}, N_{1z})$$

$$N_{1x} = (AM_{AB,y} * AB_z - AM_{AB,z} * AB_y) = -0.41$$

$$N_{1y} = (AM_{AB,z} * AB_x - AM_{AB,x} * AB_z) = 1.66$$

$$N_{1z} = (AM_{AB,x} * AB_y - AM_{AB,y} * AB_x) = -0.69$$

Plane Equation

$$N_{1x}(x - A_x) + N_{1y}(y - A_y) + N_{1z}(z - A_z) = 0$$

$$-0.41x + 1.66y - 0.69z - 16.25 = 0$$

Solve uses following three equations and produces two solutions, S_1 and S_2 , for C_1

$$|\overrightarrow{R_1}| = \left((A_x - X_1)^2 + (A_y - Y_1)^2 + (A_z - Z_1)^2 \right)^{1/2}$$

$$|\overrightarrow{r_1}| = \left((B_x - X_1)^2 + (B_y - Y_1)^2 + (B_z - Z_1)^2 \right)^{1/2}$$

$$-0.41X_1 + 1.66Y_1 - 0.69Z_1 - 16.25 = 0$$

$$S_1 = (-1.98, 9.93, 1.53)$$

$$S_2 = (1.08, 10.97, 2.19)$$

Script compares the distances to M_{AB} using vector equations;

$$|\overrightarrow{S_1 M_{AB}}| = 2.22$$

$$|\overrightarrow{S_1 M_{AB}}| = 1.11$$

$$|\overrightarrow{S_1 M_{AB}}| > |\overrightarrow{S_1 M_{AB}}|, \therefore S_1 = C_1$$

Section angle β_1 is calculated using $\overrightarrow{R_1}$ and $\overrightarrow{r_1}$ which can be calculated since C_1 is known;

$$\beta_1 = \cos^{-1} \left(\frac{\overrightarrow{R_1} \cdot \overrightarrow{r_1}}{|\overrightarrow{R_1}| |\overrightarrow{r_1}|} \right) = 90^\circ$$

$$\overrightarrow{R_1} \cdot \overrightarrow{r_1} = R_{1x} * r_{1x} + R_{1y} * r_{1y} + R_{1z} * r_{1z}$$

Tilt angle θ_1 is calculated, after C_2 is calculated therefore $\vec{L}$ is known:

$$\theta_1 = \cos^{-1} \left(\frac{\overrightarrow{N_1} \cdot \vec{L}}{|\overrightarrow{N_1}| |\vec{L}|} \right)$$

$$\overrightarrow{N_1} \cdot \vec{L} = N_{1x} * L_x + N_{1y} * L_y + N_{1z} * L_z$$

$$\theta_1 = 152.63^\circ$$

Calculating twist angle α :

$$\alpha = \cos^{-1} \left(\frac{(\overrightarrow{R_1} \times \vec{L}) \cdot (\overrightarrow{R_2} \times \vec{L})}{|(\overrightarrow{R_1} \times \vec{L})| |(\overrightarrow{R_2} \times \vec{L})|} \right) = 66.06^\circ$$

4 Hierarchy of Parameters

This section outlines the tools used and steps taken for breaking down two different moulds into base surfaces, so that conical frustum parameters may be calculated. Mould decomposition can be performed on physical moulds using a number of measuring tools, or on CAD models of the moulds using SolidWorks or similar CAD software.

4.1 Mould decomposition

A first mould decomposition exercise was performed on physical moulds. Two moulds shown in Figure 4-1 were decomposed into their base surfaces allowing the conical frustum geometric parameters to be calculated for all of their base surfaces. These first sets of geometric parameters assisted in identifying which of the 12 conical frustum parameters vary most frequently. The first mould, labelled *FR*, is a simple shape with mainly flat and single curvature surfaces. The second mould, labelled *BD100*, provides a more complex surface highlighting more doubly curved features towards the base. For both of these moulds, measurements on the physical moulds were performed using a MicroScribe contact coordinate measurement machines (CMM) along with radius gauges.



Figure 4-1: *FR* (left) and *BD100* (right) moulds

4.2 Measurement tools

Measurements for the moulds were taken using metal radius gauges for radii less than 5 mm, a digital *INSIZE* radius gauge for radii 5 – 700 mm (Figure 4-2), as well as a MicroScribe G2 CMM for measuring points on the surface (Figure 4-4).

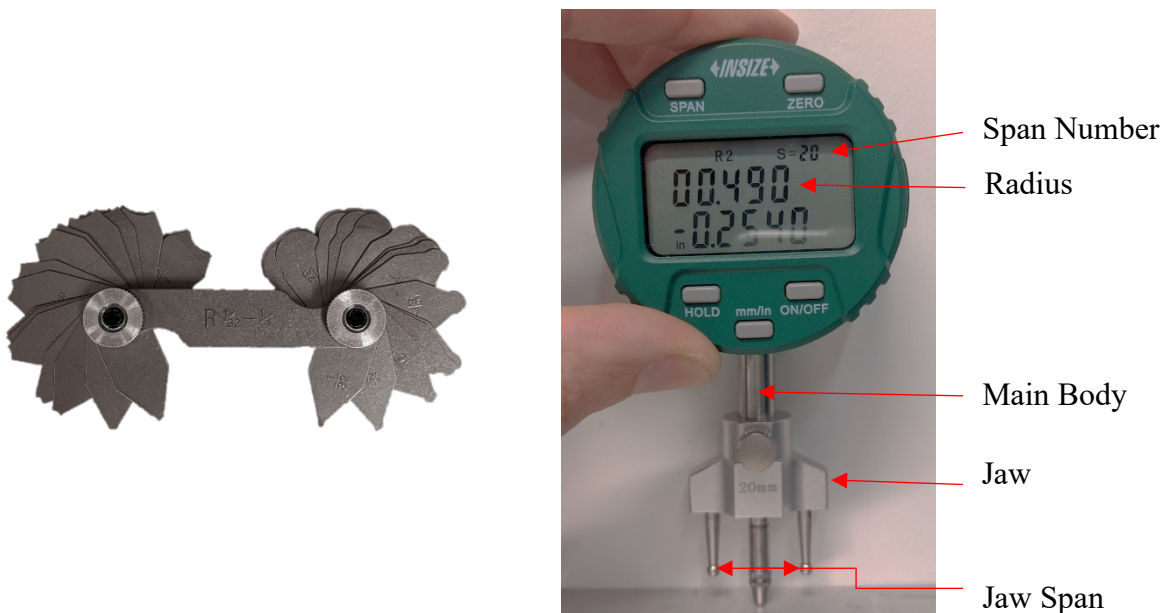


Figure 4-2: Metallic and *INSIZE* digital radius gauges

All radius gauges can be used for internal or external radii. The digital radius gauge is used by attaching the desired jaw onto the main body and adjusting the span number to the corresponding jaw span. Before each use, the radius gauge must be zeroed on a flat surface, Figure 4-3. Once a zero is set, the desired radius is measured by placing the main body in the centre of the arc and pushing the jaw pins onto the surface. Measurements were made at three different locations along the same curve.



Figure 4-3: Zeroing radius gauge (left) and radius measurement (right)

The MicroScribe G2 3-D digitizer is a precision contact-based desktop coordinates measuring machine used for capturing data points off physical objects. Instructions for using the MicroScribe are available in Imane El Karafi's undergraduate thesis [49]. The G2 model has a maximum reach of 1270 mm and a point accuracy to 0.38 mm. It allows for individual points to be selected on a surface relative to an origin found within the stylus holder or set by the user. To set a new origin, first a point is chosen on the mould surface. Next a plane of reference is set, by choosing first a point in the desired x -direction followed by a point in the desired y -direction. There is no limit to the distance required for defining the direction, but accuracy is increased by selecting points that

are further away from the designated origin. After a xy plane is defined, the MicroScribe sets the normal to this plane as the positive z direction. Since the conical frustum uses vertex coordinates as a basis for most calculations, this tool is incredibly useful in the rare cases where a physical mould is available without a CAD model. The coordinates of individual points were extracted from each mould using the MicroScribe Utility software, allowing a list to be generated in a Microsoft Excel spreadsheet.

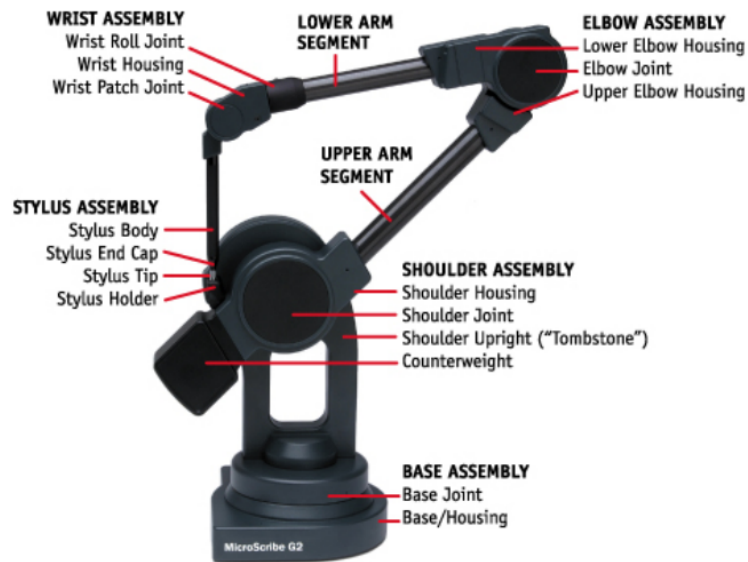


Figure 4-4: MicroScribe G2 component diagram [50]

4.3 *FR* mould decomposition

The first mould decomposed into base surfaces was the *FR* mould. As stated above, this mould's surface featured a limited number of simpler base surfaces, mainly flat and single curvature. First, individual base surfaces were identified and labelled as they were encountered. A total of 13 distinct base surfaces were identified: five flat surfaces, six single curvature surfaces and two double curvature surfaces, Figure 4-5. Next, the MicroScribe CMM was used for extracting vertex coordinates for 23 independent points, setting point 1 as the origin. Following point extraction,

radii and length measurements were taken for all base surfaces using a ruler and a digital radius gauge. Arc lengths are not measured during this process.

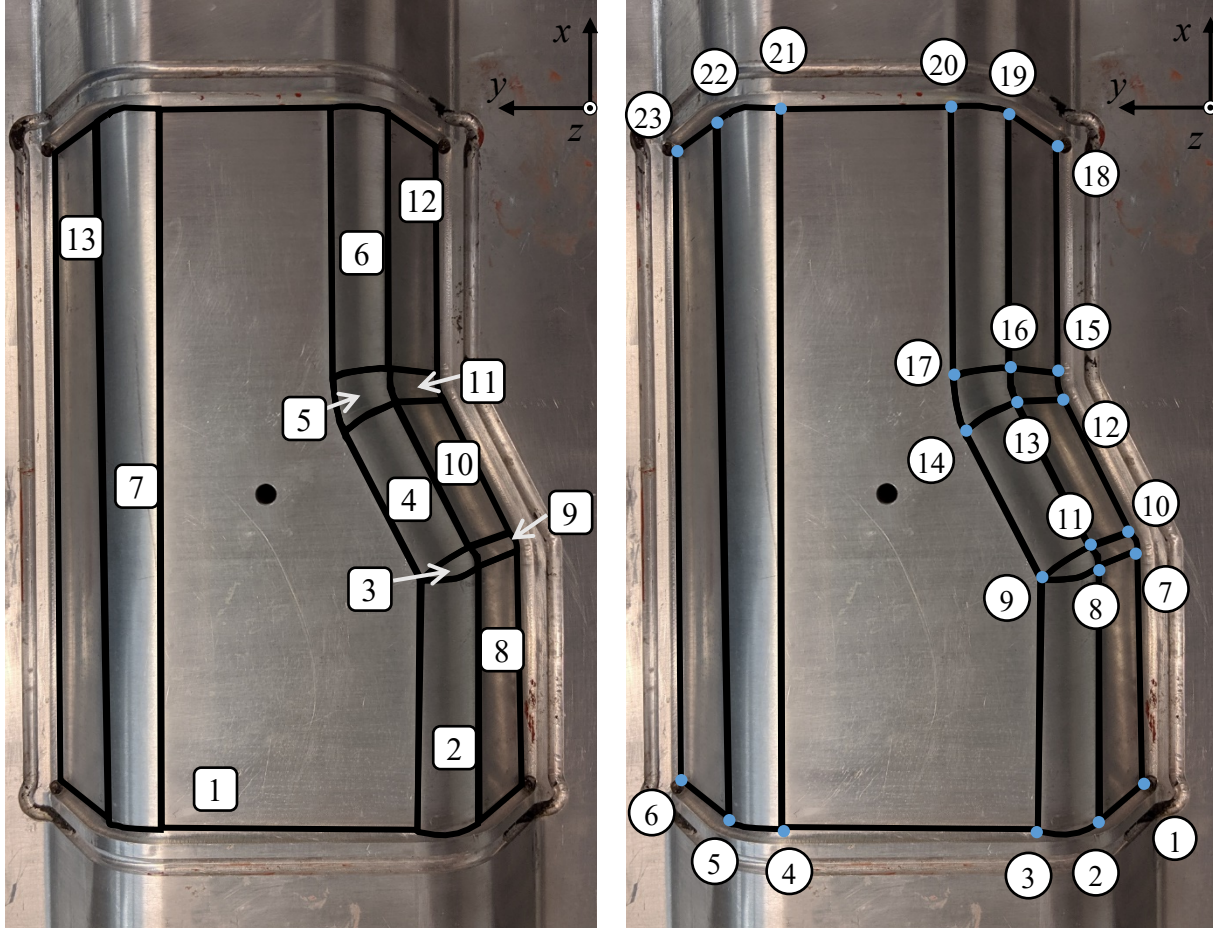


Figure 4-5: *FR* mould base surface (left) and vertex identification (right)

Next, the 12 geometric parameters of the conical frustum were calculated for each base surface using the vertex coordinates and measured data. For conical frustum calculations, each base surface must have three or four vertices. In the case of base surface 1 which features 7 vertices, the surface had to be split so that conical frustum parameters could be calculated, Figure 4-6. The original surface was split in four base surfaces using sections of edge 4-21 perpendicular to the x -axis, adding three new vertices between vertices 4 and 21, directly in line with vertices 9, 14 and

17. The resulting surfaces were then relabelled as base surfaces 1-1, 1-2, 1-3 and 1-4. These new vertices were added along a common edge shared by base surface 7 adjacent to base surface 1. These vertices do not need to be considered when listing the vertices for the conical frustum calculations, as edge 4-21 is straight. In cases where the edge may be curved, if the radius remains constant through all additional vertices only the end vertices need to be considered for the base surface.

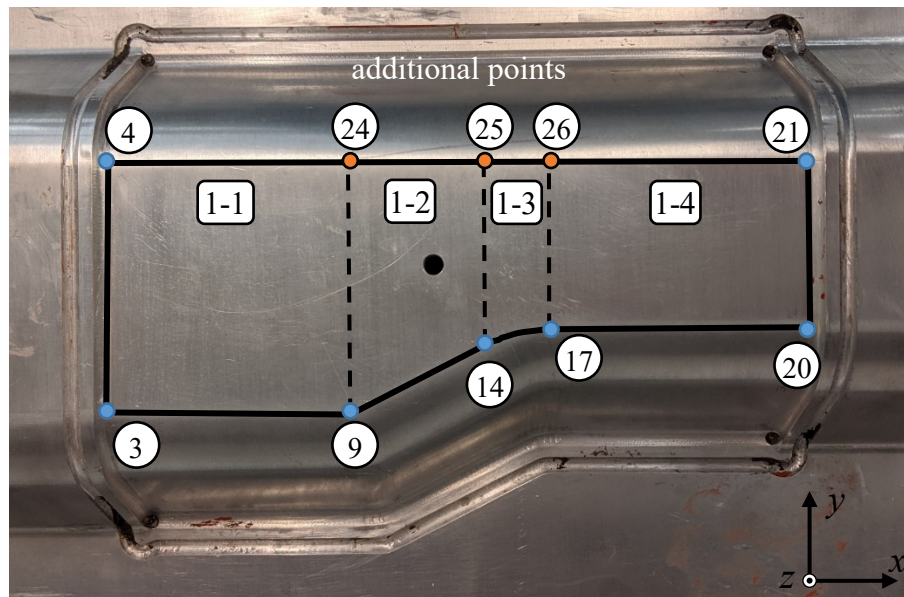


Figure 4-6: Surface splitting and additional points

With the mould surface fully decomposed into base surfaces, conical frustum calculations are performed of each base surface. Table 4-1 lists all the base surfaces types and their associated vertices with the appropriate conical frustum labels. Table 4-2 lists all the conical frustum parameters for each base surface.

Table 4-1: FR mould base surface vertices

Base Surface		Vertex	Coordinates			Conical Frustum Vertex Label
			x	y	z	
1-1	Flat	2	0	1.5	2.5	A
		3	0	3.5	4	B
		24	7	10.5	4	D
		9	7	3.5	4	C
1-2	Flat	9	7	3.5	4	A
		24	7	10.5	4	B
		25	11.25	10.5	4	D
		14	11.25	5.5	4	C
1-3	Flat	14	11.25	5.5	4	A
		25	11.25	10.5	4	B
		26	13	10.5	4	D
		17	13	5.75	4	C
1-4	Flat	17	13	5.75	4	A
		26	13	10.5	4	B
		21	20	10.5	4	D
		20	20	5.75	4	C
2	Single Curvature	2	0	1.5	2.5	A
		3	0	3.5	4	B
		9	7	3.5	4	D
		8	7	1.5	2.5	C
3	Double Curvature	8	7	1.5	2.5	C
		9	7	3.5	4	A
		11	7.85	1.6	2.5	D
4	Single Curvature	11	7.85	1.6	2.5	A
		9	7	3.5	4	B
		14	11.25	5.5	4	D
		13	12	4	2.5	C
5	Double Curvature	13	12	4	2.5	C
		14	11.25	5.5	4	A
		17	13	5.75	4	B
		16	13	4.25	2.5	D
6	Single Curvature	16	13	4.25	2.5	A
		17	13	5.75	4	B
		20	20	5.75	4	D
		19	20	4.25	2.5	C
7	Single Curvature	4	0	10.5	4	A
		5	0	12.5	2.5	B
		22	20	12.75	2.5	D
		21	20	10.5	4	C
8	Flat	1	0	0	0	A
		2	0	1.5	2.5	B
		8	7	1.5	2.5	C
		7	7	0	0	D
9	Single Curvature	7	7	0	0	C
		8	7	1.5	2.5	A
		11	7.85	1.6	2.5	B
		10	7.85	0.1	0	D
10	Flat	10	7.85	0.1	0	A
		11	7.85	1.6	2.5	B
		13	12	4	2.5	C
		12	12	2.5	0	D
11	Single Curvature	12	12	2.5	0	C
		13	12	4	2.5	A
		16	13	4.25	2.5	B
		15	13	2.75	0	D
12	Flat	15	13	2.75	0	A
		16	13	4.25	2.5	B
		19	20	4.25	2.5	C
		18	20	2.75	0	D
13	Flat	5	0	12.5	2.5	A
		6	0	14	0	B
		23	20	14.25	0	C
		22	20	12.75	2.5	D

Table 4-2: *FR* mould conical frustum parameters

Base Surface		Conical Frustum Parameters											
		$ \vec{R_1} $	$ \vec{r_1} $	$ \vec{R_2} $	$ \vec{r_2} $	C_R	C_r	$ \vec{L} $	β_1	β_2	θ_1	θ_2	α
1-1	F	∞	∞	∞	∞	∞	∞	7	n/a	n/a	180°	0°	0°
1-2	F	∞	∞	∞	∞	∞	∞	4.37	n/a	n/a	165°	25°	0°
1-3	F	∞	∞	∞	∞	3.5	∞	1.75	n/a	n/a	176°	14°	0°
1-4	F	∞	∞	∞	∞	∞	∞	7	n/a	n/a	180°	0°	0°
2	SC	2.5	2.5	2.5	2.5	∞	∞	7	60°	60°	180°	0°	0°
3	DC	0	0	3	3	2.5	2.5	2.5	n/a	18°	n/a	0°	0°
4	SC	2.5	2.5	2.5	2.5	∞	∞	4.75	60°	60°	180°	0°	0°
5	DC	3.5	3.5	3	3	2.5	2.5	2.5	29°	18°	126°	64°	0°
6	SC	2.5	2.5	2.5	2.5	∞	∞	7	60°	60°	180°	0°	0°
7	SC	2.5	2.5	2.5	2.5	∞	∞	20	60°	60°	180°	0°	0°
8	F	∞	∞	∞	∞	∞	∞	7	n/a	n/a	180°	0°	0°
9	SC	2.5	2.5	3	3	∞	∞	2.69	20°	16°	158°	22°	0°
10	F	∞	∞	∞	∞	∞	∞	4.8	n/a	n/a	180°	0°	0°
11	SC	3	3	3	3	∞	∞	3.2	24°	20°	141°	49°	0°
12	F	∞	∞	∞	∞	∞	∞	7	n/a	n/a	180°	0°	0°
13	F	∞	∞	∞	∞	∞	∞	20	n/a	n/a	180°	0°	0°
* F – Flat, SC – Single Curvature, DC – Double Curvature													
** All radii and lengths in centimetres													

From the calculations of the conical frustum parameters, one can observe that parameters that are capable of having different values within a single base surface often do not change. Looking specifically at the use of circles instead of ellipses, all the base surfaces found on the mould *FR* do not have different values for $|\vec{R_1}|$ and $|\vec{r_1}|$, or $|\vec{R_2}|$ and $|\vec{r_2}|$ values, meaning ellipses were not used. For flat base surfaces all radii are infinite, causing the sectioning and twist angles to become obsolete. The only conical frustum values that were needed for modelling flat base surfaces are infinite radii, the distance between the centres of edges *AB* and *CD*, $|\vec{L}|$ and the opposing tilt angles. The tilt angles in flat base surfaces directly influence one another always summing to 180°, Figure 4-8 in section 5.5. Double curvature base surfaces present differences in $|\vec{R_1}|$, $|\vec{R_2}|$, and $|\vec{r_1}|$, $|\vec{r_2}|$ but maintain a constant connect radius causing $C_R = C_r$. From the decomposition of mould *FR* it can be noted that many base surfaces are described by similar or the same geometric parameters.

4.4 *BD100* mould decomposition

A similar method for mould decomposition was followed for mould *BD100*. However, the symmetry found on this mould required that only half of the surface be broken into base surfaces as the results can be applied to both halves, Figure 4-7. The MicroScribe CMM was used for extracting 46 vertices, including 8 additional for split flat surfaces, leading to 24 base surfaces. Radii were measured using the digital radius gauge, even the very large ones. In some cases where the radius gauge could not be used, a third point was extracted so that the radius could be calculated.

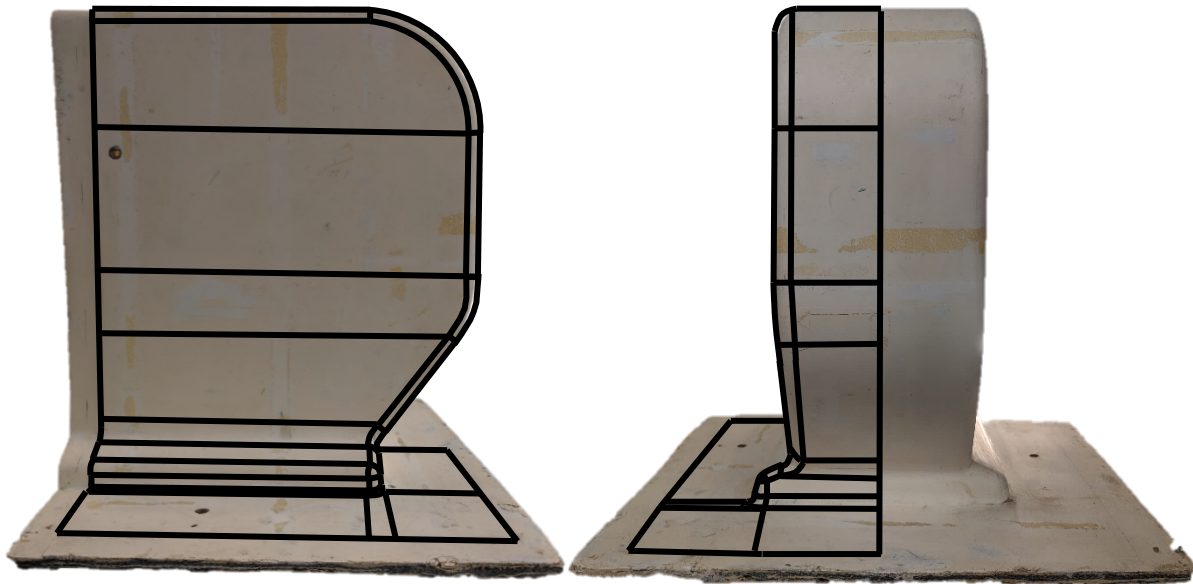


Figure 4-7: *BD100* symmetric half-mould decomposition

As observed with the *FR* mould, the *BD100* mould presented similarities in the repetitive nature of the conical frustum geometric parameters. The list of base surfaces and associated vertices and conical frustum vertex labels in a format similar to Table 4-1 can be found in Appendix D. The conical frustum parameters are presented in Table 4-3 below.

Table 4-3: *BD100* mould conical frustum parameters

Base Surface		Conical Frustum Parameters											
		$ \vec{R}_1 $	$ \vec{r}_1 $	$ \vec{R}_2 $	$ \vec{r}_2 $	C_R	C_T	$ \vec{L} $	β_1	β_2	θ_1	θ_2	α
1-1	F	∞	∞	∞	∞	∞	∞	23	n/a	n/a	180°	0°	0°
1-2	F	∞	∞	∞	∞	∞	∞	1.42	n/a	n/a	180°	0°	0°
1-3	F	∞	∞	∞	∞	∞	∞	6	n/a	n/a	180°	0°	0°
1-4	F	∞	∞	∞	∞	∞	∞	10.89	n/a	n/a	180°	0°	0°
2	SC	0.32	0.32	0.32	0.32	∞	∞	23	90°	90°	180°	0°	0°
3	F	∞	∞	∞	∞	∞	∞	23	n/a	n/a	180°	0°	0°
4	SC	1.2	1.2	1.2	1.2	∞	∞	23	30°	30°	180°	0°	0°
5	F	∞	∞	∞	∞	∞	∞	23	n/a	n/a	180°	0°	0°
6	SC	1.11	1.11	1.11	1.11	∞	∞	23	88°	88°	180°	0°	0°
7-1	F	∞	∞	∞	∞	∞	∞	7.15	n/a	n/a	161°	19°	0°
7-2	F	∞	∞	∞	∞	9.5	∞	5.2	n/a	n/a	169°	11°	0°
7-3	F	∞	∞	∞	∞	∞	∞	12	n/a	n/a	180°	0°	0°
7-4	F	∞	∞	∞	∞	8.65	∞	8.65	n/a	n/a	154°	26°	0°
8	SC	1.35	1.35	1.35	1.35	∞	∞	21.5	90°	90°	180°	0°	0°
9	F	∞	∞	∞	∞	∞	∞	6.25	n/a	n/a	180°	0°	0°
10	DC	1.25	1.25	1.42	1.42	0.32	0.32	0.32	90°	90°	180°	0°	0°
11	SC	1.11	1.11	1.25	1.25	∞	∞	0.85	90°	90°	180°	0°	0°
12	DC	0	0	1.11	1.11	1.25	1.2	1.11	n/a	90°	n/a	0°	0°
13	SC	1.11	1.11	1.25	1.25	∞	∞	1.62	74.5°	74.5°	154°	17°	6°
14	DC	1.11	1.11	5	5	1.11	1.35	2.95	111°	38°	113°	67°	47°
15	SC	1.35	1.35	1.35	1.35	∞	∞	9.11	45°	55°	155°	25°	3°
16	DC	9.5	9.5	10.85	10.85	1.35	1.35	1.35	35°	35°	180°	0°	0°
17	SC	1.35	1.35	1.35	1.35	∞	∞	12	90°	90°	180°	0°	0°
18	DC	8.65	8.65	10	10	1.35	1.35	1.35	90°	90°	180°	0°	0°
19	SC	10	10	10	10	∞	∞	6.25	90°	90°	180°	0°	0°
20	F	∞	∞	∞	∞	∞	∞	6.25	n/a	n/a	180°	0°	0°
21	SC	10.85	10.85	10.85	10.85	∞	∞	6.25	35°	35°	180°	0°	0°
22	F	∞	∞	∞	∞	∞	∞	6.25	n/a	n/a	180°	0°	0°
23-1	F	5	5	∞	∞	5	∞	1.7	n/a	n/a	117°	63°	0°
23-2	F	∞	∞	∞	∞	∞	∞	10.15	n/a	n/a	100°	80°	0°
23-3	F	∞	∞	∞	∞	∞	∞	10.89	n/a	n/a	180°	0°	0°
24	SC	0.32	0.32	0.32	0.32	∞	∞	10.89	90°	90°	180°	0°	0°
*F - Flat, SC - Single Curvature, DC - Double Curvature													
** All radii and lengths in centimetres													

4.5 Hierarchy of parameters: Five parameter conical frustum

The conical frustum can be used for modelling any base surface. However, many base surfaces encountered on the two decomposed moulds discussed in the previous pages were observed to be relatively simple. This suggests that many base surfaces can be expressed by varying only some of the geometric parameters of the conical frustum, while the full list of geometric parameters and

flexibility of the conical frustum remains useful for a smaller number of scenarios. Establishing an order of importance or a hierarchy of parameters in the sense of those that are more often changed, allows geometric parameters of more importance to be featured in a core version of the conical frustum. From a more general perspective, this may be very useful because it allows a database to be populated with an initial focus on base surfaces that are more frequently encountered.

To establish a hierarchy of parameters, conical frustum parameters were analyzed based on how frequently their values changed and on their difference from one another, specifically for major radii $|\vec{R}_1|$ and $|\vec{R}_2|$, minor radii $|\vec{r}_1|$ and $|\vec{r}_2|$, connecting radii C_R and C_r , and sectioning angles β_1 and β_2 . It is important to note that the sequence in which the parameters are listed does not directly correlate to their overall importance as geometric features; the sequence solely represents the relative frequency of change.

Looking at flat base surfaces found on the two moulds, the parameter that varied most consistently was the distance between the centres of edge AB and CD , $|\vec{L}|$. Since the radii of the circles are infinite on flat base surfaces, the centres of edges AB and CD , where the centres would be for a base surface with measureable radii, are used for calculating $|\vec{L}|$, Figure 4-8 . The infinite radii cause twist angle α , and sectioning angles β_1 and β_2 to become obsolete. In most cases, tilt angles θ_1 and θ_2 remain constant at 180° and 0° respectively. When flat base surfaces are split so the conical frustum parameters can be determined, lengths of edges tend to vary, Figure 4-8. In the cases where the lengths of edges AB and CD are different, this causes the centres to be out of line resulting in a change in tilt angles θ_1 and θ_2 . These changes aren't causing any tilt out of the flat base surface plane, and therefore do not directly affect the base surface type.

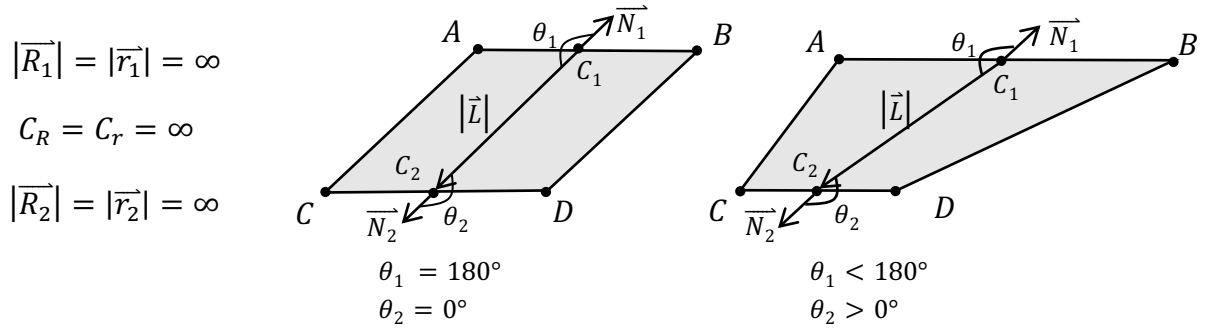


Figure 4-8: Uniform flat base surface (left); “titled” flat base surface (right)

During mould decomposition it was observed that a number of conical frustum parameters remained constant. Twist angle α remained constant at 0° (no twist) in all but three base surfaces on the *BD100* mould. For these two relatively simple moulds, twist is not a commonly repeating feature suggesting that on simpler, more common features such as edge and corner fillets, twisted base surfaces will not be encountered.

Most single curvature surfaces correspond to a portion of a full cylinder, setting the major radii $|\vec{R}_1|$ and $|\vec{R}_2|$ equal to their respective minor radii $|\vec{r}_1|$ and $|\vec{r}_2|$. Similarly for double curvature surfaces, spherical features corresponding to corners enable radii to become equal, as well as sectioning angles β_1 and β_2 and connecting radii C_R and C_r . All base surfaces in moulds *FR* and *BD100* featured tilt angles θ_1 and θ_2 that nearly always remain opposite to one another, except for three cases, surface 5 on mould *FR*, and surfaces 13 and 14 on *BD100*.

With a number of geometric parameters unchanged and equal, a simplified five parameter conical frustum shown in Figure 4-9 can be used in modelling most base surfaces. These five parameters are $|\vec{R}_1|$, $|\vec{R}_2|$, C_R , β and $|\vec{L}|$. The core conical frustum does not eliminate the other seven

parameters, but sets their values equal to parameters that exist within the same plane, to values that are opposite from one another (tilt angles only), or to zero:

- $|\vec{R}_1| = |\vec{r}_1|$
- $|\vec{R}_2| = |\vec{r}_2|$
- $C_R = C_r$
- $\beta_1 = \beta_2$
- $\theta_1 = 180^\circ, \theta_2 = 0^\circ$
- $\alpha = 0^\circ$

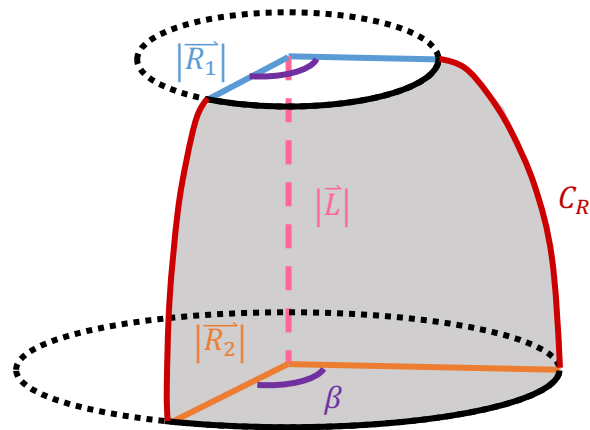


Figure 4-9: Five parameter core conical frustum

5 Basic conical frustum draping simulations: Taguchi study

The chapter discusses draping simulations performed using Nottingham DrapeIt with simulation parameters selected using Plackett-Burman/Taguchi methods for experimental design. These simulations were performed in order to analyze both geometric and draping parameters, which change over a mould for different draping scenarios. In this work, the effect of geometric and draping parameters on shear angles were investigated and introduces the fundamentals of in-plane shear. Full database population including a more in-depth look at in-plane shear is discussed in Chapter 6.

5.1 Draping parameters

Alongside geometric parameters, draping parameters must be considered when performing draping simulations. As the simulations reported in this work were performed using Nottingham DrapeIt, four variable draping parameters were identified:

- Draping starting position P_s
- Starting yarn orientation O_s
- Initial angle between warp and weft yarns (pre-existing shear) ω
- Fibre spacing S

These parameters were used for structuring the database, alongside the geometric parameters of the conical frustum. The draping starting position refers to the entry point as the fabric enters a base surface. When linking multiple base surfaces through common edges, the starting position dictates how the fabric moves onto the adjacent base surface. Starting yarn orientations can vary

along a common edge between two base surfaces when in-plane shear is observed. The starting yarn orientation along each edge of a base surface is expressed as warp and weft yarn orientation vectors extracted from draping simulations, Figure 5-1.

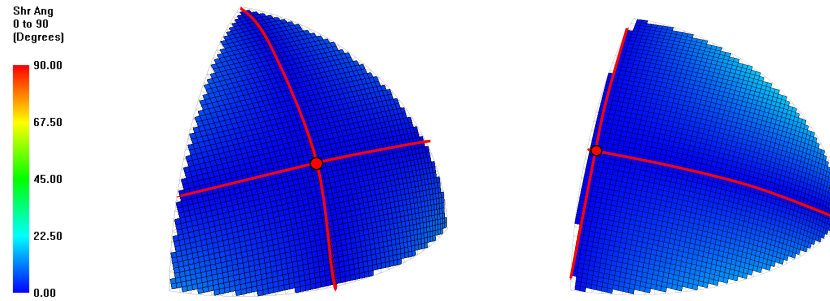


Figure 5-1: Draping parameters: Starting position and orientation

The initial angle between the warp and weft yarns at the starting position quantifies pre-existing shear present in the fabric as it enters a base surface, Figure 5-2. Moving from base surface to base surface will induce in-plane shear, especially in instances of double curvature. These angular changes between the warp and weft yarns occur following the pin-joint mechanism discussed in Section 2.4.4. Considering the behaviour of fabrics that have already undergone a degree of shear prior to moving into an adjacent surface is essential in developing predictions over multiple connected base surfaces. Parts are normally draped with a fabric that is not sheared initially, hence the initial angle in practice is generally zero - in the case of a simulation that aims at covering a full part. However, the approach developed in this thesis, multiple simulations are performed in order to populate a database of results that apply to individual base surfaces. When covering a full part, numerous base surfaces are covered sequentially. Other than at the very start, most base surfaces will be draped with an initial angle that is different from zero as a result of the shear induced by other base surfaces prior in the sequential process of draping the full part. Therefore,

this characteristic is seldom use in traditional draping simulation where a trial and error simulation approach is used in attempting to drape a full part all at once. However for the current approach, the initial angle is critically relevant. Finally, fibre spacing was included so that all variable parameters in Nottingham DrapeIt were investigated, but results presented in Section 5.6 confirmed the expected findings that this draping parameters does not affect in-plane shears angle or yarn orientations.

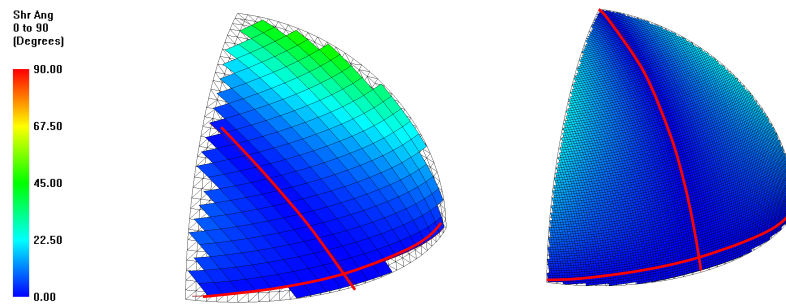


Figure 5-2: Draping parameters: Initial angle between the warp and weft, and fibre spacing

5.2 Design of experiments

To analyze which parameters had the greatest effect on in-plane shear, the effects of the individual geometric and draping parameters need to be quantified. To do this, two levels were set for each of the parameters so that base surfaces could be generated from the 12 conical frustum geometric parameters, and draping simulations could be applied to the generated base surfaces. As a result of performing all the combinations what is known as a full factorial of experiments, simulations in this case, would be completed.

When considering performing a full factorial of experiments, the reduction in geometric parameters established by a hierarchy of parameters, discussed in Chapter 4, assists in reducing the total number of draping simulations. Using the 12 conical frustum geometric parameters, $2^{12} =$

4096 base surfaces could be generated, without even considering the draping parameters. In total, $2^{16} = 65\,536$ simulations would need to be performed for a full factorial. Reducing to 5 geometric parameters, $2^9 = 512$ simulations would be needed when draping parameters are included. Achieving this is time consuming, hence a fractional plan featuring specific runs was chosen.

As per Taguchi methodology discussed in Section 2.6, the resolution of the database can be increased on an ongoing basis using the 12 parameter conical frustum, by running more simulations. For this work it was decided that 16-runs would be used for the analysis on the 9 factors. This reduces that amount of work required very significantly but confounds the effects of several interaction of factors. The patterns of confounding between the main effects of parameters and the effects of interaction are explicit and known precisely, as per Taguchi methodology. The simulations were performed following a basic 16-run 2-level Plackett-Burman design with nine factors. Instead of designing the simulation plan, existing, well-established screening tables existing in literature were used. The experimental design presented in Table 5-1 was adapted directly from [51]. From this table, interactions can be observed between the parameters however more interactions are present and discussed in Chapter 6. The + and – in the table represent the two levels used for each of the parameters and their selected values.

Upon completing this initial experimental design, another plan was performed to clarify and further confirm the findings. This is achieved using a reflected run where the design is mirrored, selecting all the modalities other than those used in the first plan. This reflected study can determine whether responses are affected by main factors or by confounding factors, relieving some of the confounding found in the initial plan. The reflected study also allows factors that exhibit very strong or weak effects to be removed so that a comparison can be performed between the other factors.

Table 5-1: 16-run design experiment for the analysis of geometric and draping parameters

Run	Contrast and Confounding Factors								
	A	B	C	D	G	H	K	M	N
	-BC	-AC	-AB	-CG	-CD	-CK	-CH	-CN	-CM
			-DG						
			-HK						
			-MN						
1	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	+	+	+	+
3	-	-	-	+	+	+	+	-	-
4	-	-	-	+	+	-	-	+	+
5	-	+	+	+	-	-	+	+	-
6	-	+	+	+	-	+	-	-	+
7	-	+	+	-	+	+	-	+	-
8	-	+	+	-	+	-	+	-	+
9	+	+	-	-	-	-	-	+	+
10	+	+	-	-	-	+	+	-	-
11	+	+	-	+	+	+	+	+	+
12	+	+	-	+	+	-	-	-	-
13	+	-	+	+	-	-	+	-	+
14	+	-	+	+	-	+	-	+	-
15	+	-	+	-	+	+	-	-	+
16	+	-	+	-	+	-	+	+	-

5.2.1 Input Factors

For this experimental simulation plan nine binary factors were used; the five geometric parameters of the basic conical frustum and the four draping parameters highlighted in Section 5.1. These factors are presented in Table 5-2.

Table 5-2: Taguchi experimental design including parameter modalities

Conical Frustum and Draping Parameters	Taguchi Parameter	Modalities	
		-	+
Radius of circle 1 $ \vec{R}_1 $	A	0 cm	2 cm
Radius of circle 2 $ \vec{R}_2 $	B	1 cm	2 cm
Distance between circles $ \vec{L} $	C	0.5 cm	2 cm
Radius connecting circles C_R	D	2 cm	20 cm
Sectioning angle β	G	45°	90°
Draping starting position P_s	H	Centre of base surface	Centre of longest edge
Starting fibre spacing S	K	1/20 of longest edge	1/50 of longest edge
Angle between warp and weft ω	M	90°	60° (30° of shear)
Starting yarn orientation O_s	N	0°	45°

The modalities of the geometric parameter were chosen so that the single curvature and double curvature base surfaces could be modelled for the experimental plans. Near flat surfaces are represented in the selection of the modalities however from the definition in Chapter 4, they cannot be considered flat. Flat surfaces do not affect shear thus they were not considered for modelling in this experiment. Similarly, single curvature typically does not affect shear either, however testing the behaviour of these surfaces after shear is applied is useful when linking the surfaces. The radius of circle 1 $|\vec{R}_1|$ was given a minimum of 0 cm so that double curvature corners could be built, Figure 5-3. With no radius, the top circle becomes a single point connected to the 2nd circle by radius C_R , and therefore is always represented by the two points C and D from the conical frustum.

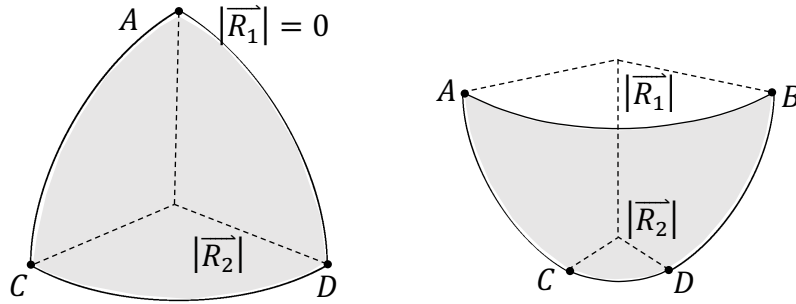


Figure 5-3: Double curvature base surfaces

The maximum radii for both circles were set to the same value so that single curvature base surfaces could be modelled when the connecting radius C_R is set to its maximum of 20 cm, Figure 5-4. This value was set much larger than any other radii so that a straight line could be approximated, in comparison to other radii. Since the maximum distance between circles is set to 2 cm, a radius 10 times that length effectively simulates a straight edge.

In the case where $|\vec{R}_1| = 0$ cm, the large connecting radius generates surfaces that are near flat, especially when both minimum values for $|\vec{L}|$ and β are used. In comparison to the other single

curvature and double curvature base surfaces these surfaces have a very low amount of curvature and are expected to behave as flats.

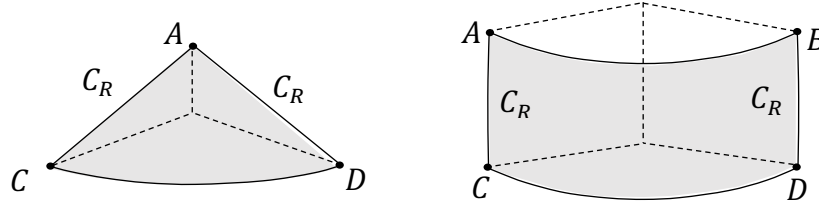


Figure 5-4: Near-flat and single curvature base surfaces

Double curvature surfaces always induce in-plane shearing during draping and they contribute directly to the amount of shear on the base surface. An increase and decrease in curvature for constant values of four out of five geometric parameters is expected to show a respective increase and decrease in shear. For example, as the arc lengths increase for a base surface based on the increase of sectioning angle β , the observable shear will be greater than the observable shear on a base surface with a smaller β as an effect of increased curvature. The maximum and minimum modalities of both β and $|\vec{L}|$ were set to observe the effect of changing the arc lengths, especially for double curvature features, Figure 5-5.

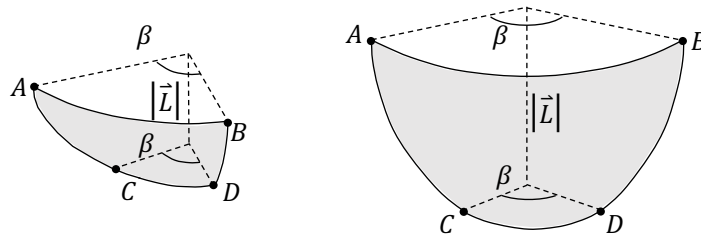


Figure 5-5: Increasing base surfaces surface curvature

For the draping parameters, the drape starting position will almost always be along a common edge between base surfaces thus the middle of the longest edge is chosen. A central draping point is a good representation when considering the first base surface to be draped in a draping strategy. Since fibre spacing is a variable parameter in Nottingham and it is used for completeness. It is expected to have no effect on the shear results. As stated in Section 5.1 the angle between the warp and weft quantifies the amount of shear on the fabric prior to draping. It is a very useful parameter since in most cases the fabric will be sheared prior to entering an adjacent base surface when building a draping strategy. This parameter is expected to have a large effect on the shear response in this experiment since it is applying 30° of shear. When performing draping simulations with a starting angle between warp and weft yarns greater or less than 90°, Nottingham calculates shear based on the starting orientation and does not include the additional shear. Post-processing techniques outlined in Appendix C discuss the necessary steps for calculating the actual shear angle. The images in the following sections do not display this adjustment. Finally, yarn starting orientation can contribute to the total shear on a surface. The most common draping orientations are 0/90° and $\pm 45^\circ$ thus those values were selected as starting orientation for this experiment.

5.3 Draping simulations

A total of 32 base surfaces were modelled using ANSYS Mechanical APDL following the steps described in Appendix A. This modelling work built 16 double curvature, 12 single curvature and 4 near-flat base surfaces. Model numbers and base surface types were assigned to each base surface and linked to the appropriate draping parameters for the 16 runs in this experimental simulation plan, Table 5-3. Models that were created but not used in this initial draping experiment and group of simulations are used at a later point in the thesis, during database population discussed in Chapter 6.

ANSYS models enable precise node locations to be determined so that the centres of base surfaces and the midpoint of the longest edge can be identified for the simulation starting positions. These locations were used in Nottingham DrapeIt when performing draping simulations so that the correct starting position were used. For each draping simulation, the maximum, minimum, average and absolute average shear angles over all elements were calculated from exported direction vector files (.mdv). The file listed warp orientation vectors and weft orientation vectors pointed in the same direction for each element dependent on the starting orientations of the simulation. Appendix C details how each orientation vector was used to calculate shear angles for each base surface.

In the direction vector files, shear angles were denoted as positive or negative. A positive shear angle refers to angles calculated between the warp and weft yarn vectors decreasing from 90° , whereas a negative shear angle refers to angles between warp and weft yarn vectors increasing from 90° . Shear is always measured from warp to weft. For example, if an expressed a shear angle of 10° , the angle between the warp and weft vectors would be 80° . Similarly, for a shear angles of -10° , the angle between the warp and weft vectors would be 100° . The average and average of the absolute shear angles quantify overall shear over a base surface. When these values are equal of similar, the variation of shear over the surface is limited. A positive or negative average shear is directly influenced by the base surface and whether a positive or negative shear is induced. Traditionally for double curvature surface, saddle configurations tend to induce negative shear whereas sphere configurations induce positive shear. A great absolute shear than average shear suggests that there is a great amount of an increase in both positive and negative shear, resulting in more overall shear on the surface. The post-processing steps for the simulations can be found in Appendices B and C.

Table 5-3: 16-run draping simulations

Run	Draping Parameters				Model Number	Base Surface Type
	Starting Point	Fibre Spacing	Angle between Warp and Weft	Fibre Orientation		
1	Centre of Base Geometry	1/20 Longest Edge	90° (no shear)	0°	1	DC
2	Middle of Longest Edge	1/50 Longest Edge	60° (30° of shear)	45°	1	DC
3	Middle of Longest Edge	1/50 Longest Edge	90° (no shear)	0°	4	NF
4	Centre of Base Geometry	1/20 Longest Edge	60° (30° of shear)	45°	4	NF
5	Centre of Base Geometry	1/50 Longest Edge	60° (30° of shear)	0°	26	NF
6	Middle of Longest Edge	1/20 Longest Edge	90° (no shear)	45°	26	NF
7	Middle of Longest Edge	1/20 Longest Edge	60° (30° of shear)	0°	27	DC
8	Centre of Base Geometry	1/50 Longest Edge	90° (no shear)	45°	27	DC
9	Centre of Base Geometry	1/20 Longest Edge	60° (30° of shear)	45°	21	DC
10	Middle of Longest Edge	1/50 Longest Edge	90° (no shear)	0°	21	DC
11	Middle of Longest Edge	1/50 Longest Edge	60° (30° of shear)	45°	24	SC
12	Centre of Base Geometry	1/20 Longest Edge	90° (no shear)	0°	24	SC
13	Centre of Base Geometry	1/50 Longest Edge	90° (no shear)	45°	14	SC
14	Middle of Longest Edge	1/20 Longest Edge	60° (30° of shear)	0°	14	SC
15	Middle of Longest Edge	1/20 Longest Edge	90° (no shear)	45°	15	DC
16	Centre of Base Geometry	1/50 Longest Edge	60° (30° of shear)	0°	15	DC

*NF = Near-flat, SC = Single Curvature, DC = Double Curvature

5.3.1 Runs #1 and #2

The first draping simulations were performed on a double curvature base surface, Figure 5-6. In run #1, in-plane shear is more spread out around the central starting point. Limited values of maximum and minimum shear, 2.3° and -4.3° respectively, were observed due to the small surface area in comparison to the surface radii. In run #2, a maximum in-plane shear of 37° occurs on the opposite side of the base surface from the starting point. The large shear is inherently caused by run #2 starting with the angle between the warp and weft at 60°. A sectioning angle β of 45° causes the small surface area, directly influencing the total in-plane shear.

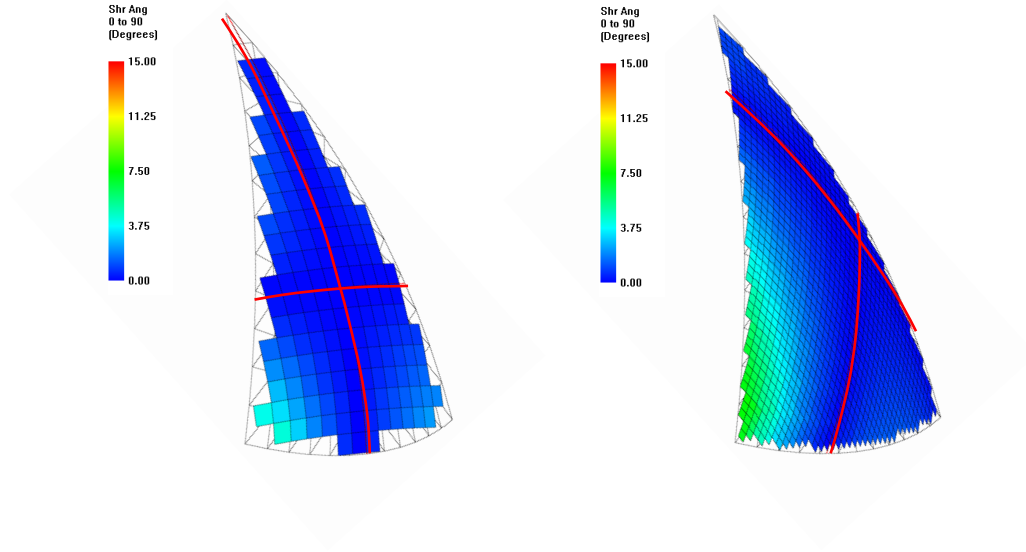


Figure 5-6: Runs #1 and #2 draping simulation results

5.3.2 Runs #3 and #4

Runs #3 and #4 were performed on a large, near-flat base surface, Figure 5-7. The only observable shear is due to the initial angle between warp and weft set to 60° in run #4 resulting in maximum and minimum shear angles of 30° each. In both simulations the angle between warp and weft remains constant over the entire surface.

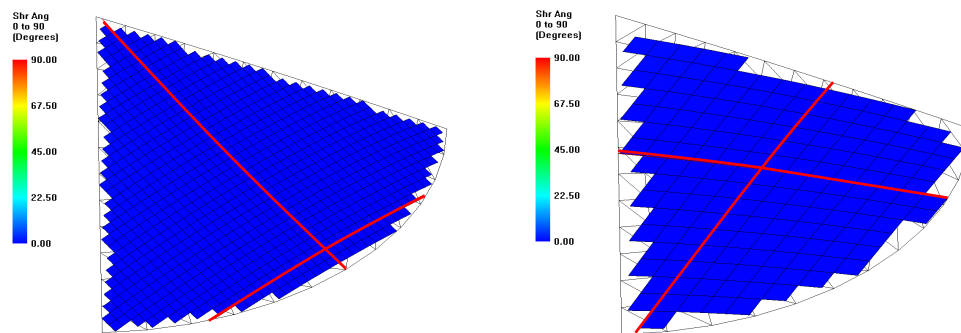


Figure 5-7: Runs #3 and #4 draping simulation results

5.3.3 Runs #5 and #6

Runs #5 and #6 exhibit results similar to those seen in run #3 and #4. The flat surface generates no additional shear in both cases, Figure 5-8. For run #5 maximum and minimum shear angles of 30° and 29° respectively were observed. For run #4, a single maximum and minimum shear angle of 0° was observed.

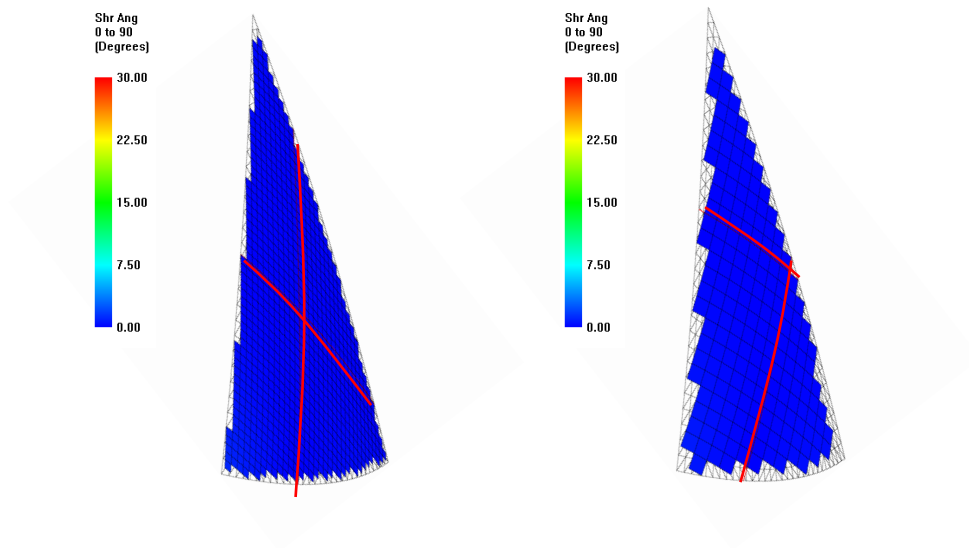


Figure 5-8: Runs #5 and #6 draping simulation results

5.3.4 Runs #7 and #8

Runs #7 and #8 were performed on a large double curvature surface, Figure 5-9. Run #7 shows a large amount of shear along the opposite edge from the starting position, with a minor amount of shear along the third edge. The maximum and minimum in-plane shear for run #7 is 35° and -2.5° respectively. Shear is more distributed in run #8 with a central starting point, showing minimal amounts of shear along each edge, in comparison to run #7. The maximum and minimum shear angles for run #8 were 8.5° and -17° respectively.

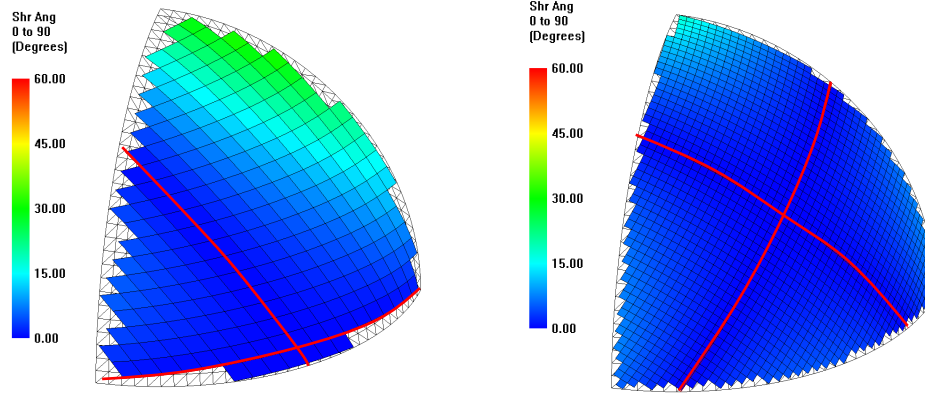


Figure 5-9: Run #7 and #8 draping simulation results

5.3.5 Runs #9 and #10

The model used for runs #9 and #10 is a small double curvature base surface, Figure 5-10. With a small distance between circles, $|\vec{L}|$, connecting radius C_R acts as a nearly straight edge causing the surface to appear as single curvature. Run #9 returns a maximum shear angle of 32° and a minimum of shear angle 29° due to the starting angle between warp and weft orientations. For run #10, no in-plane shear is observed.

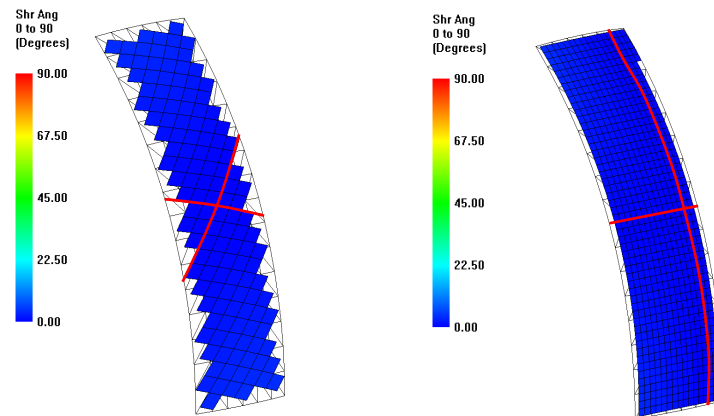


Figure 5-10: Runs #9 and #10 draping simulation results

5.3.6 Runs #11 and #12

A single curvature base surface was used in run #11 and #12, Figure 5-11. As observed in the previous runs, single curvature base surfaces did not add any additional shear over the surface. A maximum shear angle of 31° was found in run #11 and no shear was observed in run #12.

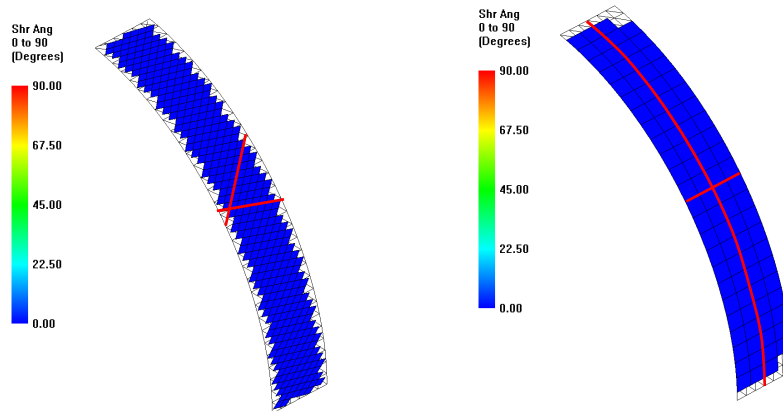


Figure 5-11: Runs #11 and #12 draping simulation results

5.3.7 Run #13 and #14

Runs #13 and #14 were performed on a large single curvature base surface, Figure 5-12. The results, as expected, remain similar as for runs #11 and #12. No in-plane shear was present in run #13 and a maximum shear angle of 30° was observed in run #14 due to the angle between the warp and weft.

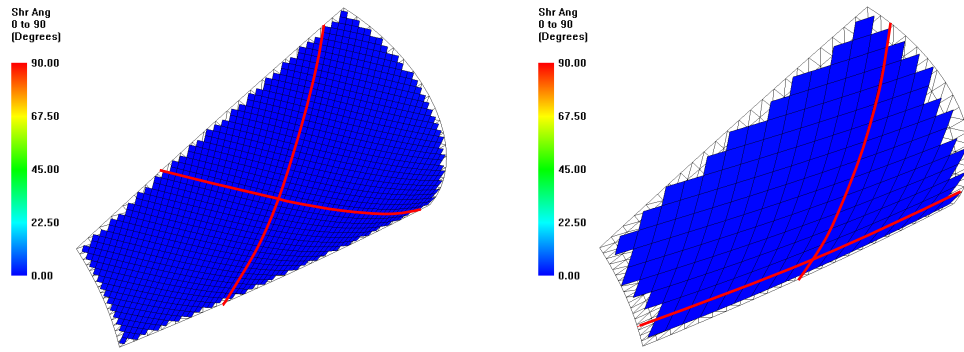


Figure 5-12: Runs #13 and #14 draping simulation results

5.3.8 Runs #15 and #16

Finally, runs #15 and #16 used a double curvature base surface, Figure 5-13. The change in starting orientation and position between the two runs led to significant differences in shear distribution over the surface. Similar to other runs with in-plane shear, run #15 exhibited maximum and minimum shear angles of 9° and -24° , respectively on the edge directly opposite from the starting point. Run #16 exhibited a similar trend with maximum and minimum shear angles of 39° and 5° respectively, that are reached at opposite ends of the base surface.

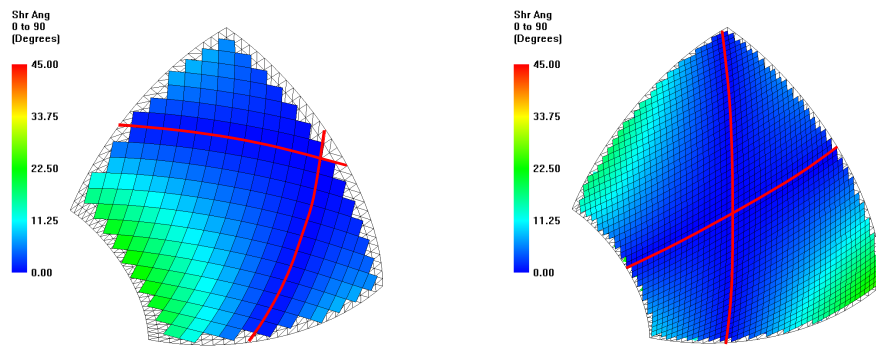


Figure 5-13: Runs #15 and #16 draping simulation results

5.4 Discussion

The results obtained from the draping simulations are presented in subsequent tables. Table 5-4 shows the maximum, minimum, average and absolute average shear angles for each of the runs. As expected, the near-flat and single curvature base surfaces did not contribute to in-plane shear; they only acted as transferring surfaces for the sheared fabric. Runs 3, 4, 5, 6, 9, 10, 11, 12, 13, and 14 were all performed on models that were either near-flat or single curvature surfaces. The only observable shear on these surfaces was due to the starting angle between the warp and weft yarns. The remaining runs were all performed on double curvature models that contributed more in-plane shear over the surfaces.

Table 5-4: Basic 16-run Taguchi simulation results

Run	Shear Angle (°)			
	Maximum	Minimum	Average	Absolute Average
1	2.33	-4.28	-0.12	0.81
2	31.76	22.91	29.21	29.21
3	0.57	-0.69	-0.04	0.21
4	30.64	29.69	30.17	30.17
5	30.33	28.77	29.94	29.94
6	0.60	-0.38	0.10	0.18
7	35.91	-2.53	22.54	22.52
8	8.54	-17.47	-0.15	3.10
9	32.47	29.14	30.45	30.45
10	0.09	-0.74	0.56	0.94
11	31.21	28.32	29.14	29.14
12	1.38	-1.48	-0.01	0.54
13	0.33	-1.09	-0.21	0.27
14	30.76	28.90	29.84	29.84
15	9.09	-24.18	-5.66	8.43
16	39.89	5.71	25.92	26.04

The base surfaces with the greater curvature always generated a greater amount of in-plane shear.

In-plane shear was still present in the smaller base surfaces such as Run #1 for example but did

not reach the same maximum as observed in the larger base surfaces. These surfaces have more curvature causing the fabric to continue to shear. The geometric parameters correlate directly to the change in surface area and define explicitly different base surface. Draping parameters, excluding fibre spacing also cause a change in shear. Fibre spacing appears to affect only the allowable covered surface in the simulations, resulting in some surfaces having more accurate or extensive results than others. Further statistical investigation is reported below to confirm which of the nine parameters have the largest effect on in-plane shear.

The main factors affecting in-plane shear were identified using Scree plots of the sum of square estimated contrasts $SS(C)$. Scree plots for all responses are shown in Figure 5-14. All the plots have the same appearance with factor M , the initial angle between warp and weft yarns, showing a significantly large effect on in-plane shear. This result was expected since changing the initial angle between the warp and weft directly shears the fabric.

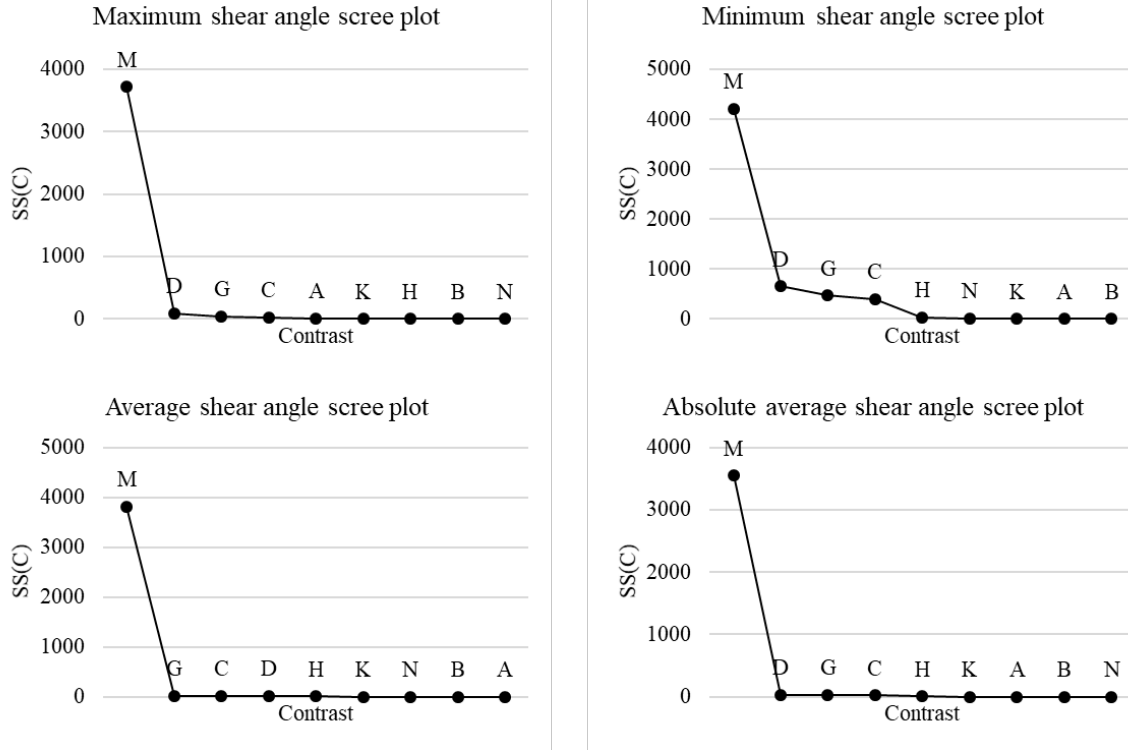


Figure 5-14: Basic 16-run shear angle Scree plots

Since factor M overshadows the other factors in Figure 5-14, another set of Scree plots excluding this factor were produced, Figure 5-15 revealing the effects of the remaining factors. In all cases factors D , C and G all have a significant effect on shear angle. Factors D , C and G are all geometric parameters linked to defining double curvature base surfaces and surface with higher curvature. Factor D , the radius C_R connecting the circles, dictated whether flat, single curvature or double curvature base surfaces were generated. With a greater number of double curvature surfaces featured in the runs, factor D is observed to have the second largest effect on in-plane shear. Both factors G and C , the sectioning angle β and the length between circles $|\vec{L}|$ respectively, directly influence the overall curvature of base surfaces. Double curvature leads to increased shear over the surface, hence these parameters having a significant effect on overall in-plane shear.

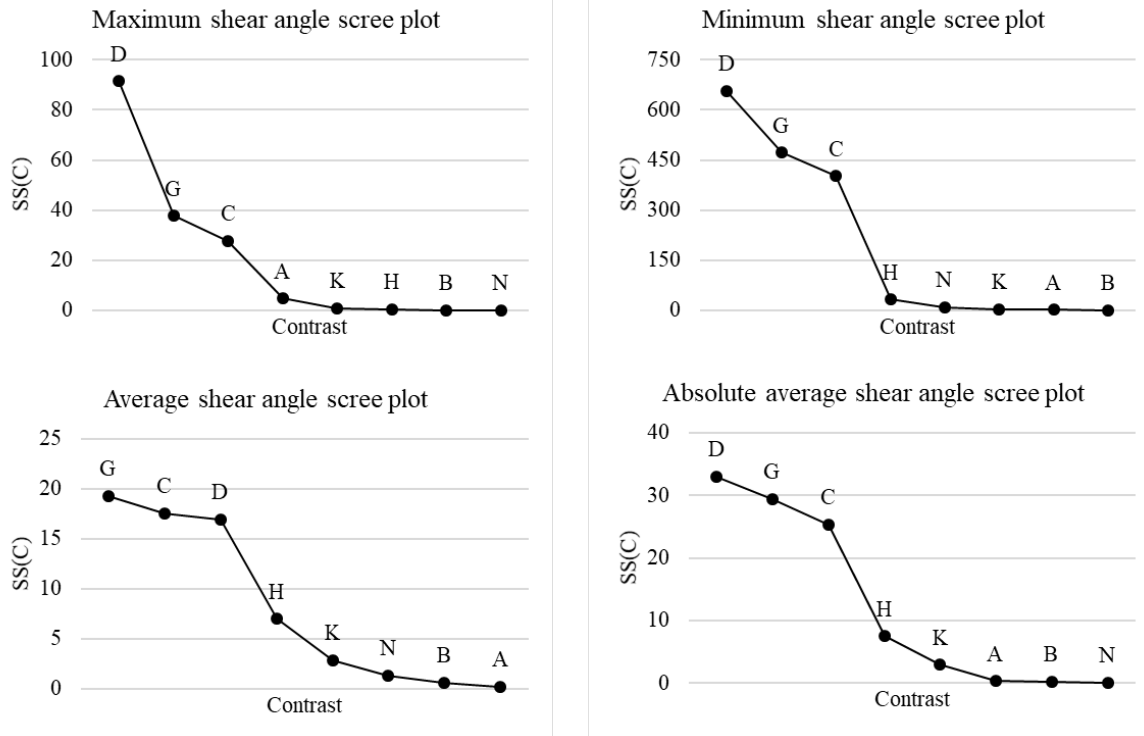


Figure 5-15: Basic 16-run shear angle Scree plots, M removed

5.5 Follow up study: Reflected runs

A follow up study was performed using the same geometric and draping parameters. A similar two level 16-run design was used inverting all the maximum and minimum modalities as outlined in Table 5-5. This classic investigation technique is known as performing a fully reflected simulation plan [46]. By inverting the modalities of the factors, the second study could confirm the results obtained in the initial plan and dictate if the contrasts are main effects or interaction of factors. Furthermore, all simulations and data extracted performed during these experiments were added to Database A.

Table 5-5: 16-run design experiment for fully reflected study

Run	Fully Reflected Contrast and Confounding Factors								
	A	B	C	D	G	H	K	M	N
	-BC	-AC	-AB	-CG	-CD	-CK	-CH	-CN	-CM
			-DG						
			-HK						
			-MN						
R1	+	+	+	+	+	+	+	+	+
R2	+	+	+	+	+	-	-	-	-
R3	+	+	+	-	-	-	-	+	+
R4	+	+	+	-	-	+	+	-	-
R5	+	-	-	-	+	+	-	-	+
R6	+	-	-	-	+	-	+	+	-
R7	+	-	-	+	-	-	+	-	+
R8	+	-	-	+	-	+	-	+	-
R9	-	-	+	+	+	+	+	-	-
R10	-	-	+	+	+	-	-	+	+
R11	-	-	+	-	-	-	-	-	-
R12	-	-	+	-	-	+	+	+	+
R13	-	+	-	-	+	+	-	+	-
R14	-	+	-	-	+	-	+	-	+
R15	-	+	-	+	-	-	+	+	-
R16	-	+	-	+	-	+	-	-	+

5.5.1 Runs #R1 and #R2

The first set of reflected runs were performed on a large single curvature base surface, Figure 5-16. No additional shear occurred on this surface. The maximum and minimum shear angles observed in Run #R1 were 31° and 28° respectively and were caused by the starting angle between the warp and weft yarns.

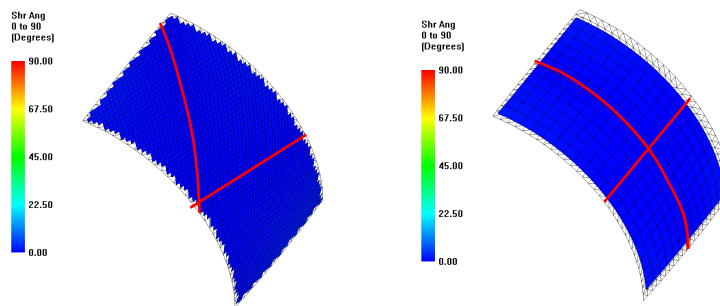


Figure 5-16: Runs #R1 and #R2 draping simulation results

5.5.2 Runs #R3 and #R4

In run #R3 shear was distributed on opposite ends of the double curvature base surface, Figure 5-17. In similar fashion in run #R4, shear reached a maximum at locations on the surface furthest from the starting position. Maximum shear angles of 32° were reached in #R3 and 18° in #R4. For #R3, a minimum shear of 19° was observed suggesting that a larger degree of negative shear occurred in this case since the starting angle between the warp and weft was 30° . Simulation #R4 returned a minimum shear angle of -19° .

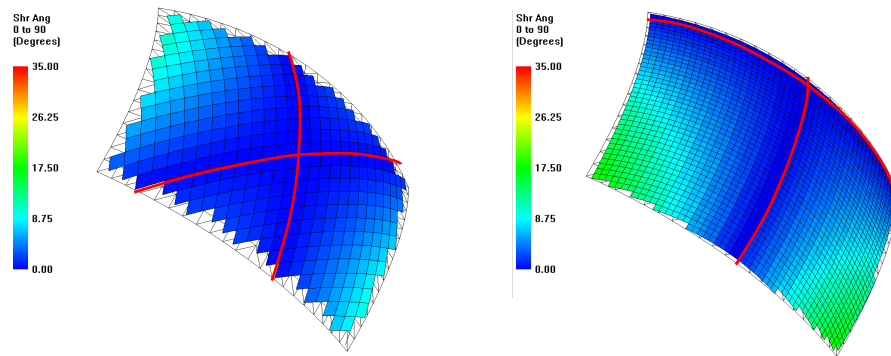


Figure 5-17: Runs #R3 and #R4 draping simulation results

5.5.3 Runs #R5 and #R6

Run #R5 returned a maximum and a minimum shear angles of 6° and -4° , respectively on the double curvature base surface, Figure 5-18. The starting point limited the amount of shear that occurred as shear was more evenly distributed over the surface. Simulation #R6 showed a trend similar to that observed with nearly every double curvature case where the maximum and minimum shear angles occurred at opposite ends of the base surface at points furthest away from the starting point. The maximum and minimum shear angle are 32° and 21° respectively, suggesting a larger amount of negative shear.

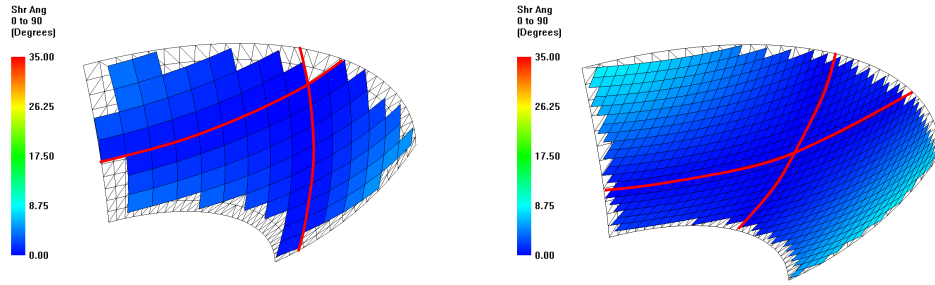


Figure 5-18: Runs #R5 and #R6 draping simulation results

5.5.4 Runs #R7 and #R8

The single curvature base surface contributed no addition to shear in runs #R7 or #R8, Figure 5-19. As observed with all cases so far, the single curvature surface carried the shear induced at the starting angle between warp and weft yarns.

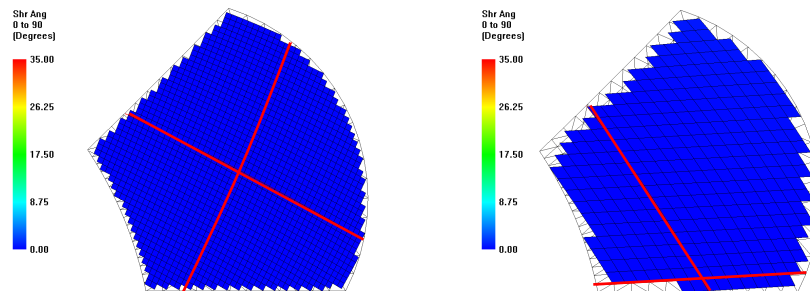


Figure 5-19: Runs #R7 and #R8 draping simulation results

5.5.5 Runs #R9 and #R10

The near flat base surface represented in runs #R9 and #R10 did no contribute any additional shear, Figure 5-20. Any observable shear was due to the initial angle between warp and weft yarns.

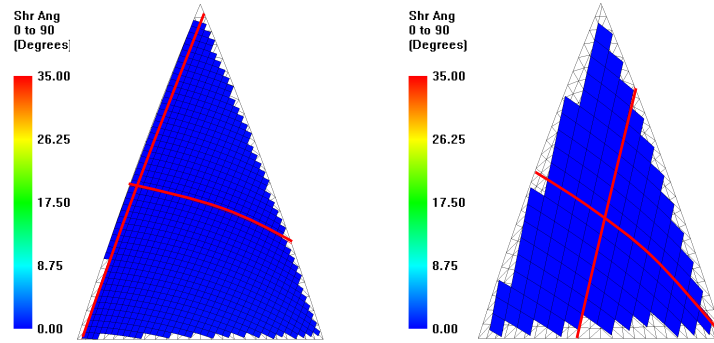


Figure 5-20: Runs #R9 and #R10 draping simulation results

5.5.6 Runs #R11 and #R12

A narrow double curvature base surface was simulated in runs #R11 and #R12, Figure 5-21. As observed in many other runs, low curvature surface, limited the amount of shear over the surface. Simulation #R11 reached maximum and minimum shear angles of 7° and -16° respectively, and simulation #R12 reached maximum and minimum shear angles of 43° and 17° . For #R12, an equal amount of positive and negative shear was observed based on the draping starting point.

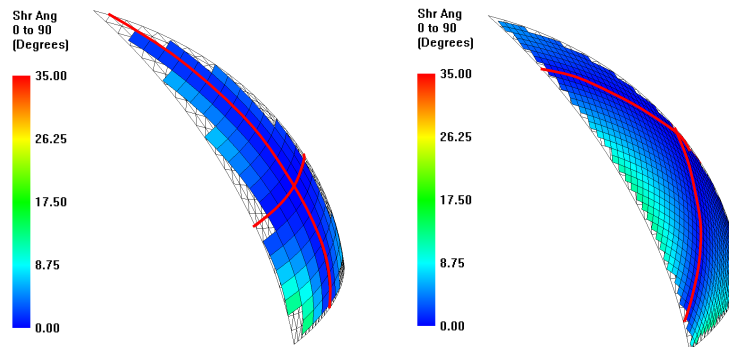


Figure 5-21: Runs #R11 and R12 draping simulation results

5.5.7 Runs #R13 and #R14

A double curvature base surface was featured in runs #R13 and #R14, Figure 5-22. The maximum and minimum shear angles in #R13 were 34° and 22° respectively, suggesting slightly more

negative shear. Maximum and minimum shear angles in #R14 were 13° and -4° respectively. As seen in most double curvature surfaces, maximum or minimum shear occurs at points further from the starting point, and shear is distributed over the entire surface.

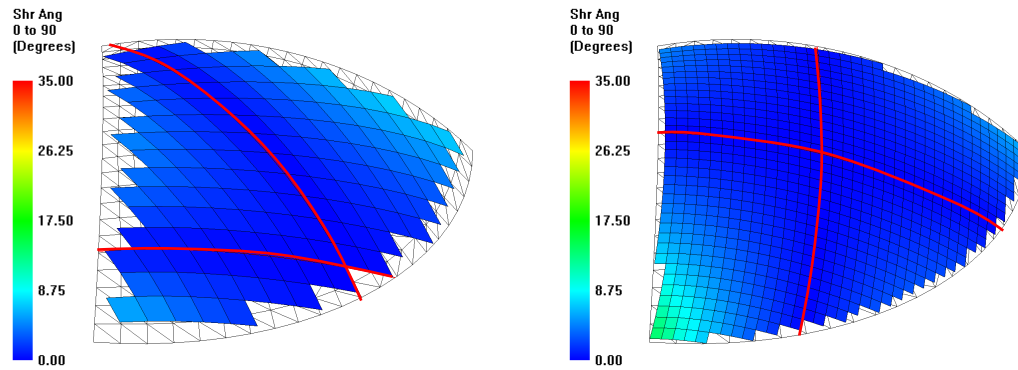


Figure 5-22: Runs #R13 and #R14 draping simulation results

5.5.8 Runs #R15 and #R16

Near flat surfaces simulated in runs #R15 and #R16 further confirmed such surfaces do not contribute to shear, Figure 5-23. These surfaces exhibited no additional shear, as observed with every near flat base surface in these simulation plans, carrying only the preliminary shear induced by the starting angle between the warp and weft.

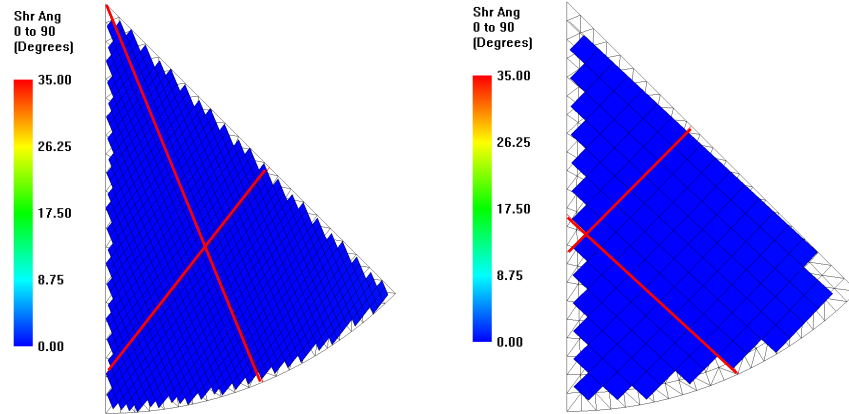


Figure 5-23: Runs #R15 and #R16 draping simulation results

5.6 Discussion

As observed with the basic runs, the reflected runs demonstrated similar behaviour. Near flat and single curvature base surfaces did not contribute any additional shear over the surface but simply carried the initial shear over the surface. Double curvature base surfaces caused shear to occur, typically in greater amounts further away from the starting point. Surfaces with higher curvature also tend to produce a greater amount of shear as there was more curvature. Table 5-6 presents all the shear responses calculated through the new simulations.

Statistical analysis using Scree plots of $SS(C)$ was used in a similar manner for interpreting the results from the fully reflected simulation plan, Figure 5-24. As observed in the basic runs, factor M had the greatest effect on shear, confirming that the initial angle between the warp and weft yarns is a main effect. Knowing that the shear is carried through flat and single curvature base surfaces and added onto double curvature base surfaces is very useful, especially once results begin to be linked when developing strategies for complete parts.

Table 5-6: Fully reflected 16-run Taguchi experiment results

Run	Shear Angle (°)			
	Maximum	Minimum	Average	Absolute Average
#R1	31.33	27.39	29.65	29.65
#R2	1.76	-1.62	0.00	0.55
#R3	32.6	19.02	27.27	27.27
#R4	17.92	-17.93	0.00	5.077
#R5	5.7	-3.84	0.36	2.063
#R6	32.02	21.55	27.78	27.78
#R7	0.17	-0.12	0.00	0.044
#R8	30.03	29.21	29.84	29.85
#R9	1.25	-0.79	0.116	0.31
#R10	30.94	29.57	30.20	30.20
#R11	7.19	-15.65	-0.27	3.42
#R12	43.22	17.13	30.11	30.11
#R13	34.97	22.55	29.06	29.06
#R14	13.44	-4.52	0.64	1.73
#R15	30.08	29.53	29.97	29.97
#R16	0.11	-0.35	-0.03	0.07

As discussed for the basic runs, a second set of Scree plots, where factor *M* was removed revealed a more useful comparison between the remaining factors, Figure 5-25. Factors *D*, *C* and *G* stood out as having a greater effect on maximum and minimum shear angles. Regarding the draping starting point factor *H*, it was observed that shear was distributed over the surfaces differently depending on whether the starting point was at the centre of the base surface or at its edge.

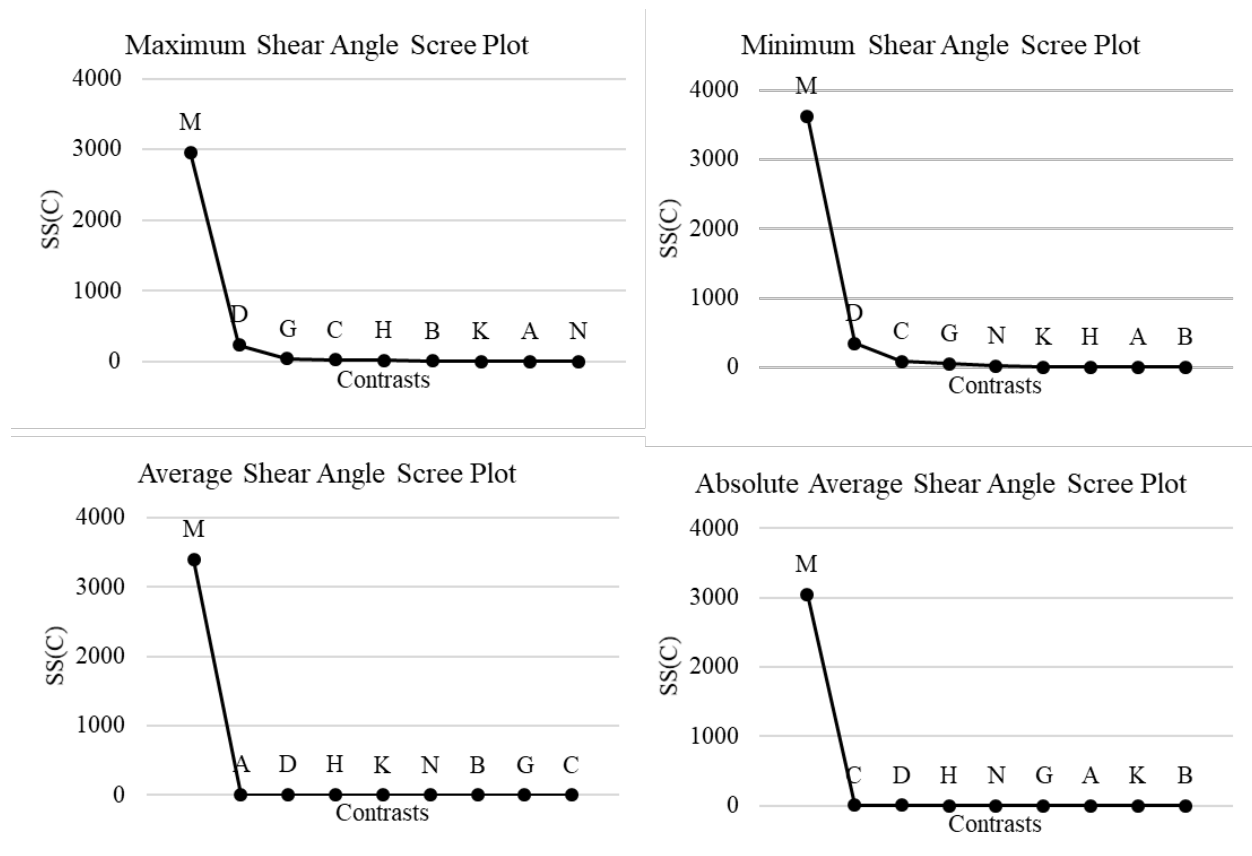


Figure 5-24: Fully reflected 16-run shear angle Scree plots

Relating back to the original results, each of these factors dictated if a base surface was nearly flat, single curvature or double curvature, and was directly connected to the overall amount of curvature for each base surface. The initial simulation results present clear evidence that double curvature base surfaces generate shear, and that near flat and single curvature surfaces do not, as expected. These observations assist in drastically reducing the total number of simulations required for populating the database, which is further discussed in the following section. The factors that appear to have no effect on shear; *A*, *B*, *H*, *K* and *N* would normally all be eliminated from further investigations. However, factors *A* and *B* are both geometric parameters required by the conical frustum to generate base surfaces, thus they are kept in the simulation plans. Factors *H* and *N*, draping starting position and draping starting orientation, may not show importance when

considering individual base surfaces. However, considering the overall draping strategy and linking individual base surface results, orientations and starting positions become important thus these factors as are kept in the simulation plans. Finally factor *K*, fibre spacing, simply presented more refined results from the draping simulations and therefore the minimal effects presented in the Scree plots are an accurate representation of the importance of fibre spacing. Details on the final decision for fibre spacing are presented in Chapter 6.

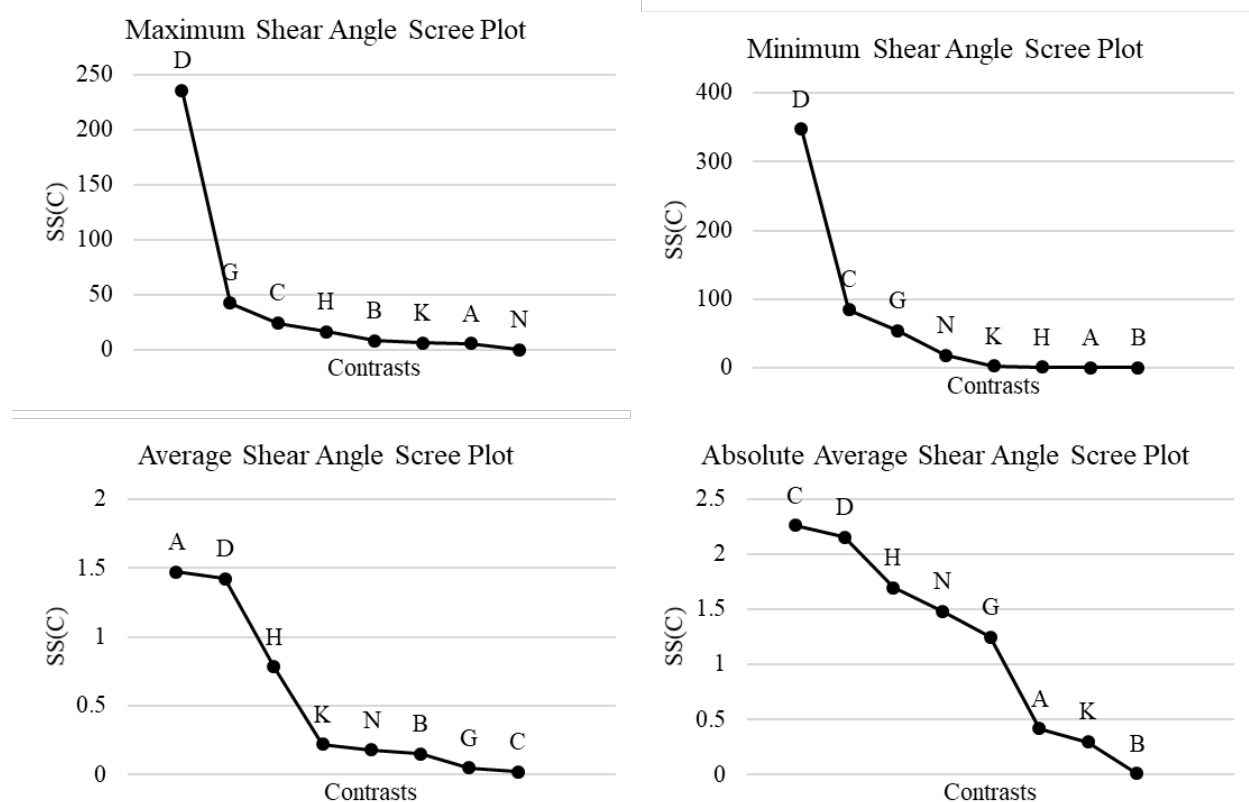


Figure 5-25: Fully reflected 16-run shear angle Scree plots, M removed

6 Database population using the basic conical frustum

6.1 Introduction

For the draping algorithm to probe through draping scenarios quickly, a database of shear results must be generated. Database A was constructed using a near full factorial experimental plan similar to those discussed in Chapter 5. The parameters with the largest effects on in-plane shear were identified as the angle between the warp and weft yarns, followed by geometric parameters C_R , β and $|\vec{L}|$. Shear angle results provide a useful base for populating Database A since shearing during draping can lead to wrinkling, however it alone cannot be relied upon to quantify accurately the behaviour of fabrics as they are being draped thus warp and weft yarn orientations are also extracted from the base surface simulations.

When considering the draping algorithm discussed in Chapter 8, the linking of base surface results cannot be done in a meaningful way when considering only the shear angles. As a mould surface is progressively covered, the orientations of warp and weft yarns along the edges of individual base surfaces will change, depending on the yarn starting orientations, base surface types and the amount of shear caused by the different base surfaces. Yarn orientations along the edges of a given base surface will be carried into adjacent base surfaces and will directly influence the shear behaviour over these adjacent base surfaces. If yarn orientations were not considered when moving into adjacent base surfaces, orientation angles along the base surface edges would essentially be reset for each base surface sequentially encountered in a single strategy. In reality, yarn orientations along base surface edges continue into adjacent base surface as the full mould surface is draped, continuing along the same angles. Therefore, incorporating yarn orientation angles in Database A is essential to probing multiple draping scenarios.

Direction vector files (.mdv) were exported using Nottingham DrapeIt, which provide warp and weft yarn orientation vectors for each element on the simulated base surfaces. The vectors are useful for calculating warp and weft yarn orientation angles along the edges of the base surfaces. Since the edges link the base surfaces, orientations of the yarns with respect to the connecting edges enable the draping algorithm to connect base surface results through smooth transitions. The warp and weft orientation angles are also directly related to in-plane shear angles, which remain important in developing draping strategies. The process in which the yarn orientation angles and shear angles are calculated from warp and weft orientation vectors is found in Appendix C.

6.2 Database population

A full factorial experimental plan featuring all combinations of modalities for the conical frustum geometric parameters and draping parameters would involve $2^{16} = 65536$ simulations. Considering that performing this number of simulations would take a significant amount of time, and that results from the experimental simulation plans using the five-parameter conical frustum are available, the five parameter conical frustum was used as a starting point to begin populating Database A.

The majority of base surfaces encountered in practice can be represented using the core conical frustum parameters based on the original mould decomposition presented in Section 4.1. For draping parameters, fibre spacing was set at a constant value for database population that allowed for good surface coverage without a significant increase of the time needed for the simulations to run.

The full factorial simulation plan was reduced down to $2^8 = 256$ simulations needed for populating the database making the simulation and data extraction procedure more manageable. Database population is a continuous process and since it is designed using Taguchi method and factorial plans, additional results can easily be added to the existing database. Results obtained from the simulation plans used in Chapter 6 indicated that near flat and single curvature base surfaces do not contribute to shear, hence results such surfaces are already known. Therefore, simulations on these surfaces do not need to be performed further reducing the total number of simulations needed down to 128. Upon completing each simulation in Nottingham DrapeIt, data was extracted from .mdv files, organized and compiled into MS Excel. The procedure for running draping simulations is detailed in Appendix B and instructions for processing the .mdv files can be found in Appendix C.

6.3 Developing linear functions of results

Since Database A was populated using Taguchi methods, linear functions can be developed from the results. Each linear function represents an individual result; maximum, minimum or average, of shear angle, warp orientation angle or weft orientation angle along a single base surface edge. These functions enable linear interpolations between results to be accessed for base surfaces defined by parameter values that lie between the minimum and maximum modalities outlined in Chapter 6. Although the sample space for the functions is constrained and delimited by the input parameters, the functions are very useful, and they will become more refined as the database is further populated.

In practical use within the probing of draping strategies the relevant linear function can easily be selected based on the base surface type, and on which edge is being interrogated. For example, a double curvature base surface is identified and given conical frustum vertex labels *A*, *B*, *C* and *D* as discussed in Chapter 3, Figure 6-1. The edges are labelled based on the end points connected to each edge: *AB*, *BD*, *DC* and *CA*. A linear equation can be identified for a specific result (maximum, minimum or average shear angle, warp orientation or weft orientation) along any of the edges for a base surface that can be created within the maximum and minimum modalities of the core conical frustum geometric parameters, that generates a four-edge base surface. Similarly, linear functions were developed for each output variable and edge of base surfaces with three edges: *AD*, *DC*, and *CA*.

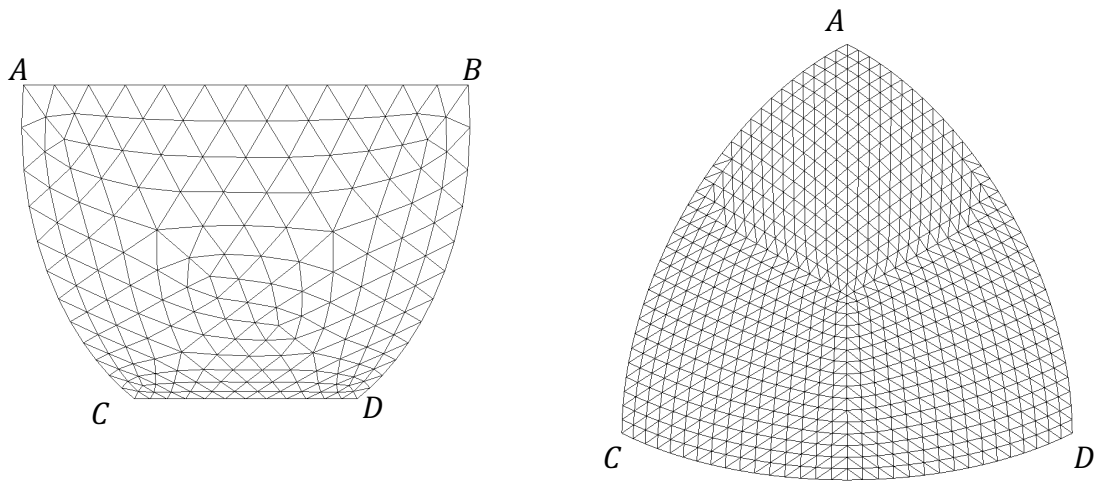


Figure 6-1: Four-edge base surface and three-edge base surface

A line plot for the average shear angle along edge *DC*, for a base surface with three edges, is presented in Figure 6-2. For the line upcoming line plots, the horizontal axis is each simulation run performed during database population and the vertical axis is the shear angle results for each run. The peaks and valleys in the line plot are due to the angle between the warp and weft, yarns adding

30° of shear to some of the results based on the experimental simulation plan used for populating the database, which include parameter M discussed in Section 5.2.1. Plots similar to this were made for each of the database results; maximum, minimum and average shear angle, warp orientation angle and weft orientation angle, for all individual edges on base surfaces featuring three edges and four edges.

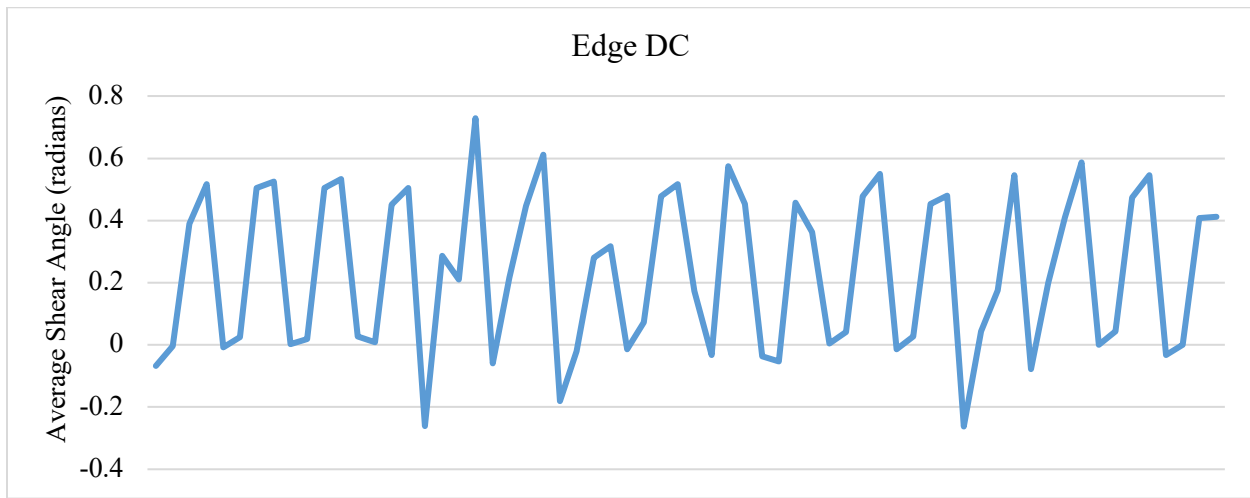


Figure 6-2: Average shear angle database results for a three edge base surface

To build the linear functions, responses were calculated for each set of individual geometric and draping parameters, as defined by their specific modalities for each simulation. Using edge DC for each base surface with three edges and specifically looking at the average shear angle along that edge, results from each run were used for calculating specific responses for each parameter. This was achieved by multiplying the result from the simulation to each modality specific to that particular simulation. Table 6-1 shows the first three simulations, the parameter modalities and their respective responses. The average for the responses was calculated for each parameter and used as a constant in the linear function. The linear function uses a sum of these averages, multiplied by their respective parameter modalities, to produce the results found in the database.

Table 6-1: Edge DC simulation results, modalities and responses

Edge DC Average Shear Angle Results			Run Specific Parameter Modalities								Run Specific Parameter Responses (radians)							
Sim.	Radians	Degrees	A	B	C	D	G	H	M	N	A	B	C	D	G	H	M	N
1	-0.07	-3.87	-1	-1	-1	-1	-1	1	-1	-1	0.07	0.07	0.07	0.07	0.07	-0.07	0.07	0.07
2	0.00	-0.19	-1	-1	-1	-1	-1	1	-1	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.39	22.25	-1	-1	-1	-1	-1	1	1	-1	-0.39	-0.39	-0.39	-0.39	-0.39	0.39	0.39	-0.39

The first linear function used only the main effects of the parameters in the simulation plan, causing the function to calculate coarsely inaccurate results when compared to those found in the database, Figure 6-3. This occurred because interactions between the parameters were not considered in this particular iteration of the functions. The addition of constants to the linear functions that quantify the effect of interactions between the parameters lead to better defined and more accurate functions. The interactions are defined as the following:

- Interactions of two parameters: *AB, AB, AG, AH, AM, AN, BC, BG, BH, BM, BN, CG, CH, CM, CN, GH, GM, GN, HM, HN, and MN*;
- Interactions of three parameters : *ABC, ABG, ABH, ABM, ABN, ACG, ACH, ACM, ACN, AGH, AGM, AGN, AHM, AHN, BCG, BCH, BCM, BCN, BGH, BGM, BGN, BHM, BHN, CGH, CGM, CGN, CHM, CHN, CMN, GHM, GHN, GMN, and HMN*;
- Interactions of four parameters: *ABCG, ABCH, ABCM, ABCN, ABGH, ABGM, ABGN, ABHM, ABHN, ACGH, ACGM, ACGN, ACHM, ACHN, ACMN, AGHM, AGHN, AGMN, AHMN, BCGH, BCGM, BCGN, BCHM, BCHN, BGMN, BGHM, BGHN, BGMN, CGHM, CGHN, CGMN, CHMN and GHMN*;
- Interactions of five parameters: *ABCGH, ABCGM, ABCGN, ABCHM, ABCHN, ABCMN, ABGHM, ABGHN, ABGMN, ABHMN, ACGHM, ACGHN, ACGMN, ACHMN, AGHMN, BCGHM, BCGHN, BCGMN, BCHMN, BGHMN, and CGHMN*;

- Interaction of six parameters: $ABCGHM$, $ABCGHN$, $ABCGMN$, $ABCHMN$, $ABGHMN$, $ACGHMN$, and $BCGHMN$;
- Interaction of seven parameters: $ABCGHMN$.

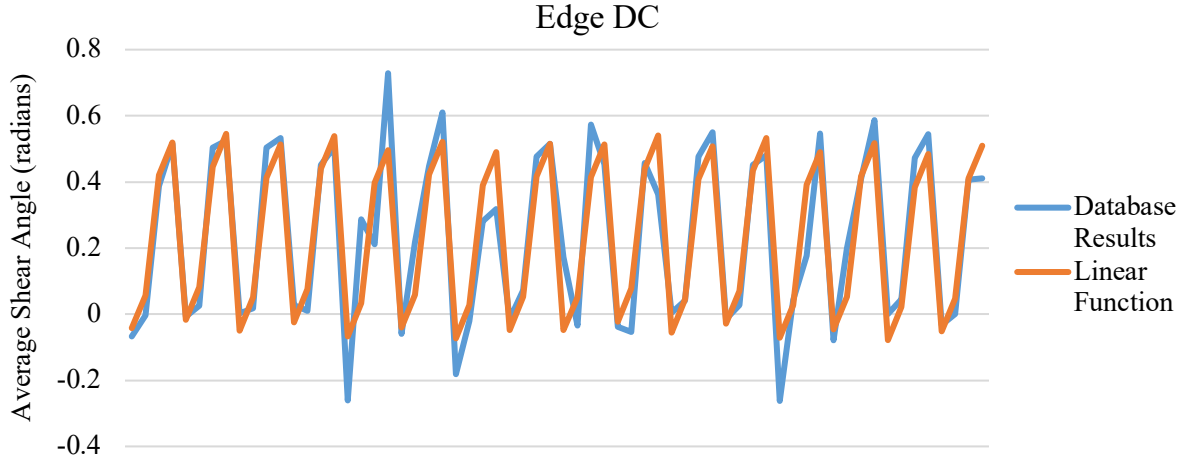


Figure 6-3: Average shear angle results and linear function (main effects only)

As the effects of interactions were added to the linear function, changes became progressively more difficult to observe due to factor M dominating the results, Figure 6-4. Changes in the function beyond interactions of three parameters become unobservable, which can be attributed to the increasing accuracy of the linear functions.

In order to better observe the effect of adding the interactions between parameters to the function, factor M was removed, Figure 6-5 a), making the difference between the simulation results and linear functions become more prominent. As interactions effects were added, the linear functions became more refined, and completely aligned with the simulation results which can be seen in Figure 6-5 b) – f). It is important to note that since factor M was removed for this later comparison, the maximum number of parameter interactions only reach a total of six rather than seven as above; however, a clear increase in linear function accuracy can still be observed. For the final linear

functions to be used in the algorithm, factor M is included since the contributions of shear are sequentially added as a draping strategy is developed. All functions for all other results and all other edges were developed in a similar manner. For each edge of the base surfaces, there are nine individual results, each represented by a different linear function. Some base surfaces feature three edges and some feature four edges, leading to a total of 63 linear functions for Database A. All linear functions take the same, uniform basic form. What changes from one function to the next is the coefficients of the functions, which are all obtained by performing simulations planned through Taguchi designs. It was decided that information at specific points along each edge were more important than the average, minimum and maximum values. Since Database A contains information at these points, functions for the end points and midpoint of each base surface were created as well, following the same procedure as the other linear functions. Chapters 9 and 10 demonstrate why specific locations are more useful than averages, maximums and minimums.

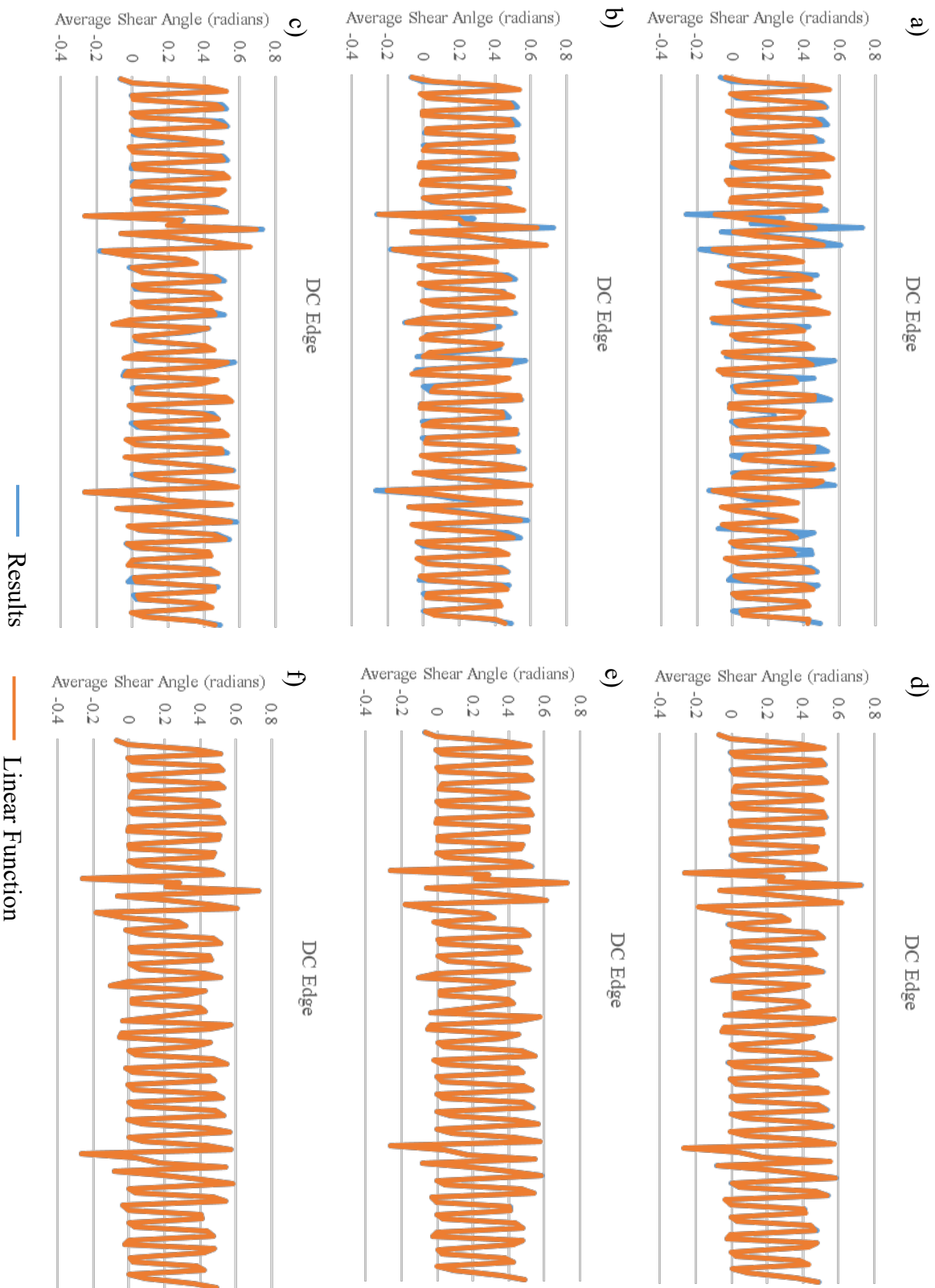


Figure 6-4: Line plot comparison: a) interactions of 2; b) interactions of 3; c) interactions of 4 d) interactions of 5; e) interactions of 6 f) interactions of 7 parameters

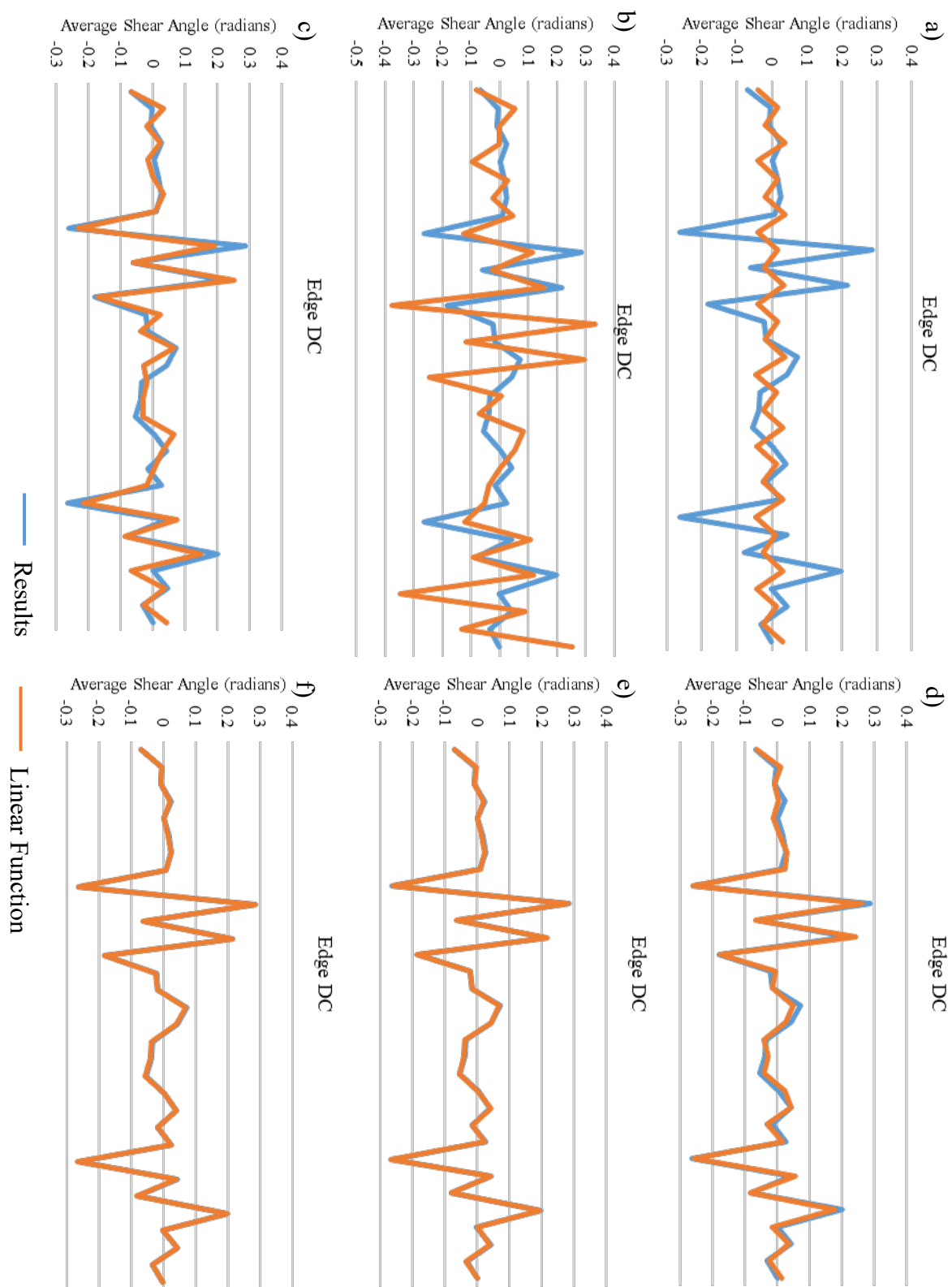


Figure 6-5: Line plot comparison: a) Main effects; b) interactions of 2; c) interactions of 3 d) interactions of 4; e) interactions of 5 f) interactions of 6 parameters

7 Unique in-plane shear occurrences

Upon populating Database A, shear behaviour through transitions along common edges between adjacent base surface types was investigated so that a better understanding of connections between surfaces could be established. It was known from previously reported simulations, Chapter 6, that flat and single curvature base surfaces do not cause any additional shear to fabrics, hence linking the results across these surfaces types causes no additional shear. However, upon introducing double curvature surfaces, the behaviour through simple transitions across common edges between base surfaces is disrupted with fibre orientations and shear angles. These changes have a direct influence on the fabrics behaviour when linking results between surfaces.

7.1 Shear progression

7.1.1 Defining shear progression

Results obtained for draping a double curvature base surface show a varying yarn orientations and shear angles along the surface edges. These changes are caused by the fabric shearing over the surface so that it can be covered without issues. Considering a transition into another adjacent surface, along the common edge the warp and weft yarn orientations will vary between the two end points of that edge, and that variation will be carried from the initial surface into the adjacent surface. Single curvature base surfaces are often adjacent to double curvature base surfaces and do not affect in-plane shear, on their own. Nevertheless, when linking base surface results moving into a single curvature base surface from a double curvature base surface, varying shear angle and yarn orientations will lead to a continued change of shear angles and fibre orientation into the adjacent single curvature surface. This phenomenon is labelled as shear progression. Shear

propagates or progresses into the single curvature surface until a maximum shear angle,, normally the shear limit of the fabric, is reached (Figure 7-1). Shear progression can occur when transitioning from:

- Double curvature sphere configuration into flat base surfaces;
- Double curvature saddle configuration into flat base surfaces;
- Double curvature sphere configuration into single curvature base surfaces;
- Double curvature saddle configuration into single curvature base surfaces;
- Double curvature sphere configuration into double curvature saddle configuration base surfaces;
- Double curvature saddle configuration into double curvature sphere configuration base surfaces.

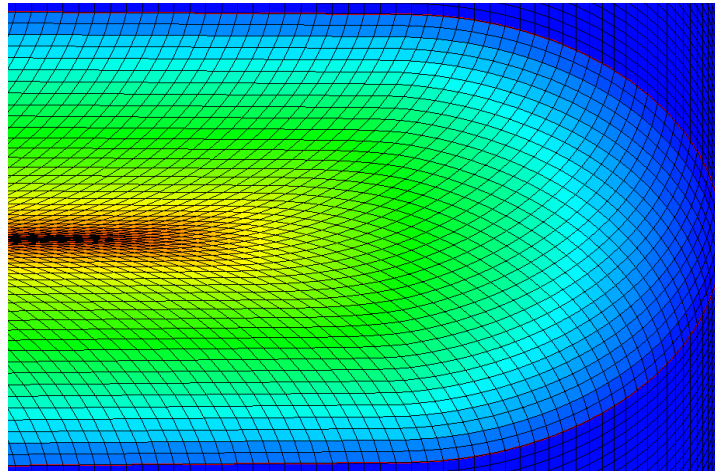


Figure 7-1: Shear progression

The limit of shear progression is defined mainly defined by the degree of curvature over the double curvature surface and the angle between surfaces, that latter of which is discussed in Section 7.3. The following sections use the results available from draping simulations, warp and weft

orientation vectors and shear angles to develop an analysis method for shear progression using vector projection, and head-to-tail vector summation.

7.1.2 Vector projection

In order for shear progression to be considered by the algorithm when probing draping strategies, a prediction method was developed and is reported in this section, enabling shear progression occurrences to be noted, a maximum shear angle to be determined and a location of maximum shear angle can be predicted.

Consider as an example a flat base surface with varying yarn orientations and shear angles along a single edge. The orientation angles are calculated using warp and weft orientation vectors, which are extracted from DrapeIt draping simulations. Upon examining warp yarn orientations along the edge and comparing them with those of a given yarn progressing through the surface, it is observed that the latter undergoes the same changes in orientation as do the different warp yarns along the common edge at their respective intersections with that edge. Each change of orientation along the common edge is projected into the parallel yarns over the adjacent surface. Since no additional shear occurs due to the surface being flat in this case, no additional orientation change occurs along the remaining edges, or over the surface. From the instance that the warp yarns change along the left side of Figure 7-2, it is can be observed that the weft yarns remain parallel below this starting warp yarn over the flat surface.

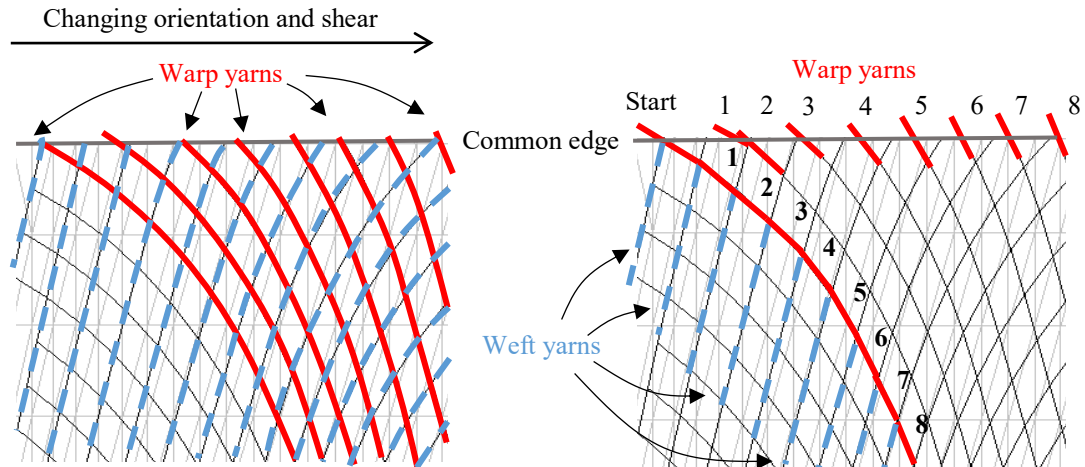


Figure 7-2: Projected orientation change on a flat base surface

In the case of moving from a double curvature to a single curvature surface, similar changes in warp and weft orientations are present. Due to surface topography, single curvature surfaces do not induce any additional shear, so yarn orientation remain parallel upon entering the single curvature from the double curvature surface as observed over flat surfaces, Figure 7-3.

Ultimately, results in Database A provide warp yarn orientations and weft yarn orientation vectors for all individual elements along all edges of the base surfaces. Warp and weft vectors for each element along the edge are used for projecting changes in orientations of yarns as they progress onto the surface, therefore enabling predictions of magnitude and location of maximum shear caused by shear progression. Considering the generalized case of moving from a double curvature surface to single curvature surfaces and observing only the warp yarns as they moving first along the common edge and second into the adjacent single curvature surface Figure 7-3. At each element along the common edge, the warp yarn vectors express a change in orientation. Each of these changes can be projected into the single curvature surface, at each instance where it does not change. These are labelled vector progression paths which are visualisation aids for this process,

Figure 7-3. Along these individual vector progression paths, the warp yarn vectors do not change as a result of the weft vectors “below” them, remaining parallel to one another. Similar vector progression paths can be observed when considering weft orientations, as a result of warp orientations remaining parallel.

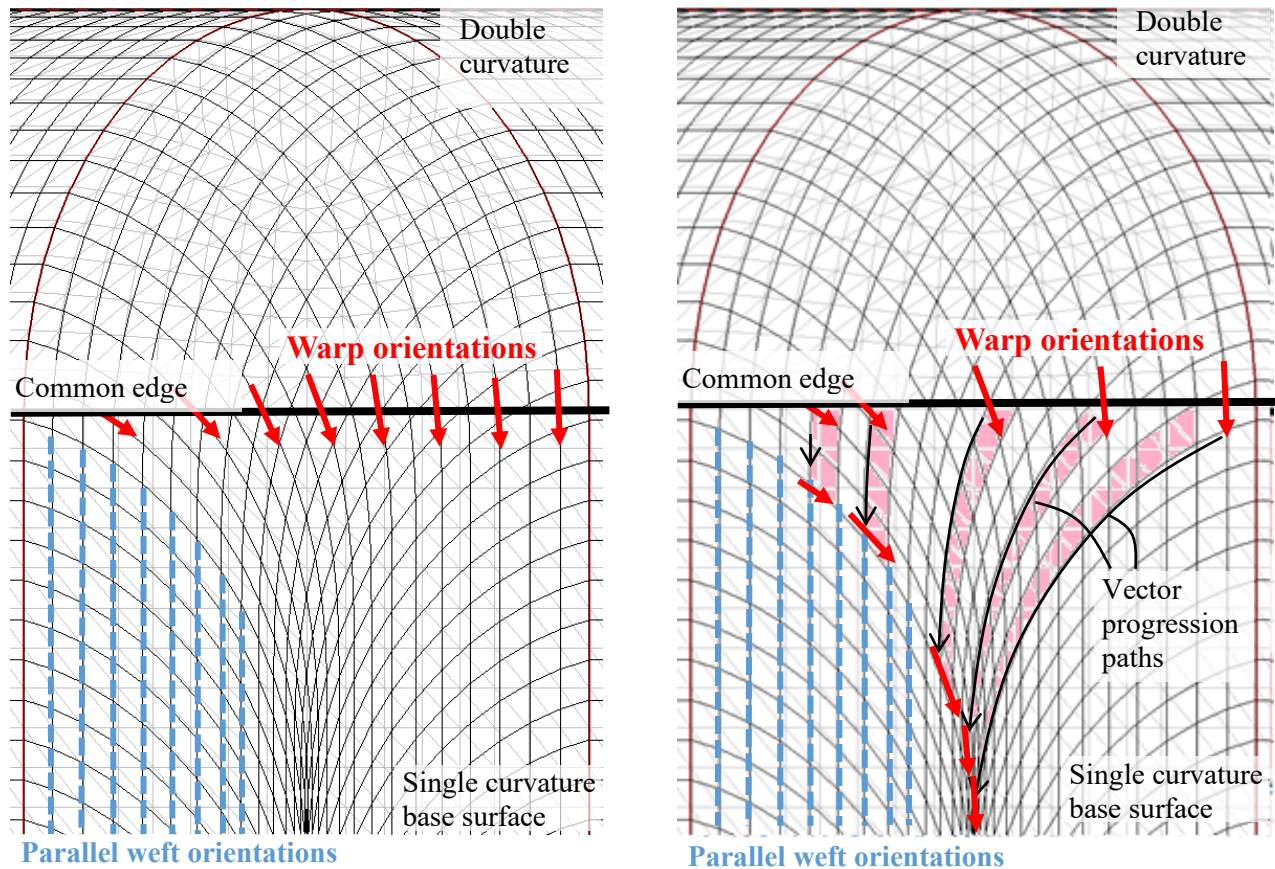


Figure 7-3: Shear progression: changing yarn orientation (left) vector progression paths (right)

The maximum shear angle can be determined by considering that the warp orientation vector $\overrightarrow{u_{warp}}$ and weft orientation vector $\overrightarrow{u_{weft}}$ at opposite ends of the common edge ultimately meet at a point over the surface base surface, Figure 7-4. These yarn orientations result from changes in orientations observed across the common edge, and therefore the angle between these vectors

denotes the maximum shear angle, Figure 7-3. This is calculated using Equation 16, substituting $\overrightarrow{u_{warp}}$ and $\overrightarrow{u_{weft}}$ for $\overrightarrow{R_1}$ and $\overrightarrow{r_1}$.

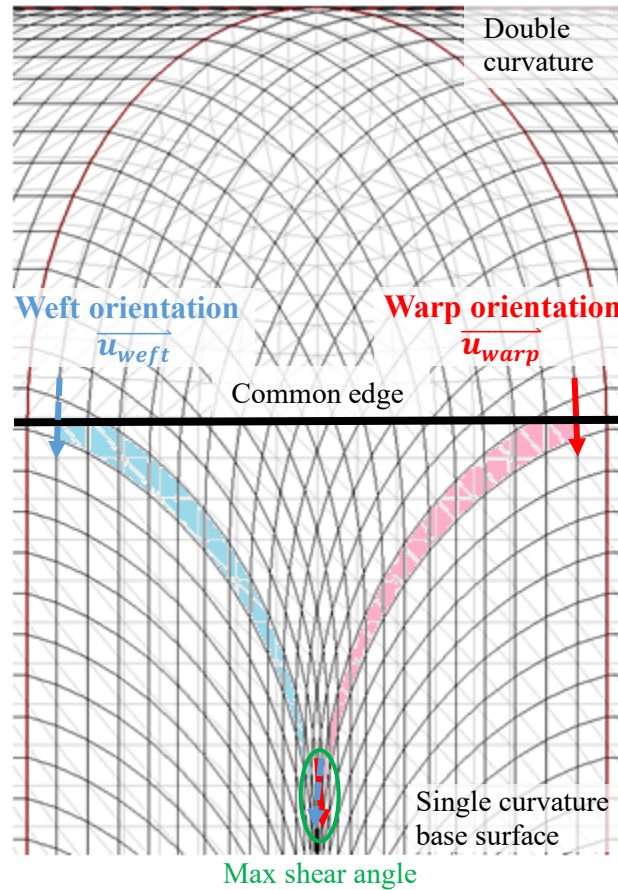


Figure 7-4: Max shear angle

To determine the location of maximum shear, a head-to-tail summation of warp or weft vectors is used. The calculation begins along the common edge at the element with the least amount of shear and with the warp orientation vector nearest parallel to the common edge, Figure 7-5. The head-to-tail summation simply adds the orientation vectors in sequence as they appear along the common edge to the end of the common edge. The result of this summation is a single vector pointing to the location of maximum shear from the starting element. Alternatively, weft orientation vectors can be used.

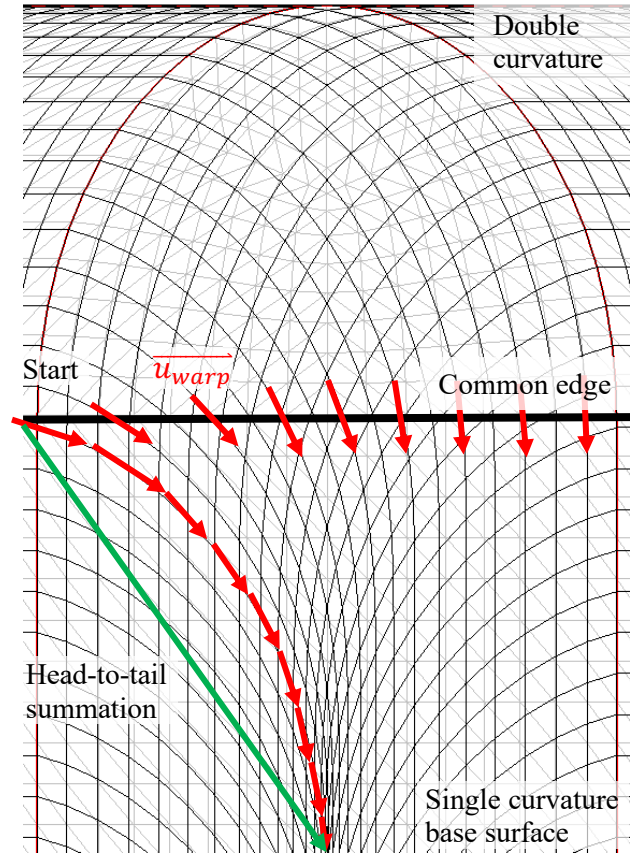


Figure 7-5: Head-to-tail vector summation

In reality, maximum shear will often occur much closer to the common edge than the vector projection predicts, since the physical locking angle is typically much lower for a real reinforcement fabric. The head-to-tail summation procedure is informative, nevertheless for cases where the maximum shear angle is less than the physical locking angle of the fabric being draped, in cases similar to that present in Figure 7-5 which has different orientation angles entering a double curvature. The procedure is also useful in cases where it will predict the locking of fabric, as this would determine the extent and end of a physical pattern that could be draped over the preform.

7.1.2.1 Alternative head-to-tail summation

A similar procedure was used for predicting more accurately where the maximum shear angle will occur. Rather than using exclusively the vectors for predicting the location, a combination of vectors and shear angles are used in this procedure. The original procedure did not consider the contribution to shear caused by the change in warp and weft orientation vectors along the common edge.

Recall from the previous section that warp vectors are projected from along the common edge into the adjacent single curvature base surface, Figure 7-3. Each vector projected onto the adjacent surface is considered a step in shear progression. It was also observed that the weft vectors following “below” the projected warp vectors remain parallel to each other. Shear progression shear angles on the single curvature surface for each step is calculated using changing warp orientation vectors and constant weft orientation vectors following the first vector progression path, Figure 7-6. Shear is also calculated between the warp and weft vectors along the common edge.

In order to consider the contribution from both the shear progression and the originally unaccounted for varying shear angle along the common edge, the difference between the two sets of shear angles is taken. Warp vectors along the common edge are matched with projected warp vectors along the vector progression path at each step of shear progression. The shear angle values that have matching warp vectors are subtracted from one another. This difference quantifies the amount of shear unaccounted for in the original head-to-tail vector summation. These differences are added back to the shear along the weft vector progression path. Head-to-tail vector summation is performed following the same steps as the previous procedure, taking the newly calculated shear angles into consideration. A maximum shear angle equal to 90° or to a defined maximum locking

angle will be encountered during this summation, leading to the identification of a location on the base surface where that angle can be found. This location is always quantified by a vector pointing into the adjacent surface from the common edge. An example of this calculation is presented in section 7.1.3. Shear progression is a common occurrence that will be encountered during draping and must be considered when probing draping strategies.

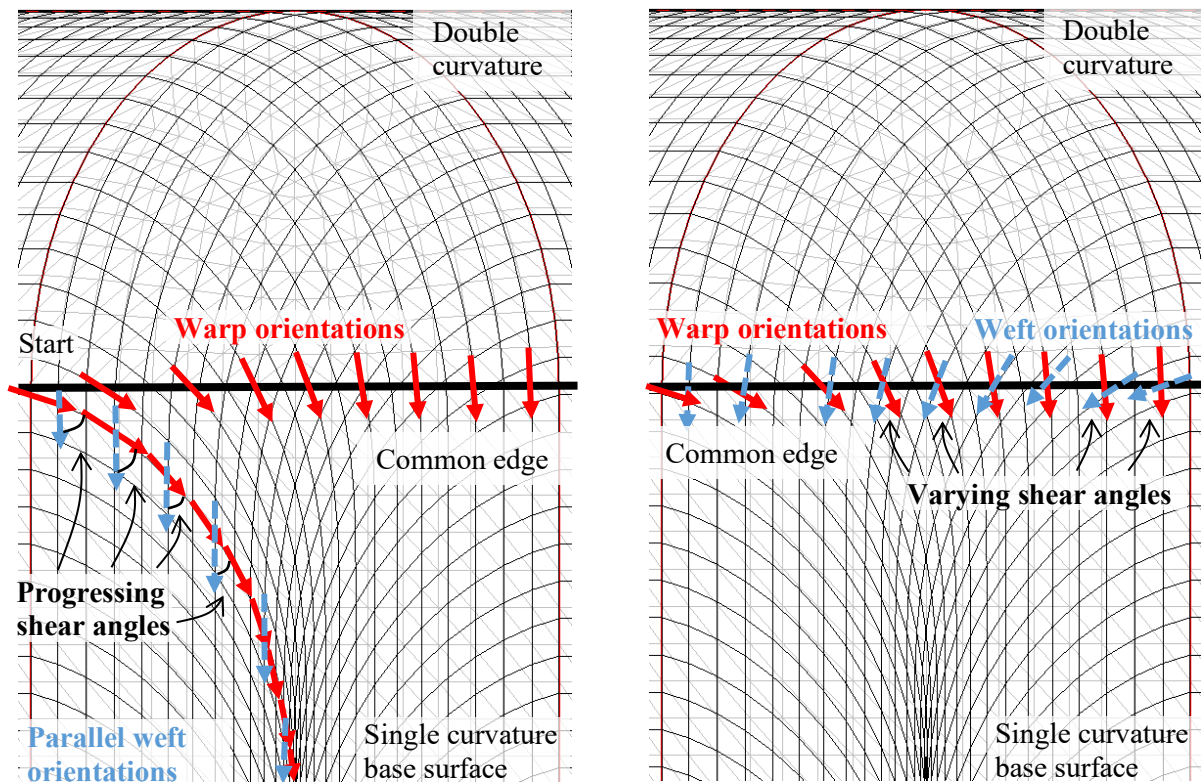


Figure 7-6: Progressing shear angles (left), varying shear angle along common edge (right)

7.1.3 Head-to-Tail Summation Example

In this example, two base surfaces defined by conical frustum parameters are found in Table 7-1. Calculations for the maximum shear angle and location proceed as follows. The second, more accurate head-to-tail vector summation technique is used for predicting the location of the maximum shear angle.

Table 7-1: Shear progression example: Core conical frustum parameters

Core Conical Frustum Parameters	Double Curvature	Single Curvature
$ \vec{R}_1 $	2 cm	2 cm
$ \vec{R}_2 $	4 cm	2 cm
$ \vec{L} $	2 cm	10 cm
β	90°	90°
C_R	2 cm	∞

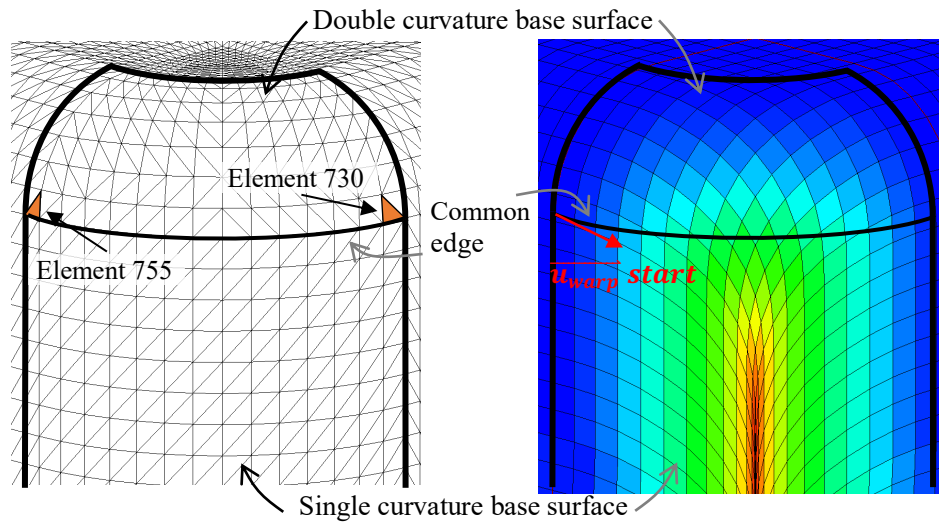


Figure 7-7: Single and double curvature model (left) shear progression (right)

For this example, element numbers are sequenced as presented in Table 7-2 from left to right along the common edge; the warp vector for element 755 is nearly parallel to the common edge. This is the starting vector for the head-to-tail summation.

Using the warp vector at element 730 and the weft vector at element 755, the maximum shear angle can be calculated using Equations 16 and 17. These vectors are chosen since they are the last instance of change for each yarn orientation across the common edge, as shown in Figure 7-8. The maximum shear angle is equal to 86°.

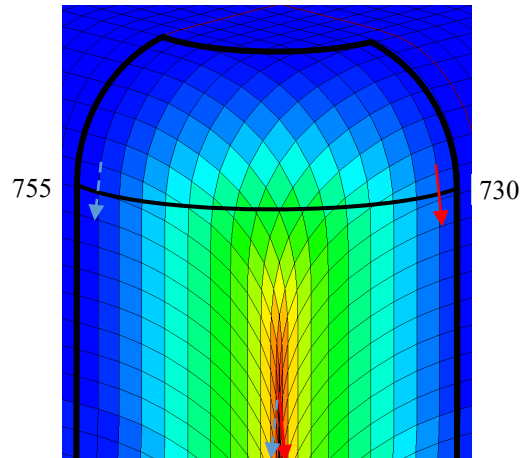


Figure 7-8: Maximum shear angle vectors

Table 7-2: Warp and weft yarn orientation vectors

Element Number	Warp Vectors			Weft Vectors			Shear Angle (°)
	x	y	z	x	y	z	
755	0.05	0.00	-0.50	-0.02	0.50	-0.06	7.02
753	0.05	0.00	-0.50	-0.07	0.48	-0.12	13.59
751	0.05	-0.01	-0.50	-0.11	0.44	-0.20	21.86
749	0.05	0.00	-0.50	-0.14	0.40	-0.26	29.38
747	0.05	0.00	-0.50	-0.16	0.35	-0.31	35.58
745	0.09	-0.06	-0.49	-0.14	0.26	-0.40	41.82
743	0.16	-0.12	-0.46	-0.10	0.17	-0.46	44.18
742	0.20	-0.13	-0.44	-0.09	0.13	-0.47	43.44
740	0.28	-0.16	-0.38	-0.06	0.09	-0.49	38.71
738	0.35	-0.16	-0.31	0.00	0.05	-0.50	35.50
736	0.41	-0.14	-0.25	-0.01	0.05	-0.50	27.56
734	0.45	-0.11	-0.19	0.00	0.05	-0.50	20.13
732	0.48	-0.07	-0.12	0.00	0.05	-0.50	13.55
730	0.50	-0.02	-0.06	0.00	0.05	-0.50	6.98

Next, shear angles within the single curvature surface are calculated using the warp vectors along the common edge and the weft vector at element 755. The resulting shear angles are presented in Table 7-3. The table lists the elements from which the vectors were taken at each step of shear progression, and the shear angle calculated between the vectors.

Once progressing shear angles are calculated, the difference between the shear angle along the common edge and the progressing shear angles is calculated, Error! Reference source not found.. This difference, as mentioned in the last section is the shear angle attributed to the varying warp and weft orientation angles along the common edge. The difference is then added back to the progressing shear, and head-to-tail vector summation is used for determining the location of maximum shear. Head-to-tail summation always adds the same orientation vector, warp in this case, to the next vector in sequence. That sum is then added to the following vector until the maximum shear angle or 90° is reached, Table 7-5, identifying the vector and therefore location of the maximum shear angle caused by shear progression.

Table 7-3: Increasing shear angle into single curvature surface

Element Number		Progressing Shear Angle (°)
Warp Vector	Weft Vector	
755	755	7.02
753	755	13.57
751	755	22.96
749	755	29.82
747	755	36.19
745	755	50.57
743	755	63.10
742	755	67.53
740	755	73.86
738	755	81.53
736	755	81.35
734	755	81.58
732	755	82.29
730	755	82.08

Table 7-4: Shear progression difference

Element Number	Warp Vectors			Shear Angle Along Common Edge (°)	Progressing Shear Angle (°)	Difference (°)
	x	y	z			
755	0.05	0.00	-0.50	7.02	7.02	0.00
753	0.05	0.00	-0.50	13.59	13.57	-0.03
751	0.05	-0.01	-0.50	21.86	22.96	1.10
749	0.05	0.00	-0.50	29.38	29.82	0.44
747	0.05	0.00	-0.50	35.58	36.19	0.62
745	0.09	-0.06	-0.49	41.82	50.57	8.74
743	0.16	-0.12	-0.46	44.18	63.10	18.91
742	0.20	-0.13	-0.44	43.44	67.53	24.09
740	0.28	-0.16	-0.38	38.71	73.86	35.15
738	0.35	-0.16	-0.31	35.50	81.53	46.03
736	0.41	-0.14	-0.25	27.56	81.35	53.80
734	0.45	-0.11	-0.19	20.13	81.58	61.45
732	0.48	-0.07	-0.12	13.55	82.29	68.74
730	0.50	-0.02	-0.06	6.98	82.08	75.11

Table 7-5: Maximum shear angle and head-to-tail vector summation results

Head-to-tail Vector Sum			Shear Angle at Vector Sum (°)
x	y	z	
0.05	0.00	-0.50	7.02
0.10	0.00	-1.00	13.54
0.15	0.00	-1.49	24.06
0.20	0.00	-1.99	30.26
0.25	-0.01	-2.49	36.81
0.35	-0.07	-2.97	59.31
0.50	-0.19	-3.43	82.01
0.70	-0.32	-3.87	91.62

7.2 Effects of double curvature configurations

One of the main differences between spherical and saddle configurations is the method in which shearing occurs. Recall from Section 5.3, positive shear is when the angle between the warp and weft yarns becomes less than 90° , and negative shear is when the angle between the warp and weft yarns becomes greater than 90° ; and shear is always measured from warp to weft. The shear induced by saddle double curvature base surfaces tends to counteract to shear induced by spherical

double curvature. This leads to interesting shearing results when linking surfaces of these two configurations. When near to or in direct contact, the opposing configuration tends to counteract any shear that occurs on adjacent surface, Figure 7-9.

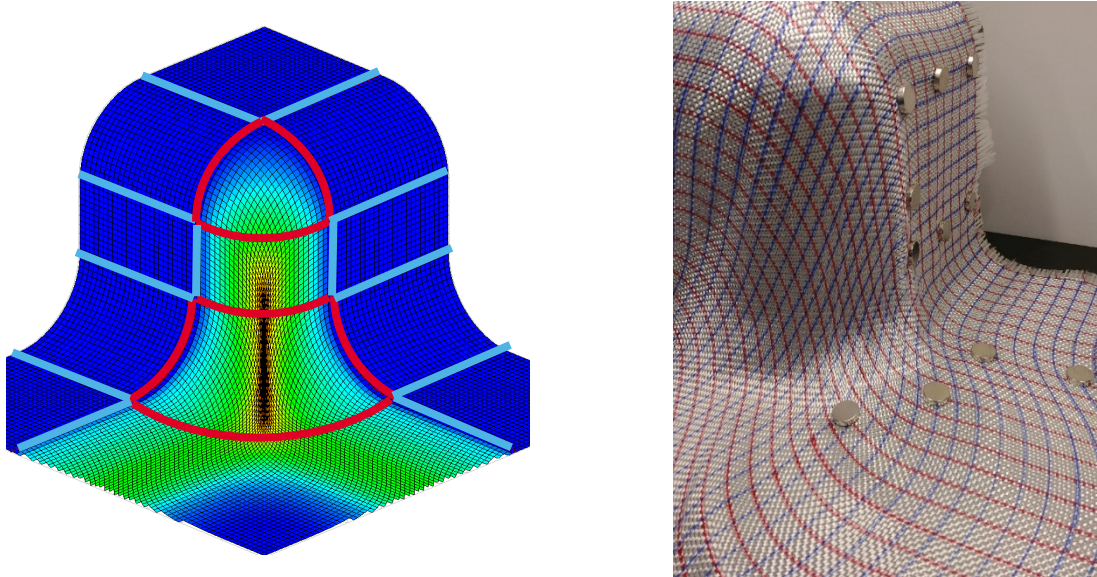


Figure 7-9: Draping double curvature configurations simulation (left), physical surface (right)

The fabric is draped starting with no shear on the flat surface at the top of this feature. Shear is observed to progress into the adjacent single curvature surface upon exiting the sphere configuration double curvature, as expected. The fabric reaches a maximum shear angle over the single curvature surface which is carried by the single curvature surface into the saddle configuration, at which point shear effectively reverses and reduces. The fabric continues to shear negatively upon exiting the saddle configuration, due to shear progression, until it reaches the original starting state, with no shear. This feature is directly attributed to the shearing behaviour featured on these surface types. This effect presents interesting insight on areas on moulds that may assist in covering more area using fewer patterns.

7.3 Angle between surfaces

Previous chapters in this thesis focused mainly on smooth edge and corner features. Sharp edges and corners are avoided in design as they can become problematic during draping, in terms of conforming the fabric to the feature. These features often force the use of darting, allowing the surface to be covered without wrinkling but increasing production time since cuts have to be made through the patterns. The effects of the angle between flat surfaces, as well as proposed solutions for mitigating wrinkling and eliminating the need for darting, are presented in the following section in the form of three cases; a sharp corner without chamfers, a single edge chamfer at a corner, and two edges chamfered at a corner. Although it is not directly used in the current version of algorithm, the mitigation technique labelled preliminary shear could be implemented in the final version of the predictive tool. All calculations within this section are independent of conical frustum parameters; any similar variables are not related to the conical frustum variables. The techniques presented for analyzing the angle between surfaces may be used in conjunction with the conical frustum. Calculations for the required angle of preliminary shear follow the discussion of all cases investigated.

7.3.1 Case 1: Corner

Corners and edges that appear to have no fillets exhibit unique shearing characteristics that become problematic during draping. The angles between the surfaces contribute to the fabric's ability to drape over all three surfaces, and determine how easy or difficult it is. Figure 7-10 shows a box corner with flat surfaces AB , BC and AC , separated by edges A , B and C . The angles between

surfaces AC and BC , AB and BC , and AB and AC are labelled α , β , and γ , respectively and start at 90° . The angle between the warp and weft yarns is labelled ω , and this also starts at 90° .

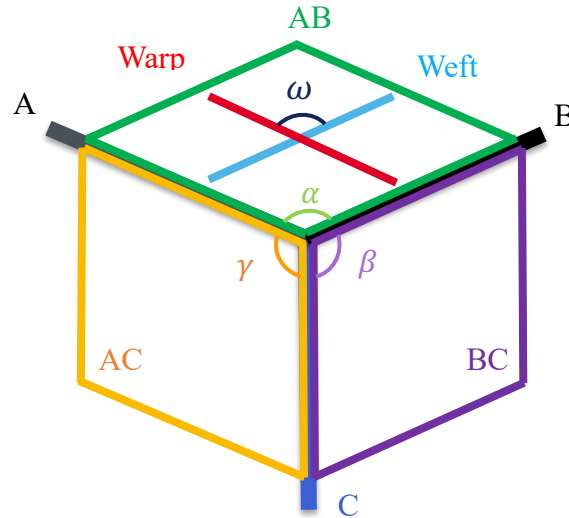


Figure 7-10: Box corner

When a fabric is placed on AB and draped over the edges in an attempt to cover all three surfaces, the warp and weft yarn orientations present conflicts with alignment. Warp and weft yarns along edge C attempt to become parallel to one another, Figure 7-11. Wrinkling is often a result of the angle between the warp and weft either becoming too large or too small due to in-plane shear. This conflict can be attributed to a shear angle of 90° . In reality, a large wrinkle occurs that starts at the corner of the feature, becoming larger as more fabric is draped over surfaces AC and BC , Figure 7-11.

Another major issue with draping features such as this is that there is an excess of material. Since the surface is already covered, the excess fabric must shear until it wrinkles and moves away from the surface. In order to mitigate this, it is proposed that a degree of preliminary shear can be applied to the fabric. By shearing the fabric directly along edges A and B in opposite directions (positive

shear along edge *A* and negative shear along edge *B*), the fabric can conform to all three surfaces without wrinkling. Figure 7-12 shows the fabric behaviour after shearing at 45° along both edges. This will cause the excess fabric to be pressed smoothly onto the surface, rather than wrinkling off the surface. Shearing at 45° along both sides yields a symmetric solution, with the amount of shear on either side being the same. A non-symmetric solution is also viable, Figure 7-12 using 60° and 30° for example. An eventual physical limitation lies in the locking angle of the fabric. The degree of shear to be applied to the edges must sum to the angle between the vertical surfaces along the connecting edge which in this case is 90°.

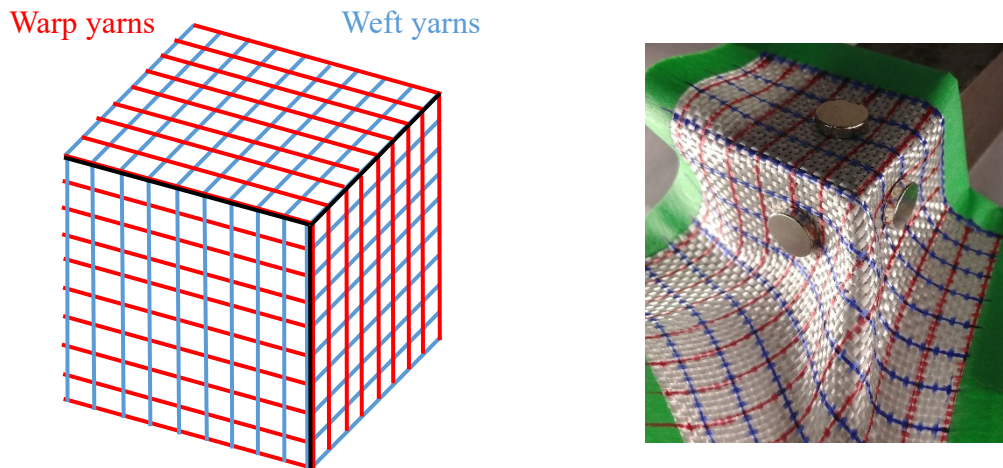


Figure 7-11: Angle between surfaces model (left) and physical draping results (right)

To test the preliminary shearing technique, a steel box measuring 3.8 x 3.8 x 10.6 cm was used with a plain weave glass fabric. Coloured lines were added to indicate the warp (red) and weft (blue) yarns more clearly. Green tape along the sample edges eliminated the amount of fraying induced on these small samples. The tape did not effect the shear behaviour of the fabric but was merely used for preserving the integrity of the samples. Small magnets were used for securing the

fabric in place, enabling shearing to be performed directly along the edges *A* and *B*. Figure 7-13 shows a symmetric and a non-symmetric case tested. It is observed that the preliminary shear technique allows for the fabric to conform to flat surfaces without wrinkling.

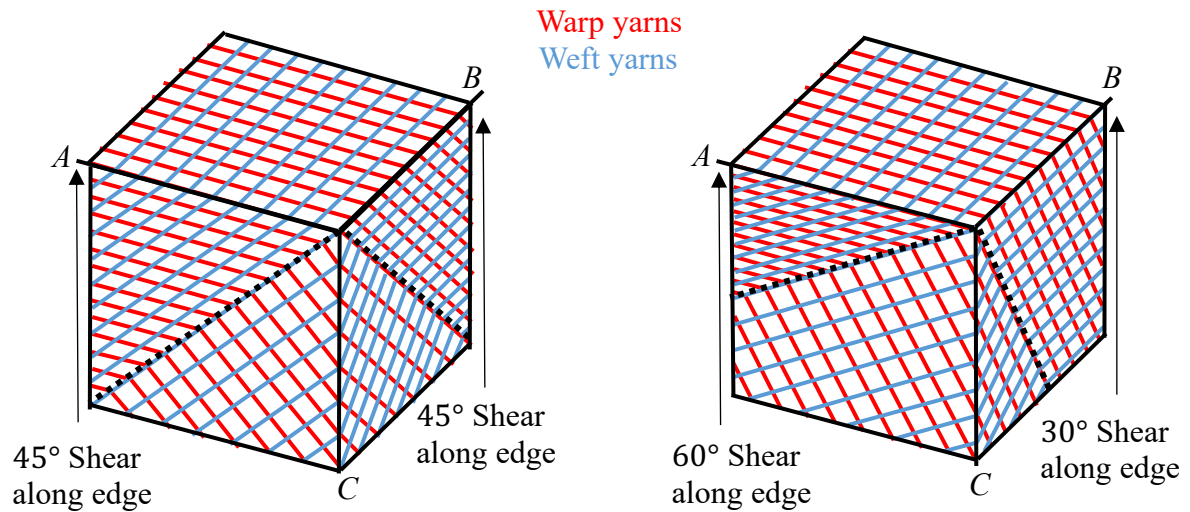


Figure 7-12: Symmetrical preliminary shear (left), unsymmetrical preliminary shear (right)

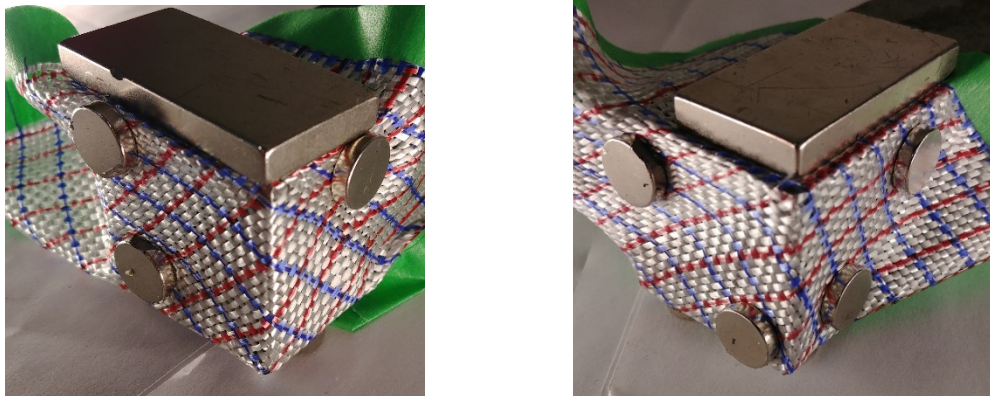


Figure 7-13: Symmetric (left) and non-symmetric (right) preliminary shear results

7.3.2 Case 2: Chamfers along a single edge at a corner

Case 2 investigates the requirements for preliminary shear when a single edge at a corner is chamfered. Angles α , β , and γ need to be calculated based on the degree of chamfer, so that the degree of preliminary shear along each edge can be determined. The amount of preliminary shear required for the fabric to conform around the sharp edges is directly related to the angles between the surfaces.

If a chamfer with angle labelled ϕ_c is along defined a single edge, Figure 7-14, the angle between top plane AC and the angled plane, BC in Figure 7-14, γ in this case, is progressively increased by the chamfer angle from a starting value of 90° . Plane AC remains perpendicular to plane AB and therefore the chamfer angle is simply added to 90° . Since the chamfer is only along edge B , α and β do not change.

This case is interesting as adding a chamfer to one edge only requires preliminary shearing along the edge that is not chamfered, edge A in this example. The fabric is able to move from AC to BC without any conflicts in warp or weft orientations, Figure 7-15a). Once edge C is encountered on surface AC , issues with fibre misalignment resulting in wrinkling occur, as observed in the original case, but to a lesser degree. The angle between surfaces is now greater than 90° and therefore has a lesser shear effect at the corner. To cover a set of surfaces like this, the total preliminary shear applied along edge A is the angle that would enable weft yarns - in this case - to become perpendicular to edge C , which is equal to the angular difference between 180° and γ . Similar to the previous example, magnets were used for holding the fabric in place during preliminary shearing and the fabric did conform to the three surfaces without wrinkling, Figure 7-15b).

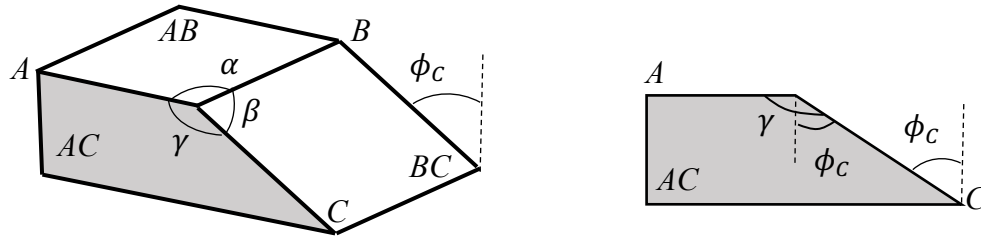


Figure 7-14: Single edge chamfer

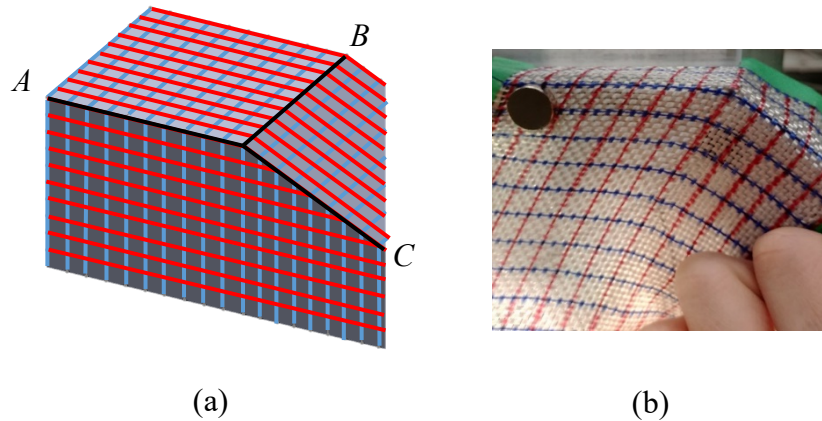


Figure 7-15: Single edge chamfer (a) yarn misalignment (b) preliminary shear drape test

7.3.3 Case 3: Chamfers along multiple edges at a corner

In cases where there is chamfer along both edges A and B , Figure 7-16, more steps are required for determining the preliminary shear angles for a symmetrical solution. Neither AC nor BC are perpendicular to AB anymore, causing both angles β and γ to change. Since β and γ are defined on their respective planes calculating them is no longer a simple addition of the chamfer angle. Calculations for β and γ are performed on their respective BC and AC planes. To begin, a cube is drawn around the chamfered feature. Two triangles are projected onto the cube surfaces, representing the area removed by the chamfer. These planes are labelled AC' and BC' for the respective chamfer plane that the cube plane is facing.

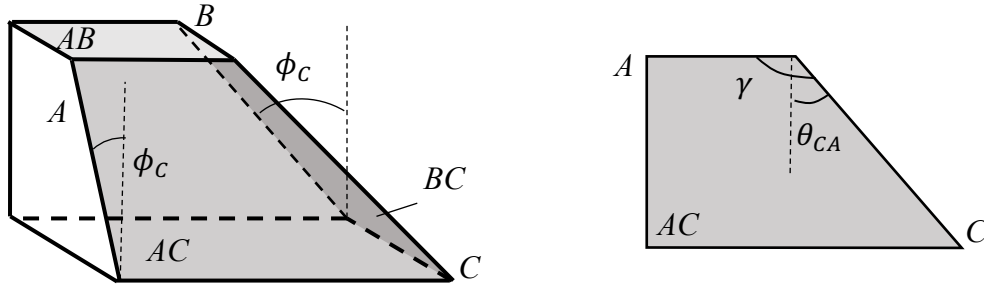


Figure 7-16: Uniform corner chamfer full body (left) and AC plane (right)

Figure 7-17b) shows specific labels for the triangle on the AC' plane of the feature; C_s is the vertical height of the chamfer, A_H is the length along the chamfer on plane AC that is perpendicular to edge A , and A_c is the distance from C_s to the end of the triangle. Similarly for plane BC' , Figure 7-17c), B_H is the length along the chamfer on plane BC that is perpendicular to edge B , and B_c is the distance from C_s to the end of the triangle. θ_{AC} is the angle from A_H to C on AC , and θ_{BC} is the angle from B_H to C on BC . The actual values of the lengths do not need to be known for these calculations. Equations 18 and 19, for angles γ and β respectively, are solely dependent on the chamfer angle and derived using fundamental trigonometric.

$$\gamma = 90 + \theta_{CA} = 90 + \tan^{-1}(\sin(\phi_c)) \quad (18)$$

$$\beta = 90 + \theta_{CB} = 90 + \tan^{-1}(\sin(\phi_c)) \quad (19)$$

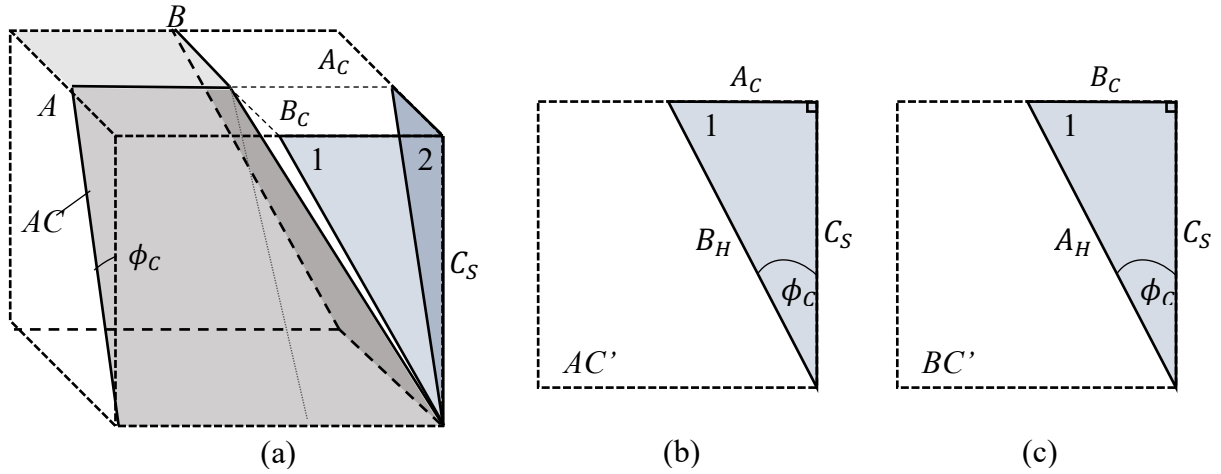


Figure 7-17: Chamfer analysis: (a) cube; (b) AC' triangle projection; (c) BC' triangle projection

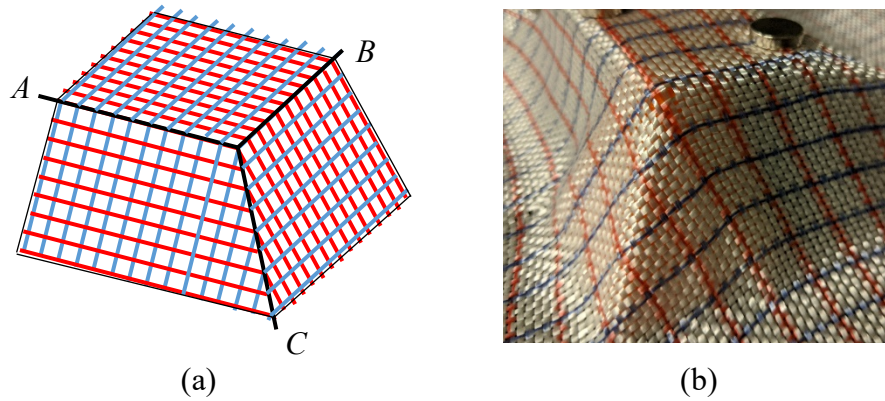


Figure 7-18: Double edge chamfer (a) yarn misalignment (b) yarn alignment (c) preliminary shear test

7.3.4 Preliminary shear angle calculations for Case 1 and 3

Once γ and β have been calculated (Case 3) or when they are known (Case 1), minimum preliminary shear angles, θ_{SA} along edge A and θ_{SB} along edge B can be determined. These angles represent the minimum amount of shear required for ensuring that no wrinkling occurs along edge C . For symmetrical cases which are the simplest for fibre manipulation, $\theta_{SA} = \theta_{SB} = \theta_S$.

Although not explicitly covered in any of the cases, angle α on AB was taken into consideration for these calculations. If the angle of the corner changes as defined by α , angles γ and β change as well, thus the inclusion of α in the preliminary shear angle equations. In a similar manner to Case 3, triangular projections of the missing corner feature caused by the chamfer are projected off the chamfered feature. In these cases, there are four triangles used as opposed to the two used in Case 3. Lengths and angles are once again used in trigonometric ratios to derive new equations, 20 and 21 for γ and β respectively, including α .

$$\gamma = 90 + \theta_{CA} = 90 + \tan^{-1} \left(\frac{\sin(\phi_c)}{\tan \left(\frac{\alpha}{2} \right)} \right) \quad (20)$$

$$\beta = 90 + \theta_{CB} = 90 + \tan^{-1} \left(\frac{\sin(\phi_c)}{\tan \left(\frac{\alpha}{2} \right)} \right) \quad (21)$$

To ensure no wrinkling, warps yarn on surface AC must intersect with edge C at 45° . Similarly on surface BC , weft yarns must intersect with edge C at 45° . This enables the angle between the warp and weft yarns to equal 90° along edge C , ensuring that no wrinkling occurs. This means that the shear angle is dependant on a portion of γ and β , specifically θ_{CA} and θ_{CB} which are the same for symmetrical solutions. Therefore, shear angles for each of the edges are determined using Equation 22:

$$\theta_s = 45 - \theta_{CA} = 45 - \tan^{-1} \left(\frac{\sin(\phi_c)}{\tan \left(\frac{\alpha}{2} \right)} \right) \quad (22)$$

Preliminary shear provides insight on techniques that could be used for mitigating or eliminating wrinkling that may arise during draping. In reality, adding a series of magnets to the draping process would not be ideal or even possible in most industrial applications. However, knowledge that manipulating the fabrics and that applying shear in some areas may reduce shear in others is extremely useful. Across large flat surfaces or large radii double curvature arcs, a degree of shear may be applied to counteract some of the effects of a tighter base surface, effectively restoring the fabric to its starting state. These particular effects require further investigation and are not used in the first version of the draping algorithm.

8 Linking base surface results: The draping algorithm

This chapter discusses the procedure in which the results from the database and linear functions are used for probing draping scenarios. The chapters leading up to this outlined the conical frustum – a 12 parameter tool developed to parameterized base surfaces and structure Database A. Database A contains in-plane shear and yarn orientation angle results built using draping simulations that followed Taguchi simulation design plans. Following the development of the linear functions represent shear, warp yarn and weft yarn orientation angles as functions of geometric and draping parameters, investigations were performed to gain a better understanding of how fabrics will move between surfaces, which lead to the development of the draping algorithm – a series of definitions and steps that uses all aspects presented in this thesis as a single function.

8.1 Draping algorithm overview

The algorithm as a whole is straightforward and it works as follows: for a given mould surface known as a CAD model or a physical mould, vertices, lengths, arc lengths and radii are used for identifying and decomposing the mould into base surfaces. From these variables, conical frustum geometric parameters are calculated for each base surface. Draping strategies, further defined in Section 9.2, identify a sequence of links between individual base surfaces that are consecutively covered until the mould is completely covered. For a selected starting base surface and starting draping parameters, conical frustum parameters for that surface are used in determining shear angles and yarn orientation angles over the surface using the database and linear functions along each of the edges. The shear angles as well as the warp yarn and weft yarn orientation angle results along the edges of the base surface are used as input draping parameters for the next base surface

listed in the draping strategy, which is defined by its own conical frustum parameters. This process of using base surface specific conical frustum geometric parameters and resulting draping parameters is repeated until the entire surface of the mould is completely covered or the drapability limit of the fabric is exceeded, for each strategy. The entire process of linking base surface results is repeated for all combinations of draping strategies until the best one is identified.

8.2 Definition of a draping strategy

When manufacturing composite components, technicians are often provided with a series of steps that are followed to ensure excellent coverage of the mould surface. These steps vary from mould to mould and are dependent on surface topography. Selecting a starting position, orientation and finally the sequence to drape are all necessary in fully defining a single draping strategy. The aim of this section is to define clearly what a strategy is, and how all the different possible strategies for a given case are organized.

Consider a mould surface that contains two base surfaces labelled in Figure 8-1 a). Since there are two surfaces, one will be the starting surface for draping and it will be covered completely first, as a start. In order to drape both base surfaces, a single step is needed in a draping strategy connecting base surfaces 1 and 2. Connections between base surfaces are labelled as the sequence in which the connection is made. For Figure 8-1 a) there are two possibilities; 1-2 if surface 1 is the starting surface, and 2-1 if surface 2 is the starting surface. Similarly in Figure 8-1 b) where there are three base surfaces, if surface 2 as a covered to start, the possible strategies are as follows: 2-1, 2-3 or 2-3, 2-1. Two steps are required to cover these surfaces, in each draping strategy. The number of steps for a draping strategy is defined by the total number of base surfaces minus 1. At a

fundamental level, a draping strategy is defined as a sequence of steps, followed one step at a time to cover a given mould surface with a single pattern. A draping strategy is made up of two core components: the connections between base surfaces, and the sequence in which these connections appear. Reliable and robust draping strategies are translated into plybooks, which are documented steps that technicians use when draping a specific component. A plybook outlines starting position, starting orientation and the steps needed to drape the part.

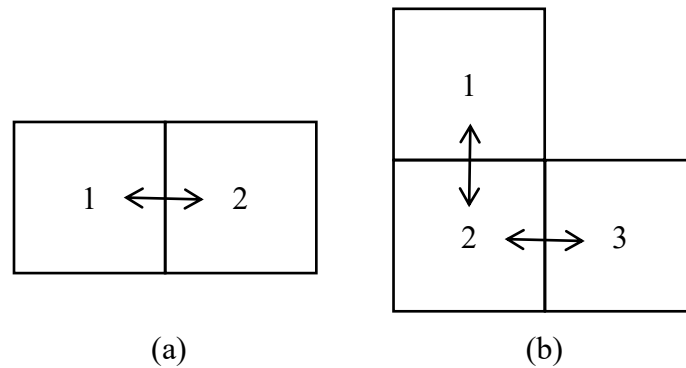


Figure 8-1: (a) Connections between two base surfaces (b) Connections between three base surfaces

In this work, two different strategy definitions 1 and 2 are established that both involve the following features:

- Starting point is defined as the centre point on a single base surface, not as any other point on the base surface, and there is only one starting surface;
- All surfaces are only covered once in a strategy and all strategies are probed at one time;
- Surfaces are exclusively covered from an adjacent surface that has already been covered;
- The surfaces are covered by moving through common edges only, not through vertices;

- When covering a given surface, some of the adjacent base surfaces may already be covered – in practice this affects the draping boundary conditions, which is discussed in Section 8.4;
- There is no method for distinguishing completely impractical solutions such as 7-8, 8-9, 9-6, 6-3, 3-2, 2-1, 1-4, 4-5 shown as an example in Figure 8-2.

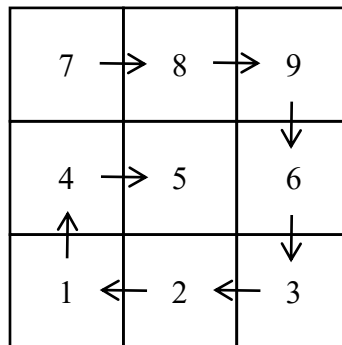


Figure 8-2: Impractical draping strategy

Strategy definition 1 defines two different strategies that involve the same individual steps but in a different sequence as two distinct strategies. For example, considering Figure 8-1 b) and starting from surface 2: two possible strategies as defined by strategy definition 1; 2-1, 2-3 and 2-3, 2-1. Definition 1 states that even though the individual steps are the same, the order in which these steps appear, and therefore the order in which the surfaces are draped is different, thus the strategies are different. Strategy definition 2 defines two different strategies that involve the same individual steps but in a different sequence as a single strategy. Using Figure 8-1 b) again, 2-1, 2-3 and 2-3, 2-1 use the same individual steps, therefore the strategies as defined by strategy definition 2, are the same. Alternative strategy definitions become useful when considering symmetric base surface. In these cases, draping strategy definition 1 will probe a large number of draping strategies where the same connections in a different order always yields the same result. By using definition

2 in this case, processing time will be reduced by grouping all the draping strategies with the same results into one strategy. For the draping algorithm in this thesis, strategy definition 1 is used.

8.3 Organization of surface connections

Once that strategies were defined, a method for organizing all the possible connections was established. For a specific mould, a list of all possible surface connections can be made. Consider a design with four base surfaces, Figure 8-3. For a single draping strategy only three connections are needed but there are eight possible connections: 1-2, 1-4, 2-1, 2-3, 3-2, 3-4, 4-1, and 4-3.

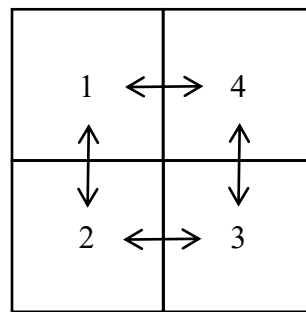


Figure 8-3: Base surface links

Since a draping strategy is a sequence of steps, each surface connection is assigned a sequence number for a particular strategy. Depending on the starting surface, some of the connections will never be used. If base surface 1 was the starting surface for example, links 2-1 and 3-1 would not be used since surface 1 always starts covered. This method of defining draping strategies is very useful, allowing tables to be generated outlining every possible draping strategy using all base surface connections. With the sequence number known and the links defined, a table is generated that lists any individual strategy for a set of base surfaces. Consider Figure 8-3; Table 8-1 lists all possible draping strategies starting from base surface 1. The first column for a particular base

surface connection identifies whether the link is in that draping strategy (Y) or not (N). If it is, it is assigned a sequence number (Seq. #), otherwise no value is assigned since that connection is not used in that strategy. In the specific case of the example it is observed that links 2-1 and 4-1 are never used if surface 1 is the starting surface. If every base surface is used as a starting point, this set of base surfaces leads to a total of 36 draping strategies, as set under definition 1.

The table is very useful as it lists all possible strategies for a particular surface, and it can easily link to Database A. Base surfaces are defined as surface variables such as, lengths, radii, arc lengths, etc. that are used for calculating conical frustum parameters. The conical frustum parameters are direct links to Database A and to linear functions where the shear angle and yarn orientation results can be accessed. As each step in a sequence for a strategy is encountered, results from Database A are continuously probed until all surfaces are covered, and final shear angles are determined.

Table 8-1: Draping strategies starting from base surface 1

Draping Strategy	Base Surface Link															
	1-2		1-4		2-1		2-3		3-2		3-4		4-1		4-3	
	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #
1	Y	1	Y	2	N	-	Y	3	N	-	N	-	N	-	N	-
2	Y	1	Y	2	N	-	N	-	N	-	N	-	N	-	Y	3
3	Y	1	N	-	N	-	Y	2	N	-	Y	3	N	-	N	-
4	Y	1	Y	3	N	-	Y	2	N	-	N	-	N	-	N	-
5	Y	2	Y	1	N	-	Y	3	N	-	N	-	N	-	N	-
6	Y	2	Y	1	N	-	N	-	N	-	N	-	N	-	Y	3
7	N	-	Y	1	N	-	N	-	Y	3	N	-	N	-	N	-
8	Y	3	Y	1	N	-	N	-	N	-	N	-	N	-	N	-

All the possible draping strategies were identified manually for the case of a mould surface featuring 7 base surfaces, Figure 8-4. 16 base surface connections were identified leading to a total of 767 draping strategies starting from base surface 1. Ensuring that these strategies are organized and easily accessed enables results to be probed, and the best draping strategy can be identified

quickly. This reinforces the meticulous nature and usefulness of this project. The number of possible draping strategies is very large for industrial components therefore accessing and probing numerous possible draping strategies quickly allows for the design and preforming process time to be reduced, ultimately reducing costs.

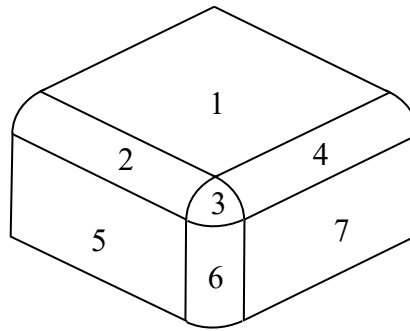


Figure 8-4: Mould with 7 base surfaces

8.4 Consolidation of boundary conditions

As the algorithm probes single draping strategies, shear angles and yarn orientation angles along common edges are used as input values, or boundary conditions, for the next base surface in sequence. Geometric conical frustum parameters do not change from draping strategy to draping strategy; however, yarn orientation angles and shear angles do change.

The reader may recall that draping starting position, draping starting orientation and angle between warp and weft yarns are used for structuring the database. When considering a draping strategy, these draping parameters can be derived from the results along each base surface edge. These newly derived draping parameters are then used on the next surface in the draping strategy sequence until all surfaces are covered.

As strategies continue to be probed and different boundary conditions are encountered, it is inevitable that a draping strategy sequence will contain a surface in the sequence that has more than one boundary condition. Consider a sequence 1-2, 1-4, 2-3, 4-7, 2-5, 5-6, 5-8, 8-9 for Figure 8-5: Draping sequence boundary conditions. Base surface 5 has multiple boundary conditions after the second step in the sequence, causing two different input orientations and shear angles along two edges, Figure 8-5. Although surface 5 is being entered from surface 2, the results along the edge between surfaces 4 and 5 cannot be disregarded.

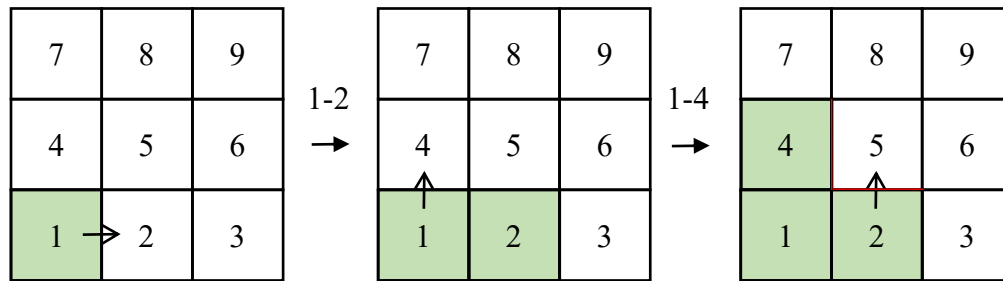


Figure 8-5: Draping sequence boundary conditions

Referring back to Chapter 5, draping simulations in Nottingham DrapeIt are entered at a single point and therefore results are presented in Database A using a single starting point. This means that for cases such as this where there are multiple boundary conditions, the results along the unbound edges (edges that have not been covered) must be a combination of the results from both boundary conditions. Averages for shear, warp yarn orientation, and weft yarn orientation angles are taken and applied to the centre of each constrained edge. This produces draping results from each set of boundary conditions on the unbound edges. By taking the average of those results for each edge, a final consolidated result remains.

Beginning with base surface 2 in Figure 8-5, warp and weft yarn orientation angles are available to be used as input values at the midpoint and at the ends of the edge. Considering warp

orientations, results exist for three points along the edge. Either the warp orientation at the centre of the edge, or an average of all three warp orientations is used in the linear functions to determine the final results for base surface 5, Figure 8-6 b). From base surface 2, warp orientation angles are generated for the edges of base surface 5 that do not have boundary conditions. The same process is performed for the weft orientation angles. Similarly for the other set of boundary conditions from base surface 4, results for the edges without boundary conditions can be generated using the midpoint input or an average of all three results along the edge. This leads to two sets of results along the remaining edges of the base surface, Figure 8-6 c). The averages of these results are taken to produce a solution for the surface in the sequence, Figure 8-6 d); the process is described in Chapter 10. This does not provide any information about the shear pattern on the surface; however, if the results along the edges remain within the fabric drapability limit, the process is continued. During the population of Database A, all the results were compiled ensuring that warp and weft orientation angles were calculated as always defined outside the surface, Figure 8-7. This ensures that when averaging the results along the edges, that average orientations do not clash.

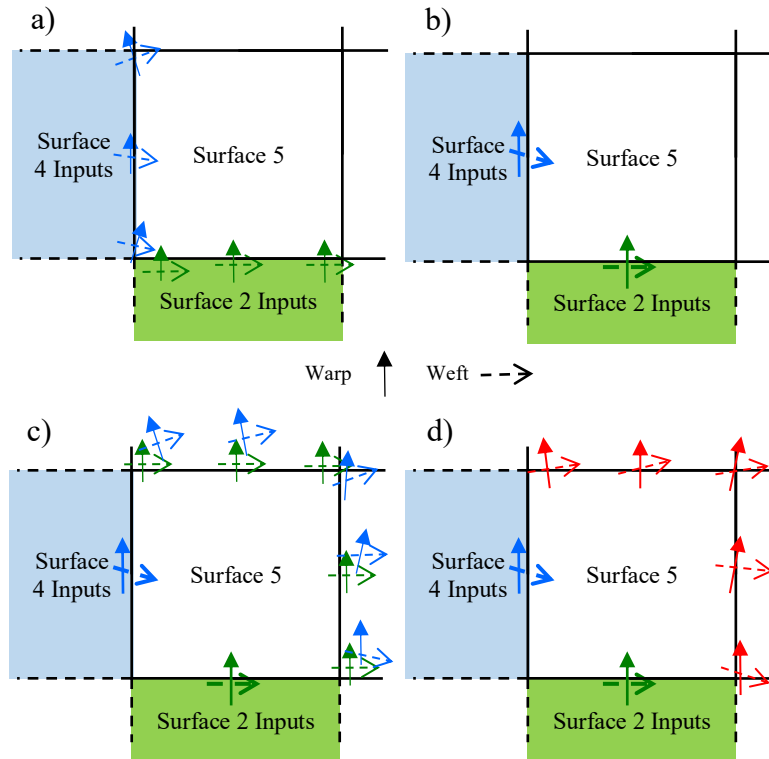


Figure 8-6: Consolidation of boundary conditions a) Boundary conditions; b) Chosen boundary condition input; c) Results from both boundary conditions; d) Average of results

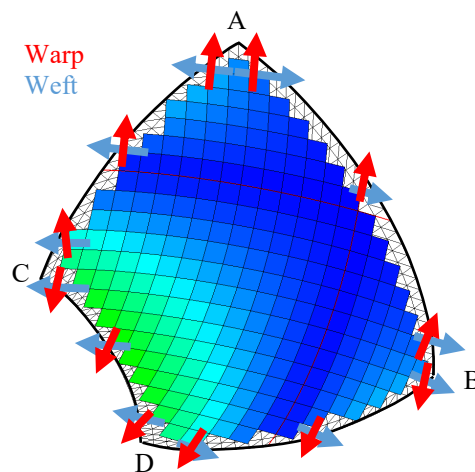


Figure 8-7: Orientation angles along base surface edges

9 Draping algorithm demonstration

Physical demonstrations of draping operations that follow a predicted draping strategy, performed on a physical mould, enabled the validation of the method for consolidating boundary conditions presented in Chapter 8. Draping on a physical mould also provided confirmation of predicted results obtained using the draping algorithm. For these demonstrations, a physical mould was built that featured three flat base surfaces, three single curvature surfaces and one double curvature surface, similar to the case shown in Figure 8-4, Chapter 8. Figure 9-1 shows the mould decomposed into individual base surfaces with conical frustum vertex labels and overall dimensions. These were used for calculating the conical frustum parameters for each base surface following the steps presented in Chapter 4. The physical mould was built using steel sheets so that individual base surfaces connections could be performed following each step in a given draping strategy sequence, applying magnets onto the fabric to hold it in place at each step.

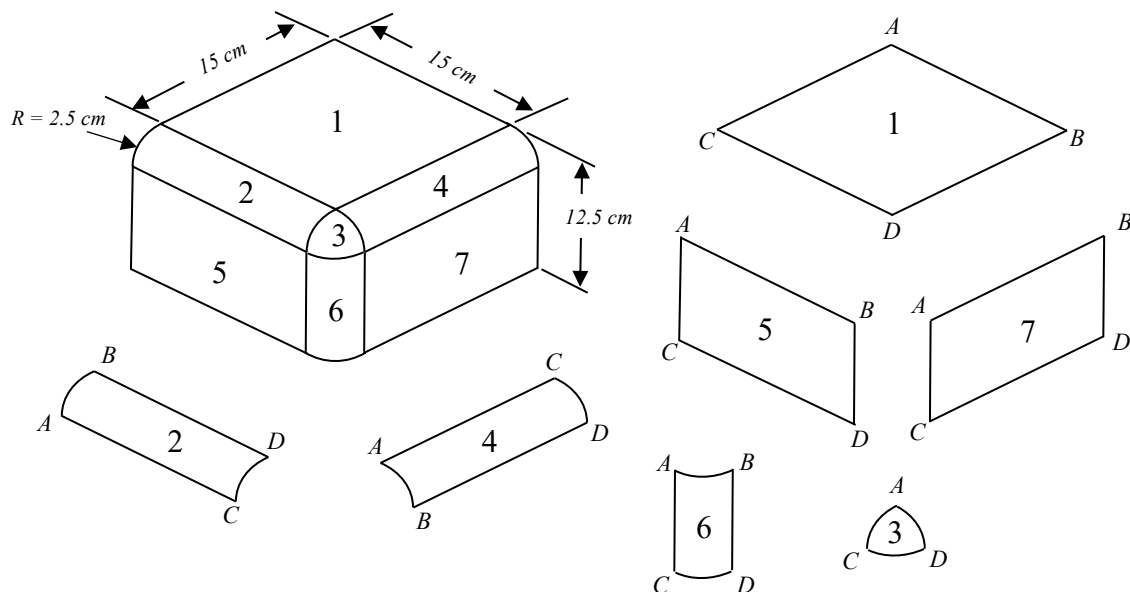


Figure 9-1: Surface decomposition and vertex labelling

9.1 Partial draping demonstrations

9.1.1 Draping starting direction

The draping algorithm requires starting directions for the warp and weft yarns to be defined so that the yarn orientation angles can be calculated for each surface. Since warp and weft yarn orientation angles are calculated using vectors, the angles are directly related to the directions along which the yarns are extending. Establishing a starting direction ensures that the angles calculated are appropriately referenced. Appendix C details how the warp and weft orientation angles are calculated using orientation vectors extracted from simulations results, and base surface edges which are identified using vectors following the same sequence for each base surface. Further details on the draping algorithm calculations are presented in Section 9.3. After the starting surface is selected and covered, results along the edges of that surface will be extracted from Database A hence known, at which time warp and weft yarns starting direction are designated. For the cases considered here, warp and weft yarns on the starting surface always begin orthogonal, and always at the centre of the starting surface. The yarns spread over the starting surface until edges common to neighbouring base surfaces are reached. At this stage, the warp yarns and weft yarns do not have a designated direction of path along which to extend further; this is defined by the draping strategy. The first connection made between two base surface, 1-2 in Figure 9-2, defines the starting and travel direction of the warp yarns, weft yarns, or both. From the designation in Nottingham DrapeIt, the starting weft direction is always 90° counter clockwise from the warp starting direction. Alternatively, the starting warp direction is always 90° clockwise from the weft starting direction.

In the event that the first draping strategy connection is along an angled edge, or the starting orientation is not 0°/90° for the first surface edges, the first base surface link still defines the starting

direction for both yarns. For example, the draping sequence in Figure 9-2 is 1-2, 1-3. Since 1-2 is the first step in the strategy, warp yarns move onto surface 2 first therefore defining the extending direction for the warp yarns. Alternatively, the starting directions for the yarns in Figure 9-2 b) simply follow the yarn paths as they follow connection 1-2 into surface 2.

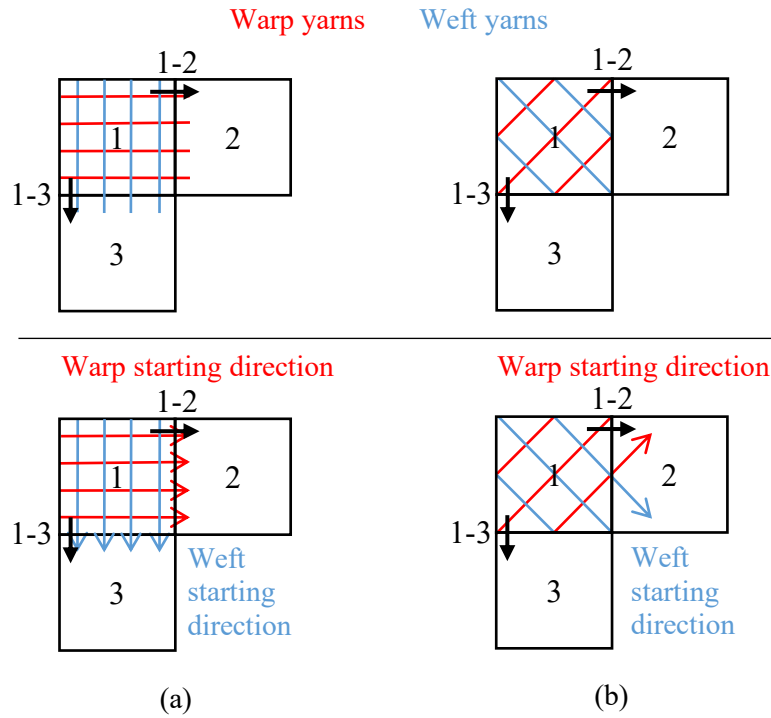


Figure 9-2: Warp and weft starting orientations (a) aligned and (b) misaligned with edges

9.1.2 Demonstration 1 and 2

In order to test different draping strategies, partial draping strategy sequences were draped onto the mould surface, to compare physical results with draping algorithm predictions and to validate the consolidation of boundary conditions method discussed in Chapter 9. The first three steps for six different draping strategies are presented in Table 9-1. Three more steps would be needed to cover this mould completely but for the purposes of confirming algorithm results and

demonstrating the consolidation of boundary conditions, only the first three steps were used for all demonstrators.

A plain weave glass fibre fabric was cut into two patterns, Figure 9-3, so that the first two steps in draping strategies 1 and 2 could be performed. A 1 *cm* x 1 *cm* grid was drafted on the patterns to denote the warp (red) and weft (blue) yarns as well as allowing shear to be observed more easily. These partial draping demonstrations confirmed the different boundary conditions introduced when probing draping strategies step by step. The starting base surface for these demonstrators was base surface 1 with warp and weft yarn orientations along edge *AC* at 90° and 180° respectively.

Table 9-1: Partial draping strategies

Draping Strategy	1-2		1-4		2-3		3-2		3-4		4-3	
	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #	Y/N	Seq. #
1	Y	1	N	-	Y	2	N	-	Y	3	N	-
2	N	-	Y	1	N	-	Y	3	N		Y	2
3	Y	1	Y	2	Y	3	N	-	N	-	N	-
4	Y	1	Y	2	N	-	N	-	N	-	Y	3
5	Y	2	Y	1	Y	3	N	-	N	-	N	-
6	Y	2	Y	1	N	-	N	-	N	-	Y	3

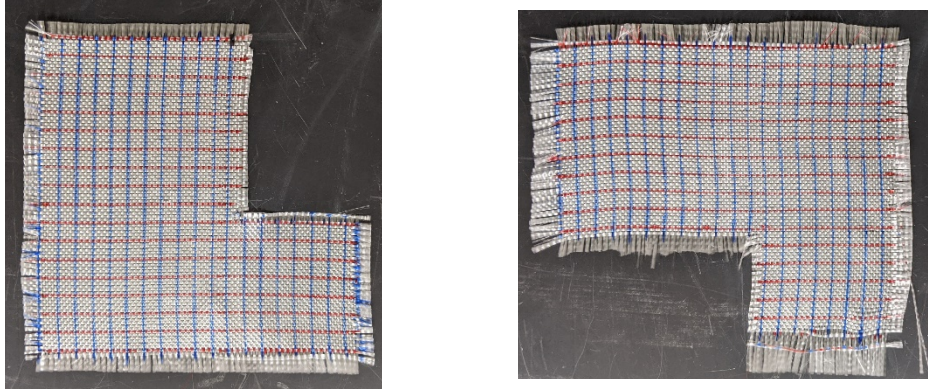


Figure 9-3: Pattern 1 (left); Pattern 2 (right)

Using pattern 1 for draping strategy 1, the first step draped the fabric from flat base surface 1 to single curvature base surface 2. Results presented in Chapter 6 confirm that single curvature and flat base surfaces do not cause shear, therefore no shear was observed during this step. Step 2 in the sequence draped the fabric from single curvature base surface 2 over double curvature base surface 3. The draping boundary conditions for the warp and weft orientations along edge *AC* of base surface 3 are 90° and 180° respectively. These boundary conditions are defined using the standard edge vector directions and the starting warp and weft yarn orientation directions, Figure 9-4. This caused similar amounts of positive shear and negative shear, following the definition highlighted in Section 6.3 on edges *CD* and *AD* respectively. For the partial demonstration, shear angles can be observed in Figure 9-5. Using pattern 2, step 1 in draping strategy 2 is similar to step 1 in draping strategy 1 – moving from a flat base surface to a single curvature base surface. Boundary conditions for warp and weft orientation angles along edge *AD* for base surface 3 were 0° and 90° respectively. Strategy 2 lead to similar results for surface 3, causing positive shear and negative shear along edges *CD* and *AC* respectively. Partial demonstration results using pattern 2 are shown in Figure 9-5.

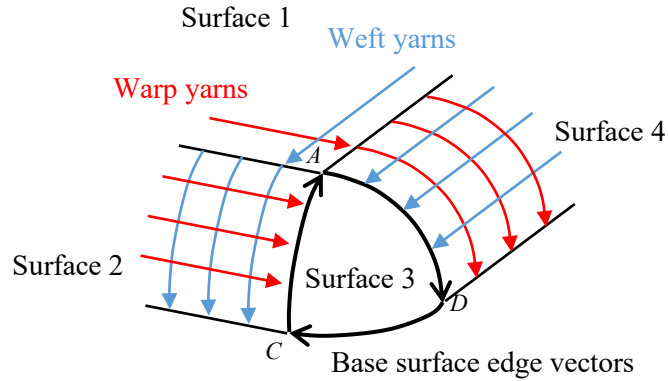


Figure 9-4: Surface 3 boundary conditions

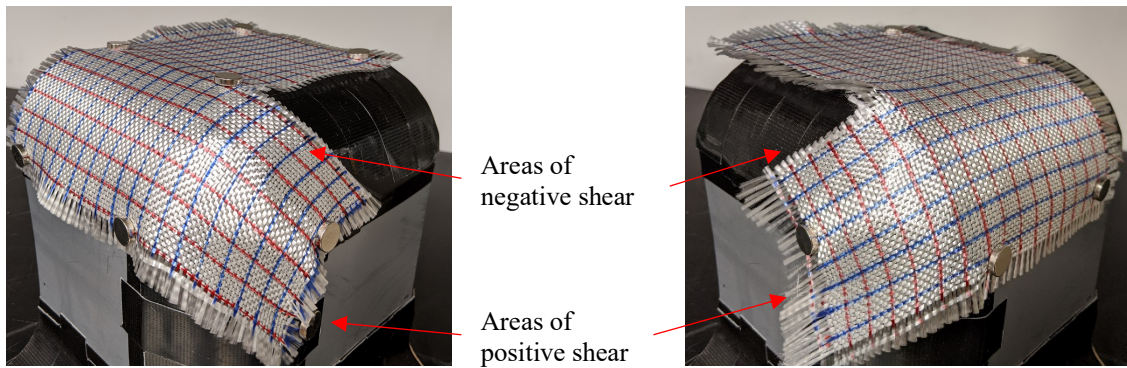


Figure 9-5: Pattern 1 (left) and pattern 2 (right) partial demonstration results

The partial draping sequence demonstrations allowed changes in yarn orientations to be observed in the first two steps of both draping strategies. These changes directly influence the next steps in each strategy, therefore confirming different boundary conditions when connecting base surfaces. For both strategies, draping algorithm predictions are outlined in Table 9-2 and Table 9-3. Comparing the results to the demonstrations is difficult due to the curved nature of the surfaces and the geometry of the patterns. Results for these algorithm predictions are confirmed later in the next section.

Table 9-2: Linear function results for surface 3 draping from surface 2

Base Surface 3		Linear Function Results (°)		
Edge	Point	Shear Angle	Warp Orientation Angle	Weft Orientation Angle
DC	D	1.53	49.9	95.6
	M_{DC}	14.4	30.76	106.3
	C	1.63	7.17	84.3
AC	A	-1.88	173.13	94.99
	M_{AC}	-16.35	148.32	105.33
	C	-0.77	134.91	135.86

Table 9-3: Linear function results for surface 3 draping from surface 4

Base Surface 3		Linear Function Results (°)		
Edge	Point	Shear Angle	Warp Orientation Angle	Weft Orientation Angle
DC	D	1.53	7.17	95.6
	M_{DC}	14.4	30.76	106.3
	C	1.63	49.9	84.3
AD	A	-1.88	85.01	6.87
	M_{AD}	-16.35	74.67	31.68
	D	-0.77	44.14	45.09

9.2 Validating consolidation of boundary conditions

Both draping strategies were performed using a square pattern of fabric to validate the consolidation of boundary conditions method presented in Chapter 9. Following draping strategy 1, the square pattern was draped over four surfaces in sequence, Figure 9-6.

It was observed that the warp and weft yarn orientations from base surface 3 along edge AD contributed to shear on base surface 4. To compare the demonstrated physical results to the results determined by the draping algorithm, calculations for the warp and weft yarn orientation angles on edges BD and CD of base surface 4 were performed using linear functions.

Since the linear functions produce results at three locations along each edge, averages were taken for the warp orientation, for the weft orientation and shear angles, for both edges, and these averages were used as boundary conditions. These boundary conditions were applied at the midpoint of edge AB on base surface 4, and at the midpoint of edge AC of base surface 4. From the midpoint of AB , no shear was observed since the fabric was moving from a flat base surface to a single curvature base surface. Alternatively, the shear induced by base surface 3 was carried into base surface 4. Averaging the results along edges BD and CD of base surface 4 using yielded a final shear angle of -22.5° .

On base surface 7, angular measurements were taken between warp and weft yarns to confirm the results calculated by the draping algorithm. Five measurements were taken along AB on surface 7, yielding a final average shear angle of -31° . The greater measured shear angle can be attributed to the fact that averages were used for the draping algorithm results along the edges. This can also be attributed to shear progression not being addressed in this iteration of the algorithm. Since a double curvature surface is encountered, a change in shear angle and warp and weft yarn orientations is observed on edge AC which would lead to shear progression. This would cause shear to increase to a maximum on the adjacent single curvature surface.

The same procedure was followed for draping strategy 2 to calculate the final shear and warp and weft orientation angles for surface 2. Due to the symmetric nature of this demonstrator, the same average shear angle of -22.5° was calculated using the draping algorithm, and an average of -33° was measured on flat base surface 5.

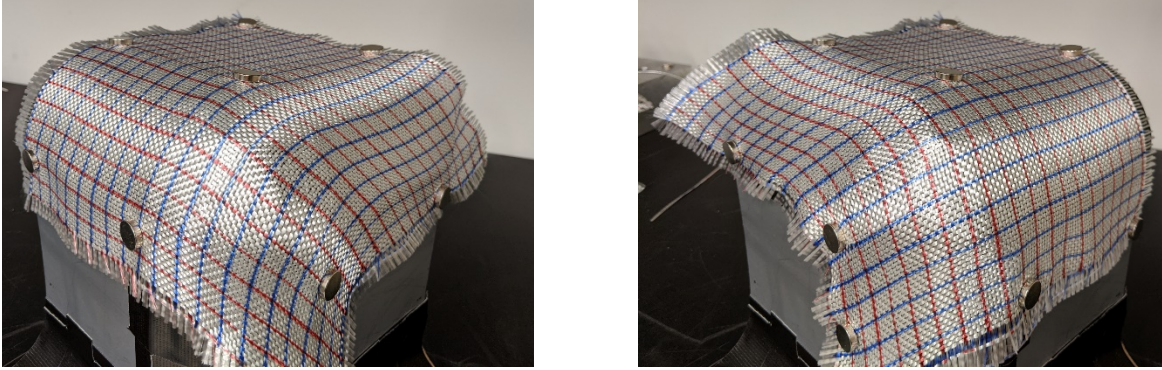


Figure 9-6: Draping strategy 1 (left) and draping strategy 2 (right) demonstrations

For the remaining draping strategies, it was observed that each one yielded the same results. Strategies 3, 4, 5 and 6 all covered base surface 3 last, causing it to have two boundary conditions. Since the boundary conditions remained the same for all four strategies, the draping algorithm always predicted the same result. Cases like these reinforce the usefulness for multiple draping definitions which were discussed in Chapter 9. Since this version uses definition 1, each of these strategies would be probed as they are each defined as unique strategies. If definition 2 had been used, the draping algorithm would have identified that these four draping strategies all contain the same base surface connections, but in a different sequence. This would eliminate three possibilities since they would all yield the same results. For symmetric mould surfaces, definition 2 may assist in drastically reducing the number of strategies that need to be probed.

9.3 Draping algorithm sample

At the time of writing, all the steps described above are performed manually for each base surface. In the future, all complete strategies and their results will be probed quickly using the final version of the predictive tool. The following calculations were performed for draping strategy number 3

using the core conical frustum parameters calculated for base surface 3, and the boundary conditions along edge AB for surface 4 and edge DC for surface 2. Core conical frustum parameters are as follows: $|\vec{R}_1| = 0$, $|\vec{R}_2| = |\vec{L}|$ $C_R = 2.5$ cm and, $\beta = 90^\circ$. Table 9-2 and Table 9-3 express the results on surface 3 using boundary conditions from surfaces 2 and 4.

As a starting point, warp and weft yarn starting directions are defined. For strategy 3, the fabric is placed on surface 1 with weft yarns travelling into surface 2, and warp yarns travelling into surface 4. Since 1-2 is the first step in the sequence, the weft yarn starting direction is defined toward base surface 2 from base surface 1, and the starting direction defined for the warp yarns is toward base surface 3 from base surface 1, Figure 9-4. Following the draping strategy in sequence, the first two steps induce no shear on the fabric since the surfaces are single curvature. Boundary conditions calculated for surfaces 2 and 4 using the starting directions and edge vectors for surface 1 are presented in Table 9-4 and, respectively.

Table 9-4: Base surface 2 edge boundary conditions

Base Surface 3		Linear Function Results (°)		
Edge	Point	Shear Angle	Warp Orientation Angle	Weft Orientation Angle
AB	A	0	90	180
	M_{AB}	0	90	180
	B	0	90	180
BD	B	0	0	90
	M_{BD}	0	0	90
	D	0	0	90
DC	D	0	90	0
	M_{DC}	0	90	0
	C	0	90	0
CA	C	0	180	90
	M_{CA}	0	180	90
	A	0	180	90

Table 9-5: Base surface 4 edge boundary conditions

Base Surface 3		Linear Function Results (°)		
Edge	Point	Shear Angle	Warp Orientation Angle	Weft Orientation Angle
AB	A	0	0	90
	M_{AB}	0	0	90
	B	0	0	90
BD	B	0	90	180
	M_{BD}	0	90	180
	D	0	90	180
DC	D	0	180	90
	M_{DC}	0	180	90
	C	0	180	90
CA	C	0	90	0
	M_{CA}	0	90	0
	A	0	90	0

Boundary conditions along edge DC for surface 2 and edge AB for surface 4 were translated into boundary conditions for edges DA and AC on surface 3. The average warp orientation and weft orientation angles were calculated at the midpoint of each edge and used in the linear functions for the draping parameters.

The conical frustum parameters were input into linear functions for warp orientation and weft orientation angles at points C , M_{DC} , and D . Since the functions were designed using the Taguchi method, the minimum and maximum levels between -1 and +1 are correlated directly to the minimum and maximum extents that exist within the database. For example, C_R database ranges from 2 to 20 cm in the database as designated in Chapter 6. These values correspond to the minimum and maximum levels for that particular Taguchi modality. Setting C_R to 2.5 causes linear interpolation to procedure between -1 and 1 setting the level of C_R to -0.94. This process was completed for all the geometric conical frustum parameters. Boundary conditions for the draping

parameters were also applied to the function. The results from the linear function yielded a final warp or weft orientation angle for a specific point on surface 3. Six linear functions were used for calculating the warp and weft yarn orientation angles at points C , M_{DC} , and D on surface 3. This process was repeated for the boundary conditions along edge AD , yielding another set of results along the edge DC . The results for each set of boundary conditions are found in Table 9-2 and Table 9-3.

The consolidation of boundary conditions was performed as a final step, yielding the warp and weft yarn orientation results for edge DC on surface 3, Table 9-6. Using the orientation angles, shear angles along the edge are also calculated. The results from the demonstrator were shown in Figure 9-7.

Table 9-6: Final draping results along edge DC for draping strategy 3

Base Surface 3		Linear Function Results (°)		
Edge	Point	Shear Angle	Warp Orientation Angle	Weft Orientation Angle
D	D	28.54	89.95	28.48
	M_{DC}	30.76	106.3	47.06
	C	28.54	89.95	28.48

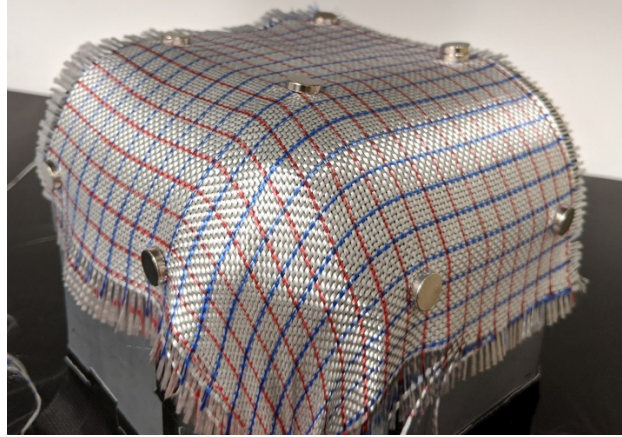


Figure 9-7: Physical demonstrator results

Results for the consolidation of boundary conditions performed by the draping algorithm were found to be truthful to the results found from the physical demonstrators. Inevitably during draping, surfaces with multiple boundary conditions will be encountered especially when probing through all draping scenarios. To determine the results for cases like these, the consolidation of boundary conditions is an effective method for generalizing the behaviour. Shear angles and warp and weft orientation angle outputs enable strategies with the least amount of shear to be identified. In general, these cases are better than those with high shear angles or large variances in warp and weft orientations. Overall, this core version of the draping algorithm will be the groundwork used for the predictive tool.

10 Conclusions and recommendations

This thesis presents contributions to the preliminary stages of a project for developing a draping predictive tool. Contributions for this project include:

- Developing of a parameterization tool – the conical frustum – that is used to uniformly quantify base surfaces on industrial moulds;
- Performing draping simulations following Taguchi methods of experimental design using the core conical frustum, and populating a database with shear, warp yarn orientation and weft yarn orientation angles;
- Designing linear functions from database results;
- Investigating unique shear behaviour observed when draping and;
- Developing the core draping algorithm used for linking base surface results.

For the continued development of the draping tool, database population should be an ongoing process. The more data the draping algorithm has access to, the more accurate the draping results will be. Ideally, this process becomes automated using all the available conical frustum parameters. This will allow the algorithm to produce more accurate results on infrequent, complex base surfaces.

Further investigations on the effect of twisted base surfaces should be performed. Although it is encompassed into the 12-parameter conical frustum, a statistical analysis featuring twisted flat, single curvature and double curvature may yield useful results.

The incorporation of shear progression is a necessary addition for the draping algorithm. This phenomenon is present on nearly every surface featuring double curvature and will undeniably affect individual draping strategies. The work in this thesis presents the fundamental mechanics

that highlight what shear progression is and calculations that determine the magnitude and location of such occurrences. Mitigation techniques such as preliminary shearing or the use of large flat surfaces as “healing” or adjustment areas for shear are suggestions for future investigations.

Currently, there are investigations being performed on how fabric properties such as bending stiffness, fabric spring back, friction, and locking angle play a role during the preforming process. The results are compiled into another database, Database B, which will be used in conjunction with Database A so that the best draping strategy can be identified using most compatible fabric. Alongside this, significant progress towards automating point extraction, base surface labelling and conical frustum calculations from CAD models has been made, and is ongoing to further hasten the identification of the best draping strategy.

11 References

- [1] J. van Suchtelen, "Product properties: A new application of composite materials," *Phillips Res. Repts*, vol. 27, 1972.
- [2] F. Campbell, *Structural Composite Materials*, Materials Park: ASM International, 2010.
- [3] A. M. Bucur, "Survey of the state of the art composites and sandwiche structures," *Acta Universitatis Cibiniensis*, vol. 67, pp. 174-178, 215.
- [4] AZO Materials, "Titanium Alloys - Ti6Al4V Grade 5," 22 07 2002. [Online]. Available: <https://www.azom.com/article.aspx?ArticleID=1547>. [Accessed 17 11 2019].
- [5] AZO Materials, "AISI 4130 Alloy Steel Datasheet," 12 11 2012. [Online]. Available: <https://www.azom.com/article.aspx?ArticleID=6742>. [Accessed 17 11 2019].
- [6] Teijin, "Tenax Filament Yarn," Teijn, 27 06 2018. [Online]. Available: https://www.tejincarbon.com/fileadmin/PDF/Datenbl%C3%A4tter_en/Filament-Product_programm__EU__v27_2018-06-27_EN.pdf. [Accessed 17 11 2019].
- [7] Matweb, "Matweb Material Property Data," [Online]. Available: <http://www.matweb.com/index.aspx>. [Accessed 17 11 2019].
- [8] P. Mallick, *Fibre Reinforce Composites - Materials, Manufacturing and Design*, Boca Raton: Taylor & Francis Group, 2007.

- [9] ASM International, ASM handbook: Volume 21, composites, Materials Park, OH: ASM International , 2001.
- [10] K. Campbell, "Airbus to start manufacturing parts for new A350 XWB in late '09," Creamer Media's Engineering News, 11 May 2009. [Online]. Available: http://www.engineeringnews.co.za/article/airbus-to-start-manufacturing-parts-for-new-a350-xwb-in-late-09-2009-05-11/rep_id:4136. [Accessed 17 11 2019].
- [11] J. Teresko, "Boeing 787: A Matter of Materials -- Special Report: Anatomy of a Supply Chain," Industry Week, 15 November 2007. [Online]. Available: <https://www.industryweek.com/companies-amp-executives/boeing-787-matter-materials-special-report-anatomy-supply-chain>. [Accessed 17 11 2019].
- [12] S. Gagne, *Formage Sous Vide de Renforts Textiles de Carbon et de Verre pour Fabrication de Pieces Composites pour l'Aerospatiale*, Ottawa, Ontario: University of Ottawa, 2017.
- [13] F. Robitaille, *Reinforcements and Micromechanics Presentation*, Ottawa, Ontario: University of Ottawa, 2018.
- [14] H. Nied and B. Bekisli, "Thermoforming of Knitted Composite Structures: FEM Simulation and Experiments," *International Journal of Material Forming*, vol. 3, pp. 615-618, 2010.
- [15] P. Irving, *Polymer Composites in the Aerospace Industry*, Elsevier, 2014.
- [16] J. Bijwe and R. Rattan, "Carbon fabric reinforced polyetherimide composites: Influence of weave of fabric and processing parameters on performance properties and erosive wear," *Materials Science and Engineering*, vol. A, no. 420, pp. 342-350, 2006.

- [17] T. Charpe, "Global Braided Composites Market Forecast 2018-20205, Dynamics, Analysis, Supply and Demands," Red Newswire, 04 04 2018. [Online]. Available: <https://www.rednewswire.com/global-braided-composites-market-forecast-2018-2025-dynamics-analysis-supply-and-demands/>. [Accessed 17 11 2019].
- [18] C. Rudd, M. Turner, A. Long and V. Middleton, "Tow placement studies for liquid composite moulding," *Composites Part A: Applied Science and Manufacturing*, vol. 30, no. 9, pp. 1105-1121, 1999.
- [19] B. Astrom, Manufacturing of polymer composites, London: Chapman and Hall, 1997.
- [20] C. Mack and H. Taylor, "The fitting of woven cloth to surfaces," *Journal of the textile institute transactions*, vol. 47, 1956.
- [21] A. L. Stewart, "Misalignment defects in unidirection composite materials," Vancouver, 2018.
- [22] A. Prodromou and J. Chen, "On the relationship between shear angle and wrinkling of textile composite preformes," *Composites part a: Applied science and manufacturing*, vol. 28, 1997.
- [23] J. Sjolander, P. Hallander and M. Akermo, "Forming induced wrinkling of composite laminates: A numerical study on wrinkling mechanisms," *Composites Part A: Applied Science and Manufacturing*, vol. 81, pp. 41-51, 2016.
- [24] A. Long, Design and manufacture of textile composties, Woodhead Publishing, 2005.

- [25] J. Cao, R. Ackkerman, P. Boisse and e. al., "Characterization of mechanical behaviour of woven fabric: Experimental methods and benchmark results," *Composites Part A: Applied Science and Manufacturing*, vol. 39, no. 6, pp. 1037-1053, 2008.
- [26] G. Lebrun, N. Bureau and J. Denaut, "Evaluation of bias-extension and picture-frame test methods for the measurements of intraply shear properties of PP/glass commingled barics," *Composites Structures*, vol. 61, no. 4, pp. 341-352, 2003.
- [27] U. Mohammed, C. Lekakou, L. Dong and M. Bader, "Shear defomration and micromechanics of woven fabircs," *Composites Part A: applied science and manufacturing*, vol. 31, pp. 299-308, 2000.
- [28] B. Yang, T. Jin, F. Bi and J. Li, "A Geometry Information Based Fishnet Algorithm for Woven Fabric Draping," *Materials Science*, vol. 20, no. 4, pp. 513-521, 2014.
- [29] R. Weinstock, *Calculus of variations*, Dover: Dover Publications, 1974.
- [30] R. Robertson, E. Hsiue, E. Sickafus and G. S. Y. Yeh, "Fibre rearrangements during the molding of continuous fibre composites. I. Flat cloth to a hemisphere," *Polymer composites*, vol. 2, 1981.
- [31] A. S. Tam and T. G. Gutowski, "The kinematics for forming ideal aligned fibre composites into complex shapes," *Composite Manufacturing*, vol. 1, no. 4, 1990.
- [32] F. Van Der Weeen, "Algorithms for draping fabrics on doubly-curved surfaces," *International journal for numerical methods in engineering*, vol. 31, pp. 1415-1426, 1991.

- [33] A. Long, B. Souer, F. Robitaille and C. Rudd, "Effects of fibre architecture on reinforcement fabric deformation," *Plastics, Rubber and Composites*, vol. 31, no. 2, pp. 87-97, 2002.
- [34] A. Long and C. Rudd, "A simulation of reinforcement deformation during the production of preforms for liquid moulding processes," *Journal of engineering manufacture*, vol. 208, 1994.
- [35] N. Siddiqui, S. U. Khan, P. C. Ma, C. Y. Li and J. Kim, "Manufacturing and characterization of carbon fibre/epoxy composite prepregs containing carbon nanotubes," *Composites: Part A*, vol. 42, pp. 1412-1420, 2011.
- [36] J.-A. Manson, M. Wakeman and N. Bernet, "Composite Processing and Manufacturing - An Overview," in *Comprehensive Composite Materials*, Elsevier, 2000, pp. 577-607.
- [37] V. Michaud, "A Review of Non-saturated Resin Flow in Liquid Composite Moulding processes," *Transport in Porous Media*, vol. 115, no. 3, pp. 581-601, 2019.
- [38] E. Sozer, P. Simacek and S. Advani, "Resin transfer molding (RTM) in polymer matrix composites," in *Manufacturing Techniques for Polymer Matrix Composites (PMCs)*, Cambridge, Woodhead Publishing Limited, 2012, pp. 245-309.
- [39] K.-T. Hsiao and D. Heider, " Vacuum assisted resin transfer molding (VARTM) in polymer matrix composites," in *Manufacturing Techniques for Polymer Matrix Composites (PMCs)*, Cambridge, Woodhead Publishing Limited, 2012, pp. 310-347.
- [40] J. A. Jacquez, "Design of experiments," *Journal of the Franklin Institute*, vol. 335, no. 2, pp. 259-279, 1998.

- [41] S. Dowlatshahi, "An application of design of experiments for optimization of plastic injection molding processes," vol. 15, no. 6, 2004.
- [42] G. E. Box, J. Hunter and W. Hunter, *Statistics of experimenters: design, innovation and discovery.*, Hoboken: Wiley-Interscience, 2005.
- [43] R. Plackett and J. Burman, "The design of optimum multifactorial experiments," 1946.
- [44] Y. Heyden, A. Nijhuis, J. Smeyers-Verbeke, B. Vandeginste and D. Massart, "Guidance for robustness/ruggedness tests in method validation," *Journal of Pharmaceutical and Biomedical Analysis*, vol. 14, pp. 723-753, 2001.
- [45] K. Vanaja and R. Shobha Rani, "Design of experiments: Concepts and applications of Plackett-Burman design," *Clinical research and regulatory affairs*, vol. 24, no. 1, pp. 1-23, 2007.
- [46] D. J. Wheeler, *Understanding Industrial Experimentation*, Knoxville: SPC Press, Inc, 1987.
- [47] Merriam-Webster, "Definition of scree," Merriam-Webster, [Online]. Available: <https://www.merriam-webster.com/dictionary/scree>. [Accessed 17 11 2019].
- [48] S. Baril-Gosselin, "Statistical Design of Experiments for the Aerodynamic Study of the Supermileage Vehicle of the University of Ottawa," University of Ottawa, Ottawa, 2010.
- [49] I. El Karafi, *Geometric modeling and heat transfer simulations for the reverse engineering of an aerospace carbon/epoxy back seat tool*, Ottawa: University of Ottawa, 2009.

- [50] Immersion, "MicroScribe G2," 17 03 2009. [Online]. Available: http://www.remtek.com/remtek/microscribe_rhino.htm. [Accessed 24 09 2019].
- [51] D. Wheeler, Tables of Screening Designs, Knoxville: SPC Press, 1987.
- [52] ANSYS, Inc., ANSYS Mechanical APDL Command Reference, Canonsburg: SAS IP, Inc, 2013.
- [53] The MathWorks Inc, *MATLAB and Statistics Toolbox Release*, Natick, 2017b.
- [54] K. Potter, "The influence of accurate stretch data for reinforcements on the production of complex structural mouldings," *Composites*, vol. 11, 1979.

Appendix A

Remodelling Base Surfaces using ANSYS

A.1 Basic essential ANSYS commands

The purpose of this Appendix is to introduce a user to basic ANSYS commands used to build base surfaces and provides one basic example and one complex example of this. ANSYS, specifically Mechanical APDL, is used to remodel base surfaces from conical frustum parameters. The simplest way to remodel the surface consists in using a text file to write a series of command prompts that build the surface and mesh. To start, a number of essential commands are outlined below, as obtained from the Mechanical APDL Command Reference [52]. These commands require the COMMAND function followed by *user inputs* to be read.

- /CLC, /CLEAR: resets all modelling
- /PREP7: Sets ANSYS into preprocessing, allowing the commands following to be initiated
- ET, *ITYPE*, *Ename*, *KOP1*, *KOP2*...: Defines a local element type from the element library
 - *ITYPE*: element reference number (1)
 - *Ename*: sets the element type for the model – the element should always be set to SHELL181 for building base surfaces
 - *KOP*: key options used to control element behavior (not assigned)
- K, *NPT*, *X*, *Y*, *Z*: Defines a key point
 - *NPT*: reference number for the key point defined at *X*, *Y*, *Z*
 - This is used to set individual key points (such as A, B, C and D from the conical frustum) within the model, that will be linked together by lines to generate areas
- LSTR, *P1*, *P2*: Defines a straight line irrespective of the active coordinate system
 - *P1*: starting key point
 - *P2*: ending key point

- LARC, *PI*, *P2*, *PC*, *RAD*: Defines a circular arc
 - *PC*: key point defining plane of arc and centre of curvature side (positive radius) that does not have to be in the centre of curvature
 - *RAD*: radius of arc
 - Note: The line number is assigned sequentially by input and not specified by the user
- AL, *L1*, *L2*, *L3*, *L4*: Generates an area bounded by lines
 - *L1*, *L2*, ...: List of lines defining the area (minimum 3)
- AESIZE, *ANUM*, *SIZE*: specifies the element size to be meshed
 - *ANUM*: should be set to “all” since we want all the areas to be meshed at the same size
 - *SIZE*: desired element size based on component size. Smaller element sizes yield more refined meshes and therefore more elements and nodes. Large element sizes tend to deform the geometry
- MSHAPE, *Key*, *Dimension*: Sets element shape to be used for meshing
 - *Key*: Indicates element shape. Always set to 1 for triangular meshes
 - *Dimension*: Specifies model dimension
- MSHKEY, *Key*: specifies whether free or mapped meshing should be used
 - *Key*: Always set to 1 for mapped meshing
- AMESH, *NA1*, *NA2*, *NINC*: generates nodes and area elements within defined areas
 - *NA1*, *NA2*, *NINC*: Mesh areas from *NA1* to *NA2* in steps of *NINC*. Setting this to *ALL* simply chooses all surfaces

The text file used for command prompts follows a standard outline.

1. Starting prompts (/CLC, /CLEAR, /PREP7)
2. Element selection (ET)
3. Key points, lines, arcs
4. Building areas
5. Meshing

A.2 Five parameter base surface remodelling example

Given the basic conical frustum parameters below, a command script can be written to build and mesh the surface in ANSYS.

Table A-1: Basic conical frustum parameters

Conical Frustum Parameters	X	Y	Z
$A=B$	-1.414	2	-1.414
C	0.586	0	-1.414
D	-1.414	0	0.586
$ \vec{R}_1 $	0 cm		
$ \vec{R}_2 $	2 cm		
$ \vec{L} $	2 cm		
C_R	2 cm		
β	90°		

The centre point for ellipse 2 is added as a key point during modelling to define the plane of the arc and the centre of curvature (PC) which are determined during conical frustum calculations.

```
/clc  
/clear  
/prep7
```

1. Starting prompts reset
ANSYS modelling space
and enters preprocessing

```
ET,1,shell1181
```

2. Enables shell elements
to be modelled

```
K,1,-1.414,0.000,-1.414  
K,2,-1.414,2.000,-1.414  
K,3,0.586,0.000,-1.414  
K,4,-1.414,0.000,0.586
```

3. Key point 1 is the centre point
for ellipse 2
Key points 2 – 4 are locations of *A*,
C, and *D*

```
LARC,2,3,1,2.000  
LARC,2,4,1,2.000  
LARC,3,4,1,2.000
```

4. Arcs drawn between the key points 2 and 3, 2
and 4, and 3 and 4, all using key point 1 as the
arc reference. All radii are equal to 2 cm (C_r)

```
AL, 1,2,3
```

4. Area build from lines 1, 2, 3

```
AESIZE,ALL,0.1275  
MSHAPE,1,2D  
MSHKEY,1  
AMESH,ALL
```

5. Surface meshed

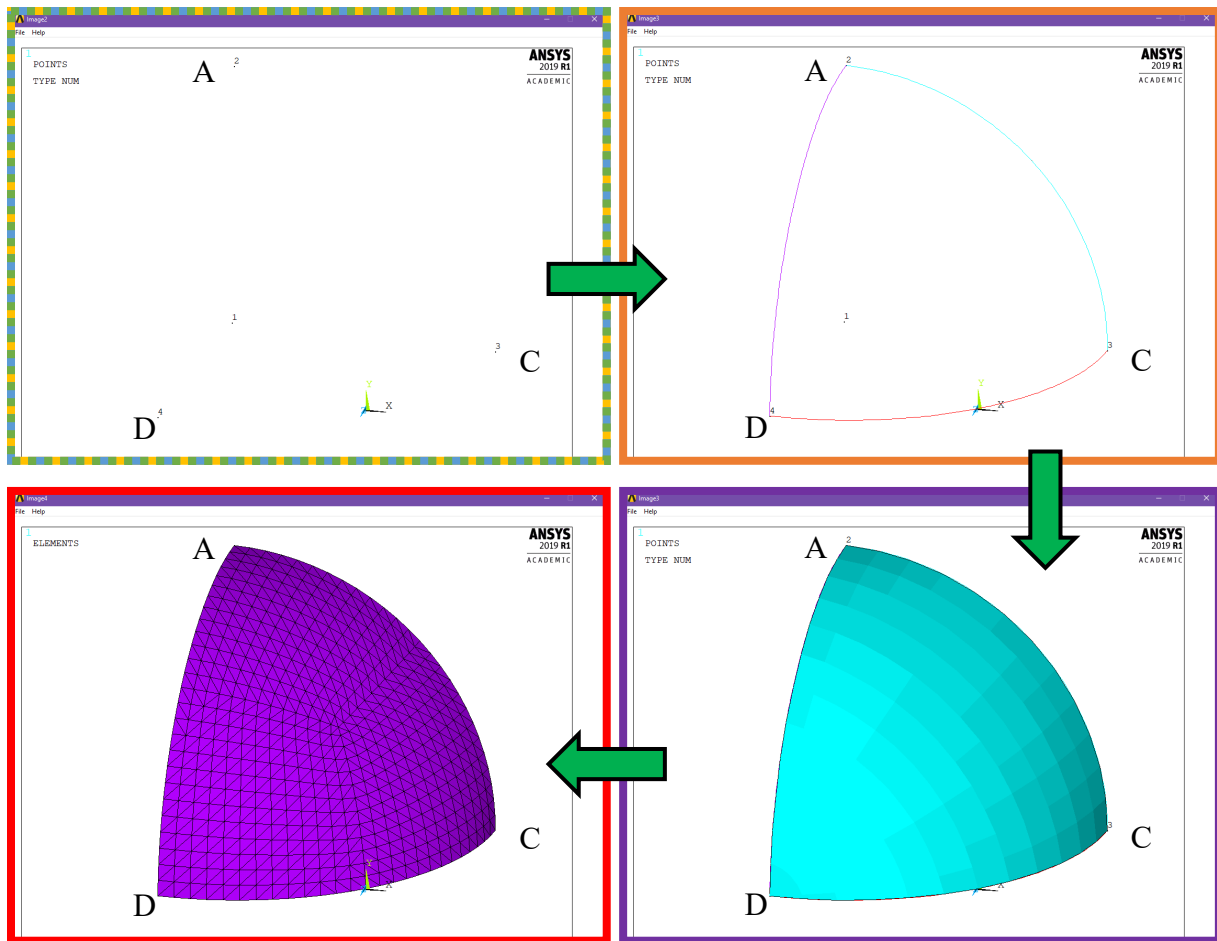


Figure A-1: Five parameter frustum modeling sequence: key points, lines, area, mesh

A.3 12 parameter base surface remodelling example

A few additional commands may be necessary when remodelling base surfaces using the 12-parameter frustum.

- *LOCAL, KCN, KCS, XC, YC, ZC, THXY, THYZ, THZX, PAR1, PAR2*: defines a new local coordinates system by location and orientation
 - *KCN*: coordinate system reference (must be greater than 10)
 - *KCS*: coordinate type (in most cases this will be 2 or SPHE for spherical)
 - *XC, YC, ZC*: location, in global coordinate system, of the origin of the new coordinates system
 - *THXY*: first rotation about local Z (positive X toward Y)
 - *THYZ*: second rotation about local X (positive Y toward Z)
 - *THZX*: third rotation about local Y (positive Z toward X)
 - Axes remain perpendicular during rotations. Rules for calculating THXY, THYZ, THZX can be found in section A.4.
 - *PAR1*: used for elliptical systems; this is the ratio of the ellipse Y-axis radius to X-axis radius in the new local coordinate system
 - *PAR2*: used for elliptical systems; this is the ratio of the ellipse Z-axis radius to X-axis radius

It is important to note that once the local coordinates system is defined, it becomes the active coordinate system. In the context of remodelling base surfaces, the main purpose of the local coordinate system is to establish centre points for ellipses and should be used in conjunction with command L (below) as the final set of commands before generating an area and the mesh.

- *L, P1, P2, NDIV, SPACE, XV1, YV1, ZV1, XV2, YV2, ZV2*
 - *P1*: Key point at the beginning of the line
 - *P2*: Key point at the end of the line
 - *NDIV*: number of element divisions along line (no need to set if using LESIZE)
 - *SPACE*: spacing ratio for elements (no need to set if using LESIZE)
 - *XV1, YV1, ZV1*: location of head of slope vector corresponding to the slope at the *P1* end of the line in active coordinate system (no need to set)
 - *XV2, YV2, ZV2*: location of head of slope vector corresponding to the slope at the *P2* end of the line (do not need to set)

Larger base surfaces may require extra key points and line segments that are stitched together, since the L command may exaggerate some features. Given the frustum parameters in Table A-2 below, a series of command prompts are written into a text file to generate the meshed surface in following sections.

Table A-2: 12 parameter conical frustum parameters

Conical Frustum Parameters	X	Y	Z
A	-1	11	3
B	0	10	0
C	2	0	0
D	0	0	-0.5
$ \vec{R_1} $	2.449 cm		
$ \vec{r_1} $	2.236 cm		
$ \vec{R_2} $	2 cm		
$ \vec{r_2} $	0.5 cm		
$ \vec{L} $	10.25 cm		
C_R	25 cm		
C_T	25 cm		
β_1	90°		
β_2	90°		
θ_1	155.91°		
θ_2	167.40°		
α	90°		
Extra known information			
C_1	-2	10	1
C_2	0	0	0
N_1	-1	5	-2
N_2	0	-1	0

A.4 Calculating *THXY*, *THYZ*, *THZX* for a new local coordinate system

As defined in the LOCAL command, *THXY*, *THYZ*, and *THZX* are individual angular rotations that can be set to define a new local coordinate system. They occur in the sequence presented.

These rotations take place in reference to the original origin vectors defined $\vec{V}_X = (1,0,0)$, $\vec{V}_Y = (0,1,0)$, and $\vec{V}_Z = (0,0,1)$. To start, the normal vector for Ellipse 1 is split into each of its components in each plane and the magnitude of the component vectors are calculated. These components outlined in Figure A-2 are projections of the normal vector on the *xy*, *yx* and *zx* planes.

For $|\vec{N}_1|$ in the example above:

$$\vec{N}_1 = (-1, 5, -2)$$

$$\vec{N}_{1xy} = (-1, 5), |\vec{N}_{1xy}| = 5.099$$

$$\vec{N}_{1yz} = (5, -2), |\vec{N}_{1yz}| = 5.385$$

$$\vec{N}_{1zx} = (-2, -1), |\vec{N}_{1zx}| = 2.236$$

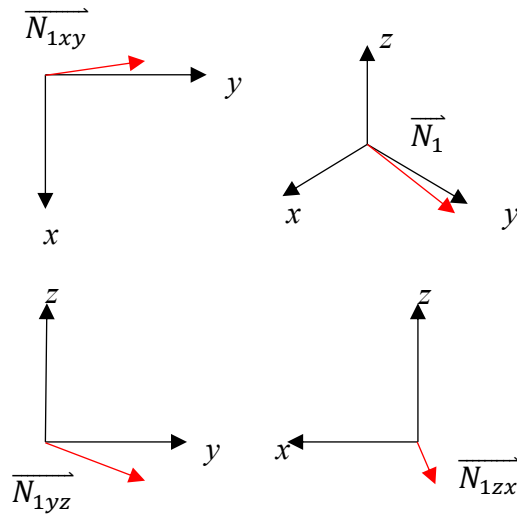


Figure A-2: Planar vectors components

Similarly, the component vectors for the origin vectors are also projected onto each plane.

$$\overrightarrow{V_X} = (1, 0, 0)$$

$$\overrightarrow{V_{X-xy}} = (1, 0), |\overrightarrow{V_{X-xy}}| = 1$$

$$\overrightarrow{V_{X-yz}} = (0, 0), |\overrightarrow{V_{X-yz}}| = 0$$

$$\overrightarrow{V_{X-zx}} = (0, 1), |\overrightarrow{V_{X-zx}}| = 1$$

$$\overrightarrow{V_Y} = (0, 1, 0)$$

$$\overrightarrow{V_{Y-xy}} = (0, 1), |\overrightarrow{V_{Y-xy}}| = 1$$

$$\overrightarrow{V_{Y-yz}} = (1, 0), |\overrightarrow{V_{Y-yz}}| = 1$$

$$\overrightarrow{V_{Y-zx}} = (0, 0), |\overrightarrow{V_{Y-zx}}| = 0$$

$$\overrightarrow{V_Z} = (0, 0, 1)$$

$$\overrightarrow{V_{Z-xy}} = (0, 0), |\overrightarrow{V_{Z-xy}}| = 0$$

$$\overrightarrow{V_{Z-yz}} = (0, 1), |\overrightarrow{V_{Z-yz}}| = 1$$

$$\overrightarrow{V_{Z-zx}} = (1, 0), |\overrightarrow{V_{Z-zx}}| = 1$$

The subscript sequence is always the sequence of vector components. Using the Equation and rules below, *THXY*, *THYZ*, and *THZX* are calculated. Figure A-3, Figure A-4 and Figure A-5 show each rotation in sequence changing from the original orientation to a new final orientation defined by $\overrightarrow{N_1}$ for ellipse 1. This coordinate system rotation allows lines to be draw between keypoints on the ellipse 1 plane.

$$\cos(\theta_{THXY}) = \left(\frac{\overrightarrow{V_X} \cdot \overrightarrow{N_{1xy}}}{|\overrightarrow{V_X}| |\overrightarrow{N_{1xy}}|} \right) \quad (23)$$

$$\cos(\theta_{THYZ}) = \left(\frac{\overrightarrow{V_Y} \cdot \overrightarrow{N_{1yz}}}{|\overrightarrow{V_Y}| |\overrightarrow{N_{1yz}}|} \right) \quad (24)$$

$$\cos(\theta_{THZX}) = \left(\frac{\overrightarrow{V_Z} \cdot \overrightarrow{N_{1zx}}}{|\overrightarrow{V_Z}| |\overrightarrow{N_{1zx}}|} \right) \quad (25)$$

1st Rotation: *THXY*

1. If the *X* component and *Y* component of $\overrightarrow{N_1}$ are both positive, use $\overrightarrow{V_{X-xy}}$, the *XY* component of $\overrightarrow{V_X}$.
2. If the *X* component and *Y* component of $\overrightarrow{N_1}$ are both negative, use $\overrightarrow{V_{X-xy}}$, the *XY* component of $\overrightarrow{V_X}$.
3. If the *X* component of $\overrightarrow{N_1}$ is positive and *Y* component of $\overrightarrow{N_1}$ is negative, use $\overrightarrow{V_{X-xy}}$, the *XY* component of $\overrightarrow{V_X}$ but add a negative to the resulting angle.
4. If the *X* component of $\overrightarrow{N_1}$ is negative and *Y* component of $\overrightarrow{N_1}$ is positive, use $\overrightarrow{V_{X-xy}}$, the *XY* component of $\overrightarrow{V_X}$.

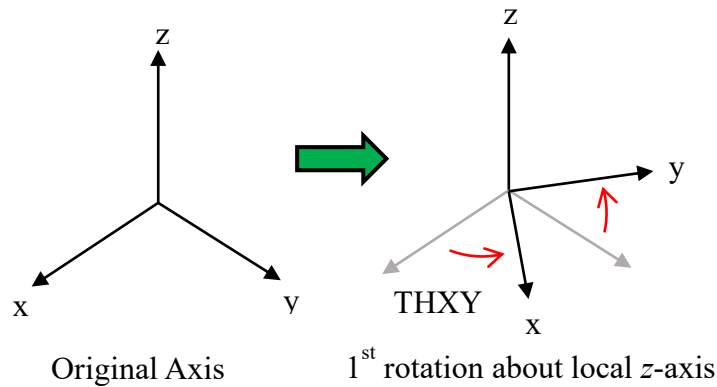


Figure A-3: 1st axis rotation: *THXY*

2nd Rotation: *THYZ*

1. If the Y component and Z component of $\vec{N}_1$ are both positive, use $\vec{V}_{Y-YZ}$, the YZ component of $\vec{V}_Y$.
2. If the Y component and Z component of $\vec{N}_1$ are both negative, use $\vec{V}_{Y-YZ}$, the YZ component of $\vec{V}_Y$.
3. If the Y component of $\vec{N}_1$ is positive and Z component of $\vec{N}_1$ is negative, use $\vec{V}_{Y-YZ}$, the YZ component of $\vec{V}_Y$ but add a negative to the resulting angle.
4. If the Y component of $\vec{N}_1$ is negative and Z component of $\vec{N}_1$ is positive, use $\vec{V}_{Y-YZ}$, the YZ component of $\vec{V}_Y$.

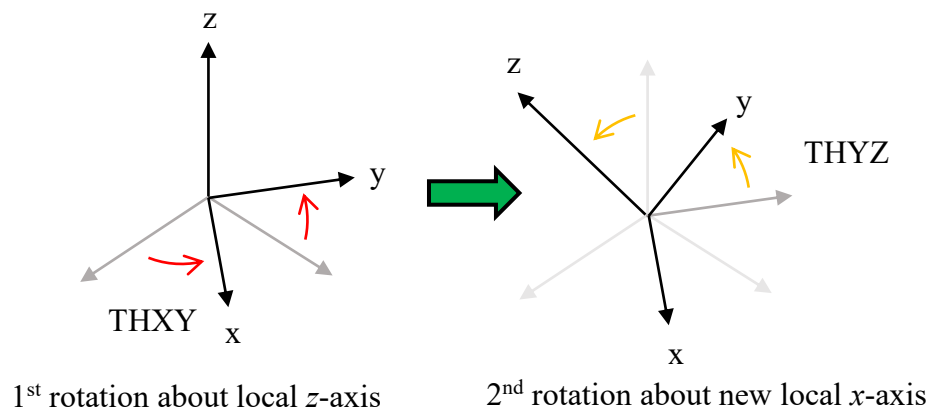


Figure A-4: 2nd axis rotation: *THYZ*

3rd Rotation: *THZX*

1. If the Z component and X component of $\vec{N}_1$ are both positive, use $\vec{V}_{Z-ZX}$, the ZX component of $\vec{V}_Z$.
2. If the Z component and X component of $\vec{N}_1$ are both negative, use $\vec{V}_{Z-ZX}$, the ZX component of $\vec{V}_Z$.
3. If the Z component of $\vec{N}_1$ is positive and X component of $\vec{N}_1$ is negative, use $\vec{V}_{Z-ZX}$, the ZX component of $\vec{V}_Z$ but add a negative to the resulting angle.
4. If the Z component of $\vec{N}_1$ is negative and X component of $\vec{N}_1$ is positive, use $\vec{V}_{Z-ZX}$, the ZX component of $\vec{V}_Z$.

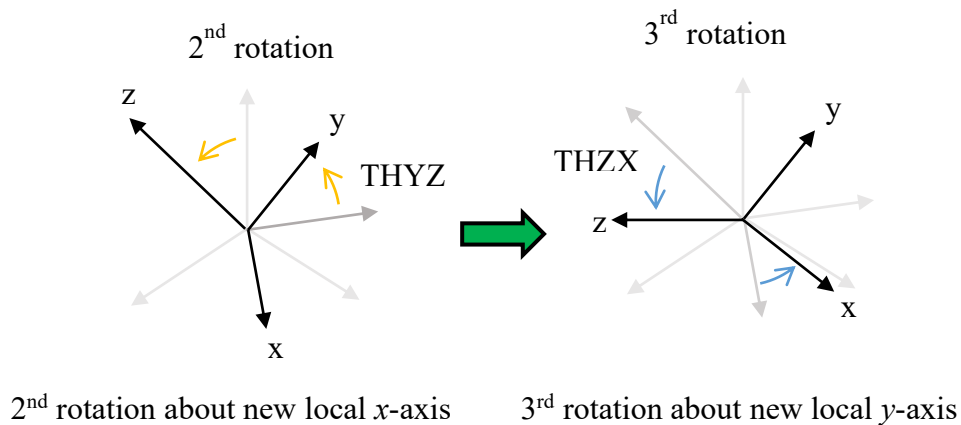


Figure A-5: 3rd axis rotation: *THZX*

A.5 12 parameter frustum ANSYS command textfile

```
/clc
/clear
/prep7
ET,1, shell1181
!NOTE: exclamation points denote comments and aren't read by ANSYS
!Keypoints
K,1.0,0.000,10.000,0.000
K,2.0,0.405,5.015,-0.553
K,3.0,-0.000,-0.000,-0.500
K,4.0,0.090,10.100,0.205
K,5.0,0.596,5.073,-0.400
K,6.0,0.200,0.000,-0.497
K,7.0,0.160,10.200,0.420
K,8.0,0.768,5.130,-0.239
K,9.0,0.400,0.000,-0.490
K,10.0,0.208,10.300,0.646
K,11.0,0.923,5.187,-0.071
K,12.0,0.600,-0.000,-0.477
K,13.0,0.233,10.400,0.883
K,14.0,1.062,5.243,0.107
K,15.0,0.800,-0.000,-0.458
K,16.0,0.232,10.500,1.134
K,17.0,1.184,5.300,0.294
K,18.0,1.000,-0.000,-0.433
K,19.0,0.200,10.600,1.400
K,20.0,1.287,5.356,0.494
K,21.0,1.200,-0.000,-0.400
K,22.0,0.128,10.700,1.686
K,23.0,1.368,5.413,0.712
K,24.0,1.400,0.000,-0.357
K,25.0,0.000,10.800,2.000
K,26.0,1.417,5.469,0.958
```

K,27.0,1.600,0.000,-0.300
 K,28.0,-0.228,10.900,2.364
 K,29.0,1.412,5.523,1.256
 K,30.0,1.800,-0.000,-0.218
 K,31.0,-1.000,11.000,3.000
 K,32.0,1.103,5.566,1.860
 K,33.0,2.000,0.000,0.000

!Arcs

LARC,1.0,3.000,2.000,-25.000
 LARC,4.0,6.000,5.000,-25.000
 LARC,7.0,9.000,8.000,-25.000
 LARC,10.0,12.000,11.000,-25.000
 LARC,13.0,15.000,14.000,-25.000
 LARC,16.0,18.000,17.000,-25.000
 LARC,19.0,21.000,20.000,-25.000
 LARC,22.0,24.000,23.000,-25.000
 LARC,25.0,27.000,26.000,-25.000
 LARC,28.0,30.000,29.000,-25.000
 LARC,31.0,33.000,32.000,-25.000

!New Coordinate System for Ellipse 1

LOCAL,11.0,2.0,-2.0,10.0,1.0,101.309,-21.8,0.00,1.00000,1.09545

!Lines connecting sections on each plane

L,1,4
 L,4,7
 L,7,10
 L,10,13
 L,13,16
 L,16,19
 L,19,22
 L,22,25
 L,25,28
 L,28,31

```
!New Coordinate System for Ellipse 2
LOCAL,12.0,1.0,0.0,0.0,0.0,0.0,90.0,0.00,0.25000,1.00000
```

```
L,3,6
L,6,9
L,9,12
L,12,15
L,15,18
L,18,21
L,21,24
L,24,27
L,27,30
L,30,33
```

```
!Areas
AL,1,12,2,22
AL,2,13,3,23
AL,3,14,4,24
AL,4,15,5,25
AL,5,16,6,26
AL,6,17,7,27
AL,7,18,8,28
AL,8,19,9,29
AL,9,20,10,30
AL,10,21,11,31
```

```
!Ordered Meshing
AESIZE,ALL,0.5
MSHAPE,1,2D
MSHKEY,1
AMESH,ALL
```

Appendix B

ANSYS to Nottingham Drape It

B.1 Nottingham DrapeIt input geometry file overview

The purpose of this appendix is to take a base surface modelled in ANSYS and create a mesh compatible with Nottingham DrapeIt. Appendix A outlined the steps to build such meshes, but conversion into the Nottingham drape format file (.ncd) is required for importing the surface into DrapeIt. These imported surfaces will allow simulations to be performed and the database to be populated. The .ncd files are formatted in the following sequence:

- Maximum and minimum volumetric coordinates in which the mesh is defined
 - This does not have to be exact so long as the volume is greater than the mesh
- Total number of nodes
- List of nodes and their respective Cartesian coordinates, in sequence and without any nodes missing
- Total number of elements
- List of elements and the nodes associated with each element, in sequence and without any elements missing

B.2 Node Extraction and Organization Steps

Once a base surface is modelled and meshed in ANSYS, nodes are extracted and organized through the following steps:

1. Select the List tab → “Nodes”
2. Check “Coordinates Only” when the Sort NODE Listing window opens → OK

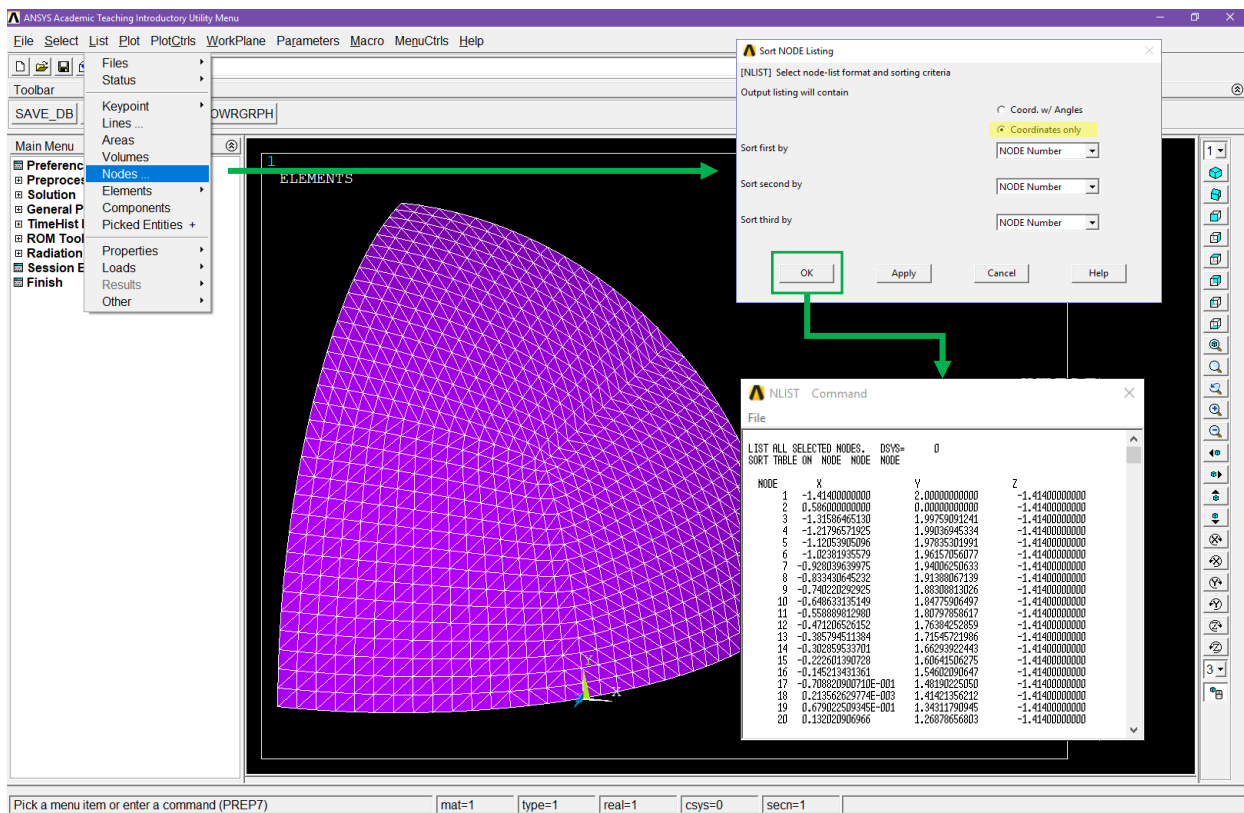


Figure B-1: Node data extraction

3. When the NLIST (node list) window appears, copy all the contents and paste into an Excel spreadsheet using the Text Import Wizard found under Paste in the Home tab of Excel
 - a. Ensure delimited is selected → Next
 - b. Choose “Space” delimiter → Next
 - c. Column data format “General” → Finish

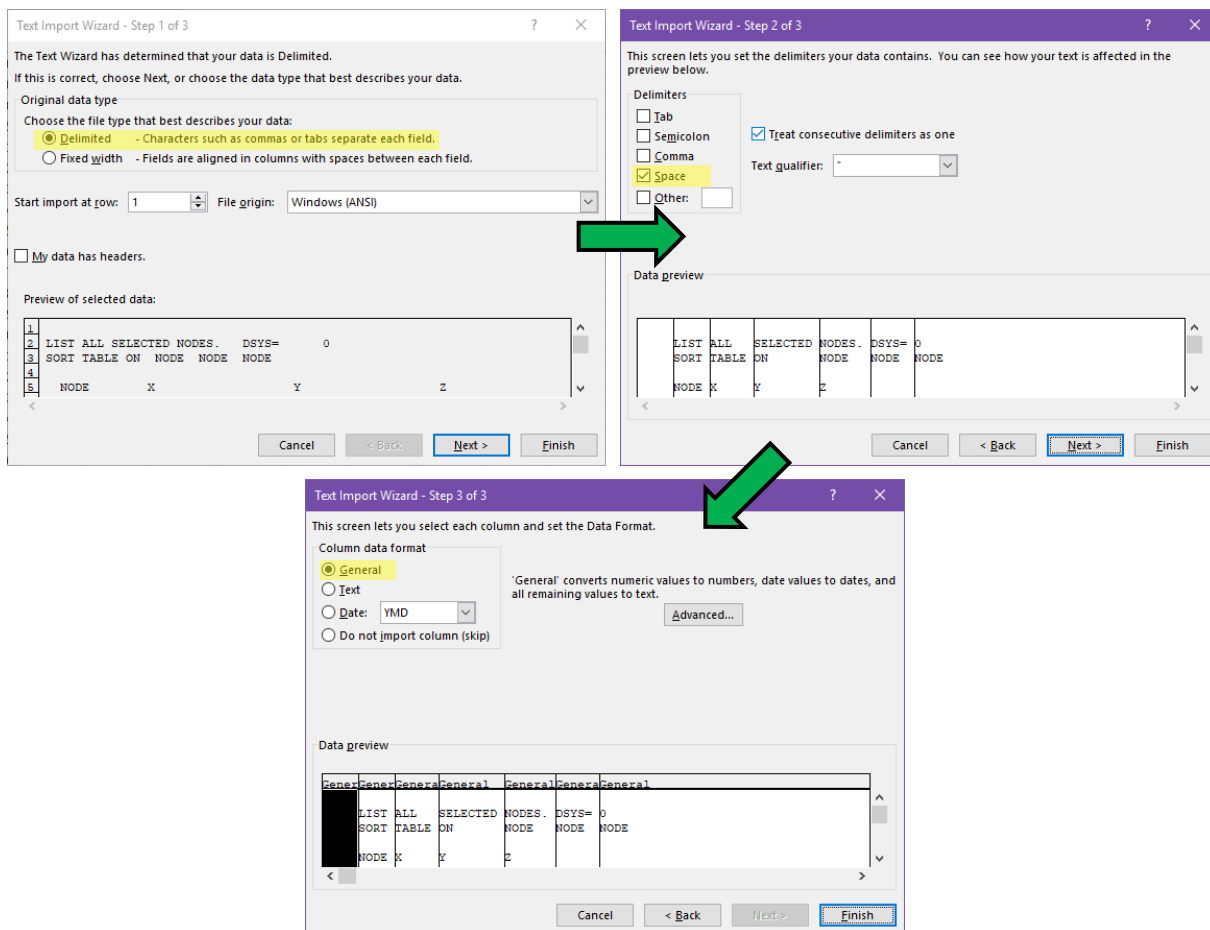


Figure B-2: Importing nodal data into Excel

- The node data will be expressed as Node Number, X, Y and Z coordinates in Excel

	A	B	C	D	E	F	G	H
1								
2		LIST	ALL	SELECTED	NODES.	DSYS=		0
3		SORT	TABLE	ON	NODE	NODE	NODE	
4								
5		NODE	X	Y	Z			
6		1	-1.414	2	-1.414			
7		2	0.586	0	-1.414			
8		3	-1.31586	1.997591	-1.414			
9		4	-1.21797	1.990369	-1.414			
10		5	-1.12054	1.978353	-1.414			
11		6	-1.02382	1.961571	-1.414			
12		7	-0.92804	1.940063	-1.414			
13		8	-0.83343	1.913881	-1.414			
14		9	-0.74022	1.883088	-1.414			
15		10	-0.64863	1.847759	-1.414			
16		11	-0.55889	1.807979	-1.414			
17		12	-0.47121	1.763843	-1.414			
18		13	-0.38579	1.715457	-1.414			
19		14	-0.30286	1.662939	-1.414			
20		15	-0.2226	1.606415	-1.414			
21		16	-0.14521	1.546021	-1.414			
22		17	-7.09E-02	1.481902	-1.414			
23		18	2.14E-04	1.414214	-1.414			
24		19	6.79E-02	1.343118	-1.414			
25		20	0.132021	1.268787	-1.414			
26								
27		NODE	X	Y	Z			
28		21	0.192415	1.191399	-1.414			
29		22	0.248939	1.11114	-1.414			

Figure B-3: Copied node data from ANSYS

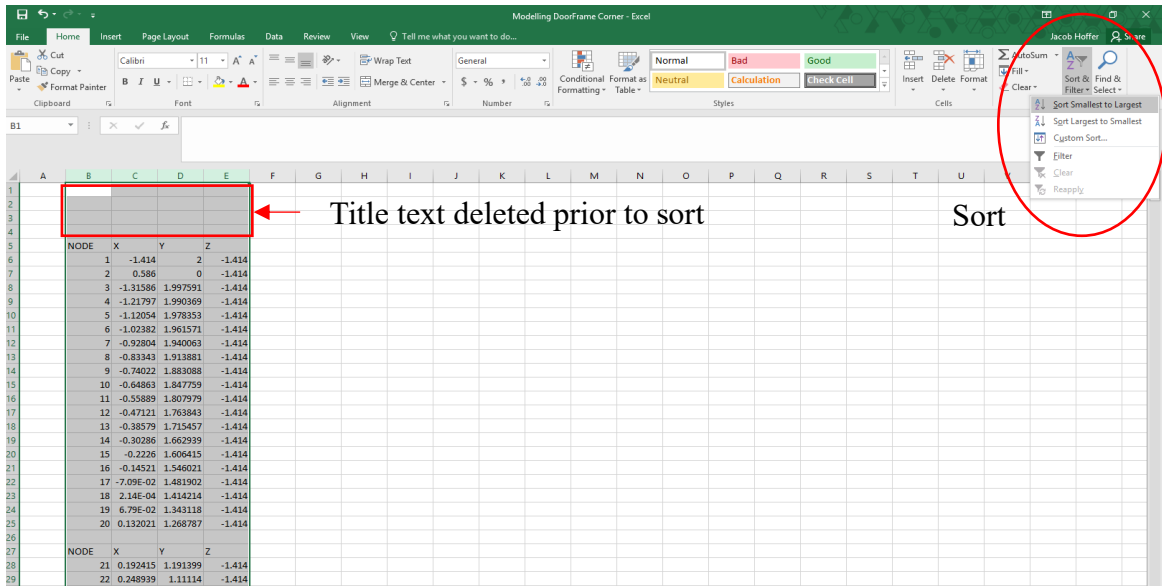
- Select all the columns → Select Sort & Filter in the Home Tab → Smallest to Largest

Note: All text needs to be deleted during this process – once the data is sorted, all the text should be at the bottom of the data. Nodes are listed in groups of 20 with new titles and spaces between each series. For the .ncd to function correctly there can be no spaces or text between the sequential node data

- Add 4 rows above the node data and input the volume coordinates that delimit the mesh

Note: These limits do not have to be exact, as long as they are larger than the base surface.

- In the 4th row directly above the node number, input the total number of nodes, the number at the bottom of the list.
- At this stage, make sure to delete any text in the spreadsheet.



		Max	Min		
	A	B	C	Rows added for volumetric coordinates	
1	X	20	-20		
2	Y	20	-20		
3	Z	20	-20		
4		817		Total Number of Nodes	
5		1	-1.414	2	-1.414
6		2	0.586	0	-1.414
7		3	-1.31586	1.997591	-1.414
8		4	-1.21797	1.990369	-1.414
9		5	-1.12054	1.978353	-1.414
10		6	-1.02382	1.961571	-1.414
11		7	-0.92804	1.940063	-1.414
12		8	-0.83343	1.913881	-1.414
13		9	-0.74022	1.883088	-1.414
14		10	-0.64863	1.847759	-1.414
...					
808		804	-0.24117	1.17283	-0.49603
809		805	-0.21547	1.198528	-0.56201
810		806	-0.19084	1.22316	-0.6292
811		807	-0.1673	1.246699	-0.69744
812		808	-0.14488	1.269117	-0.76662
813		809	-0.12361	1.290389	-0.83662
814		810	-0.10352	1.310475	-0.90734
815		811	-8.47E-02	1.329316	-0.97868
816		812	-6.71E-02	1.346856	-1.05053
817		813	-5.09E-02	1.363056	-1.12276
818		814	-3.61E-02	1.37789	-1.19524
819		815	-2.26E-02	1.391356	-1.26785
820		816	-1.05E-02	1.403486	-1.34063
821		817	-0.31137	1.102633	-0.33407
822	NODE	X	Y	Z	Delete all text below node data
823	NODE	X	Y	Z	
824	NODE	X	Y	Z	

Figure B-4: Organizing nodal data in Excel

B.3 Element Extraction and Organization Steps

A similar process is used for extracting and organizing element information:

1. Select the List tab → “Elements” → “Nodes & Attributes”
2. When the ELIST (element list) window appears, copy all the contents and paste into the same Excel spreadsheet, **below** the node information (same steps as 3, a), b) and c) in the Node Extraction and Organization Steps Section). Note: Leave at least one row between the end of the Node data and the beginning of the Element data

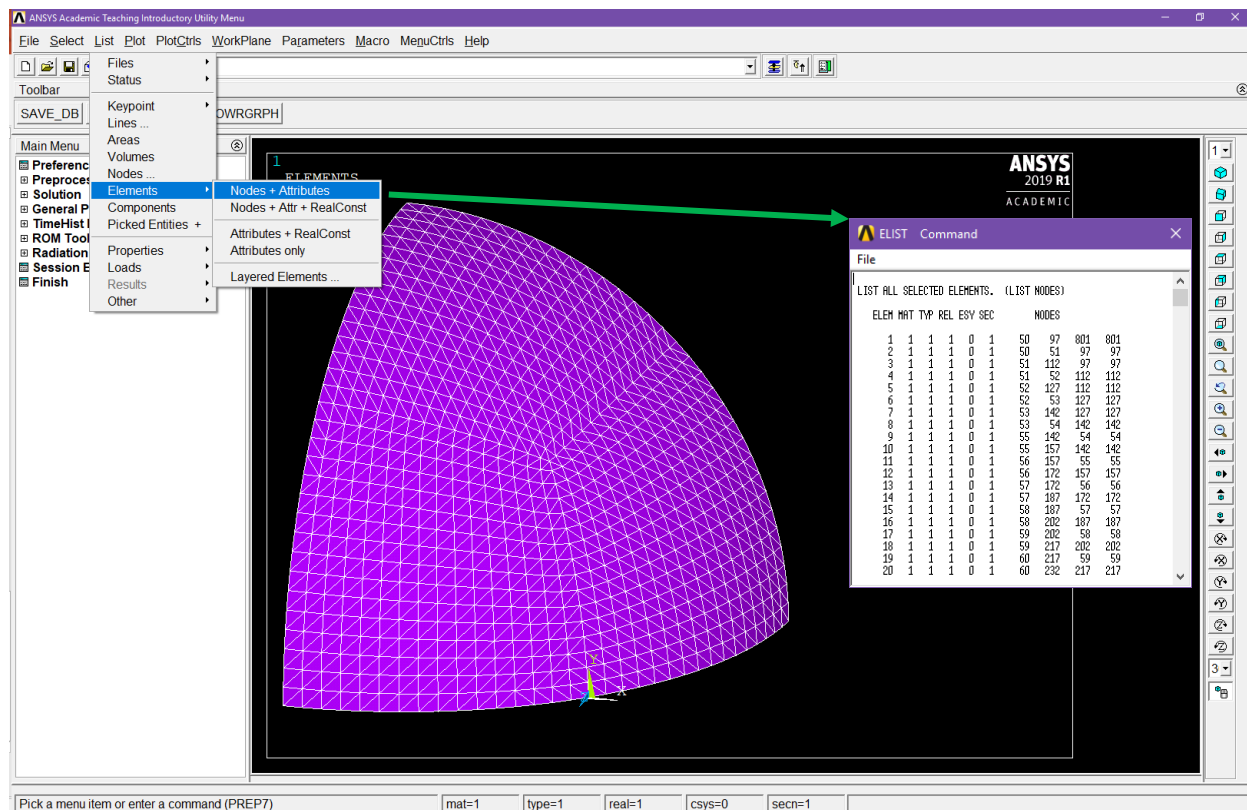


Figure B-5: Element data extraction

3. Select all the columns → Select Sort & Filter in the Home tab → Smallest to Largest
 - a. Like the node data, elements are listed in groups of 20 with new titles and spaces between each series. For the .ncd to function correctly there can be no spaces or text between the sequential element data
4. Delete MAT, TYP, REL, ESY, SEC columns in the element data (contain 1s and 0s exclusive) – these are not needed in the .ncd file. Note: Like for the node data, all text must be deleted from the spreadsheet
5. In the row directly after the node data and before the element data, input the total number of elements
6. Remove any additional columns/spaces between all data. At this point the spreadsheet is complete

A	B	C	D	E	F	G	H	I	J	K	L
814	810	-0.10352	1.310475	-0.90774							
815	811	-8.47E-02	1.329316	-0.97868							
816	812	-6.71E-02	1.346856	-1.05053							
817	813	-5.09E-02	1.363056	-1.12276							
818	814	-3.61E-02	1.37789	-1.19524							
819	815	-2.26E-02	1.391356	-1.26785							
820	816	-1.05E-02	1.403486	-1.34063							
821	817	-0.31137	1.102633	-0.33407							
822											
823	LIST	ALL	SELECTED	ELEMENTS (LIST	NODES)						
824	ELEM	MAT	TYP	REL	ESY	SEC	NODES				
825											
826											
827	1	1	1	1	0	1	50	97	801	801	
828	2	1	1	1	0	1	50	51	97	97	
829	3	1	1	1	0	1	51	112	97	97	
830	4	1	1	1	0	1	51	52	112	112	
831	5	1	1	1	0	1	52	127	112	112	
832	6	1	1	1	0	1	52	53	127	127	
833	7	1	1	1	0	1	53	142	127	127	
834	8	1	1	1	0	1	53	54	142	142	
835	9	1	1	1	0	1	55	142	54	54	
836	10	1	1	1	0	1	55	157	142	142	
837	11	1	1	1	0	1	56	157	55	55	
838	12	1	1	1	0	1	56	172	157	157	
839	13	1	1	1	0	1	57	172	56	56	
840	14	1	1	1	0	1	57	187	172	172	

A	B	C	D	E	F	G	H	I	J	K	L
818	814	-3.61E-02	1.37789	-1.19524							
819	815	-2.26E-02	1.391356	-1.26785							
820	816	-1.05E-02	1.403486	-1.34063							
821	817	-0.31137	1.102633	-0.33407							
822											
823	1	1	1	1	0	1		50	97	801	801
824	2	1	1	1	0	1		50	51	97	97
825	3	1	1	1	0	1		51	112	97	97
826	4	1	1	1	0	1		51	52	112	112
827	5	1	1	1	0	1		52	127	112	112
828	6	1	1	1	0	1		52	53	127	127
829	7	1	1	1	0	1		53	142	127	127
830	8	1	1	1	0	1		53	54	142	142
831	9	1	1	1	0	1		55	142	54	54
832	10	1	1	1	0	1		55	157	142	142
833	11	1	1	1	0	1		56	157	55	55
834	12	1	1	1	0	1		56	172	157	157

Delete these 5 columns
(MAT, TYP, REL, ESY, SEC)

A	B	C	D	E	F
818	814	-3.61E-02	1.37789	-1.19524	
819	815	-2.26E-02	1.391356	-1.26785	
820	816	-1.05E-02	1.403486	-1.34063	
821	817	-0.31137	1.102633	-0.33407	
822	1536				
823	1	50	97	801	801
824	2	50	51	97	97
825	3	51	112	97	97
826	4	51	52	112	112
827	5	52	127	112	112
828	6	52	53	127	127
829	7	53	142	127	127
830	8	53	54	142	142
831	9	55	142	54	54

Total number of elements

Figure B-6: Organizing element data in Excel

- To create the .ncd file, simply copy all the data into a text file → Save As → Select all files → [filename].ncd → Save

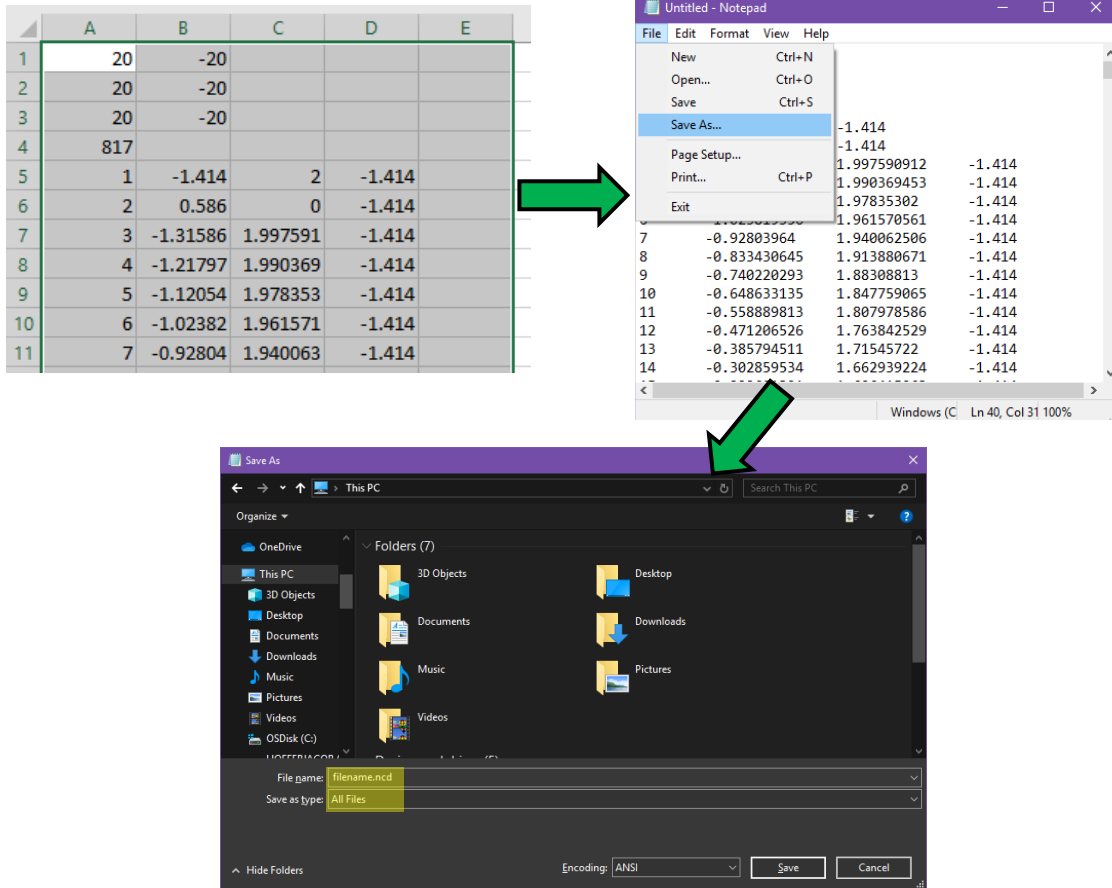


Figure B-7: Saving .ncd files

At this point it should be possible to open the file in DrapeIt without geometric or technical issues. It is advised to save the spreadsheet used for generating the .ncd file as it is used at a later stage for data organization.

Appendix C

Running Simulations and Extracting Data

C.1 Running draping simulations in Drape It

Now that compatible models of base surfaces can be loaded into Drape It, draping simulations can be performed. To run these simulations, follow these steps:

1. Load Geometry into Drape It
 - a. Be sure to orient the surface in the desired position for draping, and have a draping starting point in mind

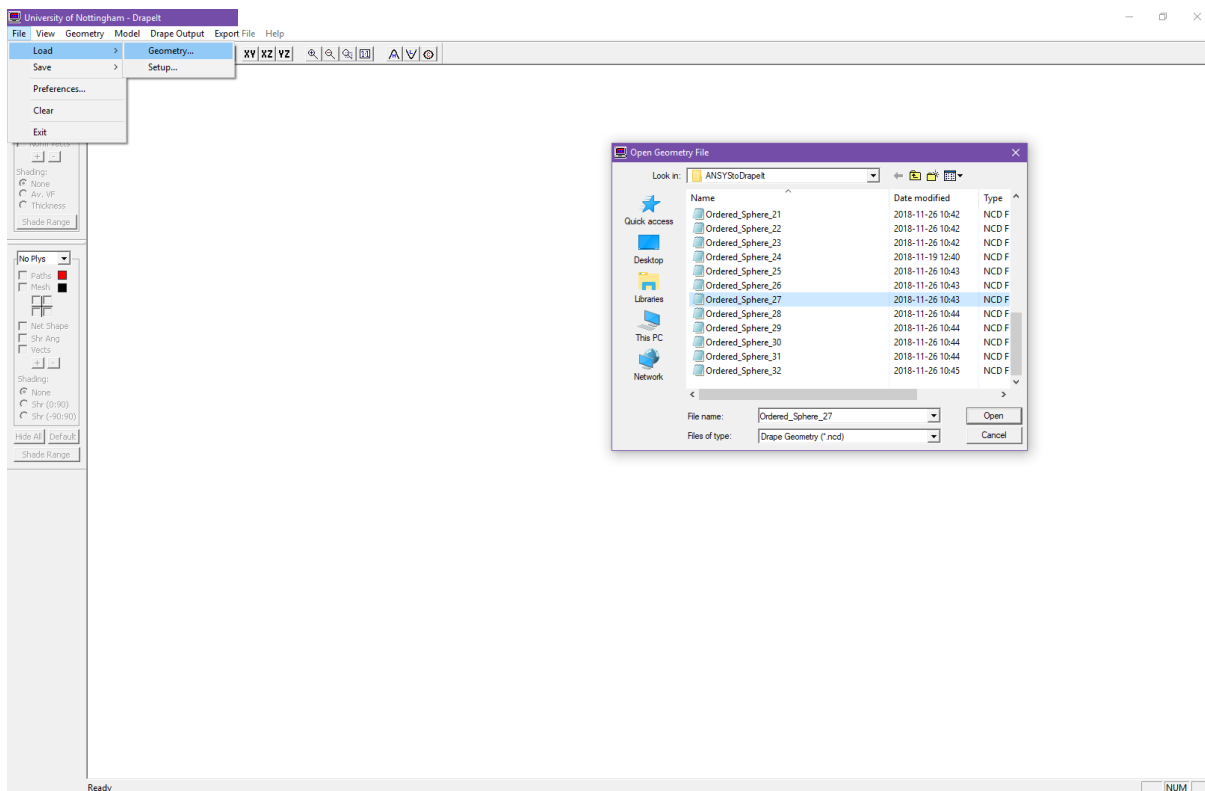


Figure C-1: Loading geometry

2. Select Model → Add Ply → Energy or Kinematic Drape

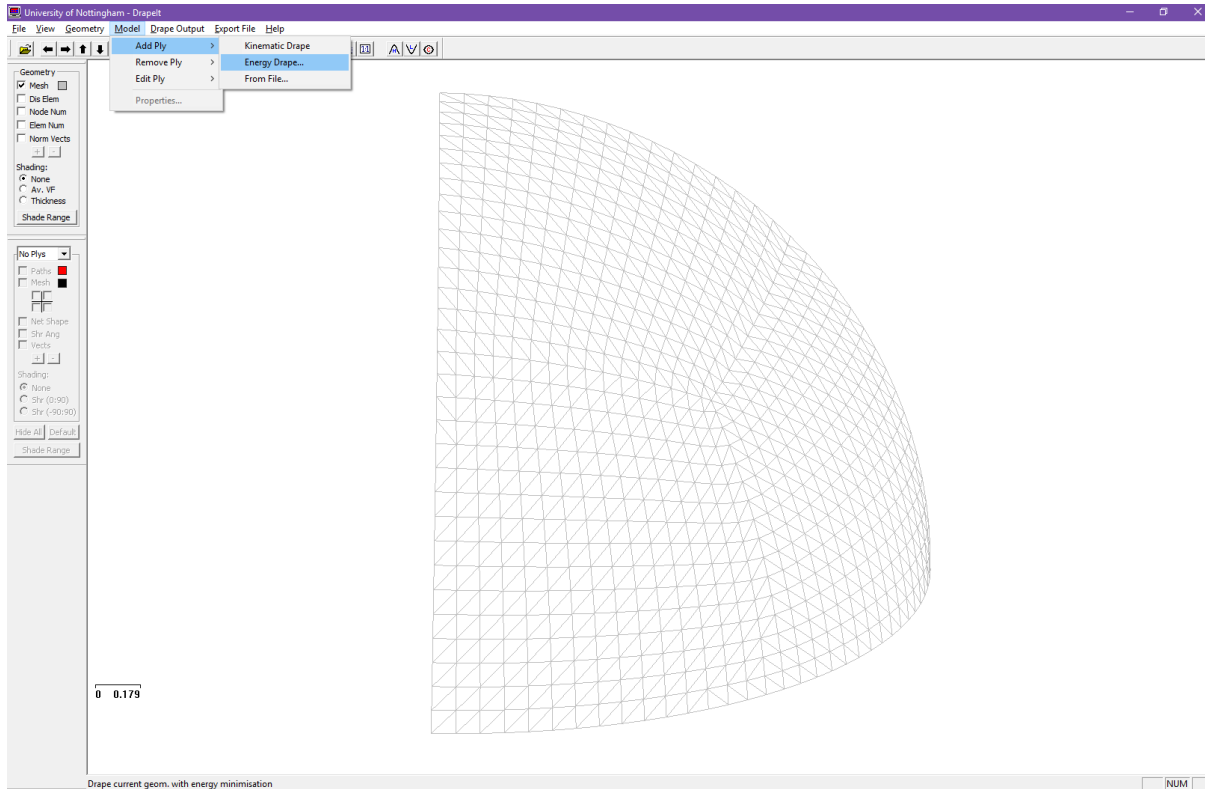


Figure C-2: Adding a new ply

3. The following draping parameters can be adjusted by the user:
 - a. Fibre orientation (degrees)
 - b. Angle between warp and weft (degrees)
 - c. Fibre spacing (mm)
 - d. “Start analysis at” provides the user with options for in determining a starting location for the simulation

During database population – “near node to mouse click” was used most frequently so that the exact starting position could be identified if desired. The analysis method should remain as “vectors”.

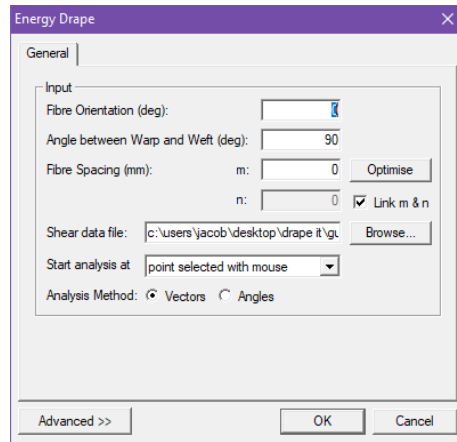


Figure C-3: Adjustable draping parameters

4. Once the draping parameters are set, a starting point is selected on the surface as a start to draping, and the software runs the simulation.

It is important to note the cursor orientation when selecting a draping starting point. This is essential when calculating warp and weft orientation angles (see section C.4) along the surface edges so that they are not mixed in the process. The warp and weft orientations for each plane are as follows (Figure C-4):

- XY Plane
 - Weft is in the positive X direction; Warp is in the positive Y direction
- XZ Plane
 - Weft is in the negative Z direction; Warp is in the positive X direction
- YZ Plane
 - Weft is in the positive Y direction; Warp is in the negative Z direction

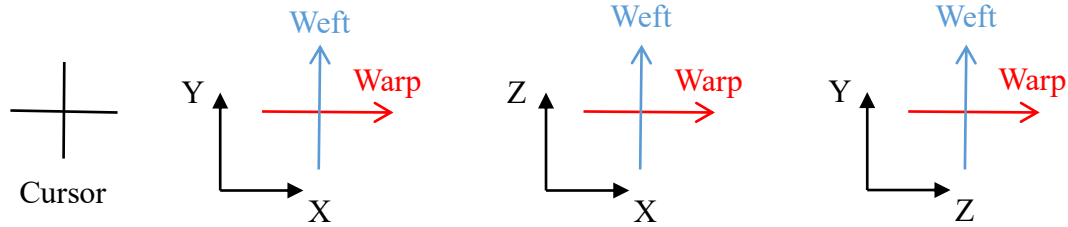


Figure C-4: Warp and weft orientations on each plane

Once complete, visual variations can be made to the output draped ply outlining the geodesic paths, the mesh and a coloured representation of the ply and its areas of shear.

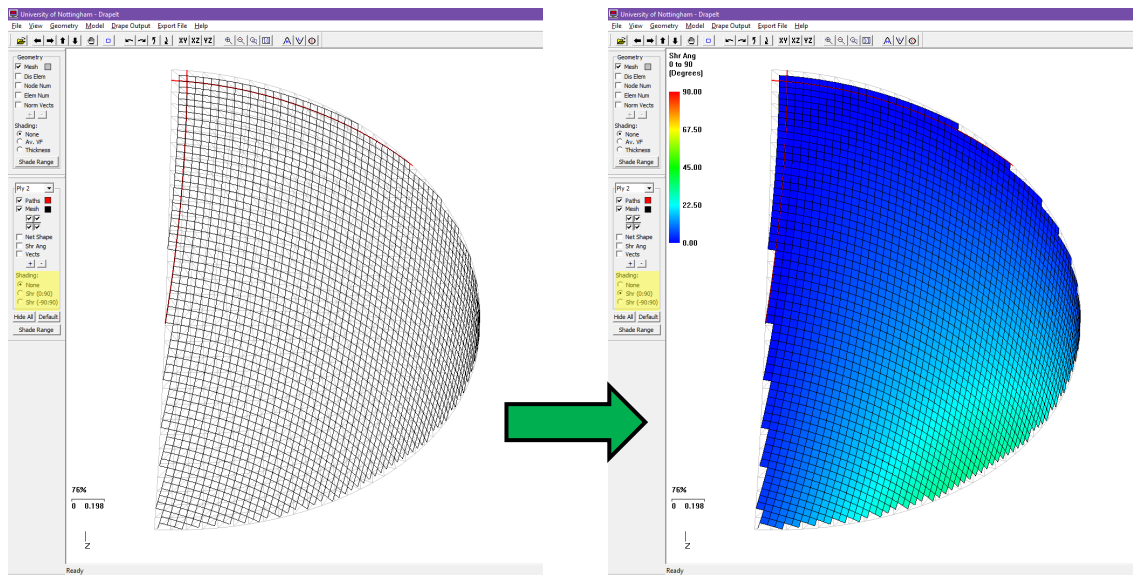


Figure C-5: Visual variations in DrapacIt

With the simulation complete, shear angle as well as warp and weft orientation vectors can be exported for the draped ply. This functionality is found in the export file tab → direction vectors.

The material direction vector (.mdv) file is formatted as follows:

Element Number	Warp Orientation Vector			Weft Orientation Vector			Shear Angle (Degrees)
	X	Y	Z	X	Y	Z	
1	0.463525	-0.850928	-0.247116	0.242684	0.867851	-0.433519	-31.255787
2	0.469746	-0.844516	-0.257163	0.235060	0.876175	-0.420789	-31.420433
3	0.494396	-0.832636	-0.249578	0.230337	0.889931	-0.393660	-31.928595
4	0.501336	-0.825284	-0.259940	0.229333	0.894016	-0.384891	-31.519999
5	0.523388	-0.815521	-0.246961	0.225749	0.904791	-0.361095	-32.042339
6	0.534548	-0.804059	-0.260283	0.222658	0.909438	-0.351206	-31.386456
7	0.557106	-0.793320	-0.245512	0.217237	0.919152	-0.328583	-31.835808
8	0.569416	-0.780412	-0.258308	0.214304	0.922959	-0.319719	-31.042511
9	0.591268	-0.770714	-0.237492	0.213169	0.929762	-0.300168	-31.282171
10	0.598507	-0.760618	-0.251495	0.209825	0.932291	-0.294629	-30.626425
11	0.637916	-0.732109	-0.238911	0.208279	0.938614	-0.274996	-29.248894

Figure C-6: .mdv file layout

The shear angle is exported in degrees and in most cases is the equal to the angle between the warp and weft vector (calculated in section C.4) minus 90° . If the initial angle between warp and weft is greater or less than 90° , further steps outlined in section C.4 are required to calculate the actual shear angle. For database population the only values of interest are the elements along the edge of the surface. If multiple yarns run through a single element, Nottingham uses an average vector to represent both yarns passing through that element.

C.2 Edge vector calculation and specific element data extraction

To extract the warp and weft orientation vectors and the shear angles for the elements along the edges of the simulated surface, the edge elements need to be identified first. Element numbering can be enabled in the geometry tool. The user should identify the frustum conical parameters at this stage so that the edges can be correctly labelled and organized into the database.

To hasten the extraction process, elements that have edges that fall on the geometry edges are noted manually and added to a MATLAB script. This script will use the exported .mdv file from section C.1. When the mesh was built in ANSYS, specific commands were used so that elements would be order in a sequence along the edges, allowing the edge to be spilt into sections where the element numbers increase uniformly. As an example for edge DC:

- Starting from point D, the element number is 544;
- The elements increase by 32 until element 1024 is reached;
- The element following 1024 is 2. Element number increase by 2 until 8 is reached;
- The final section starts at element 9, increasing by 2 until element 31 is reached at point C;

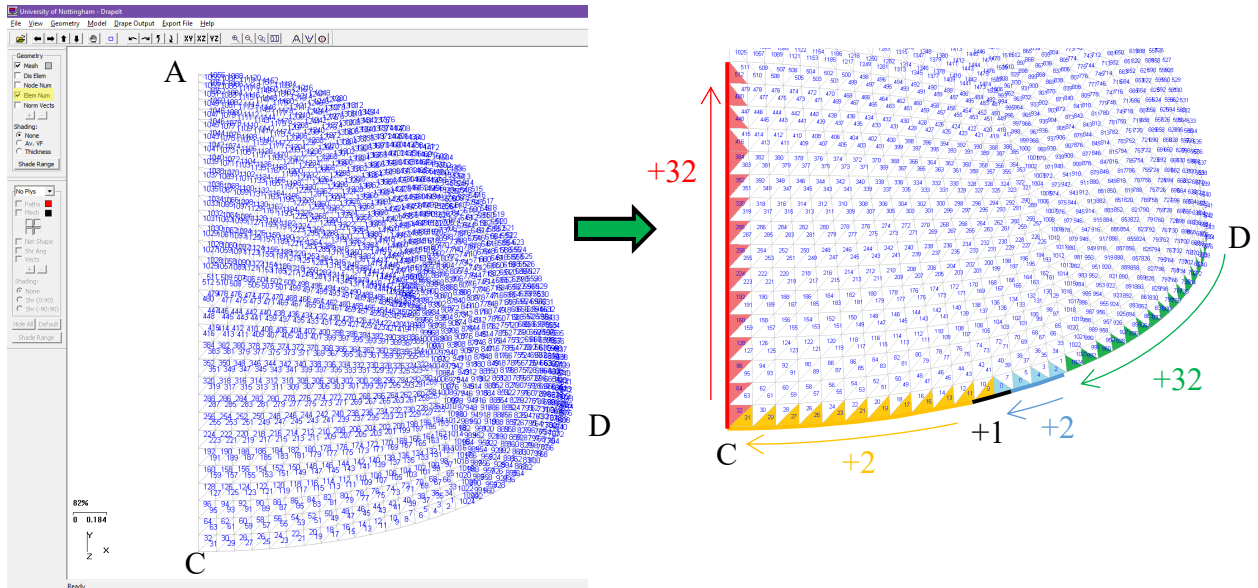


Figure C-7: Edge element identification

This is repeated for each edge of the surface so that all the edge data can be extracted automatically using the MATLAB script. MATLAB exports a new text file that contains solely the elements along the edges of the base surface as identified by the user.

C.3 MATLAB script for edge element extraction

The MATLAB script following the instructions is used for extracting specific edge elements on a base surface. The following MATLAB functions are used in the script [53]:

- *uiimport*('filename') – opens and imports the file designated in the apostrophes.
 - a new window is opened displaying importing options for data in the file
 - “Fixed Width” is always used for importing a table into MATLAB
- *fileID = fopen*('filename', 'permission') – this function is used to open and write a new file where the edge element data will go. Permission *w* allows the file to be written using print commands
- *fprintf*(fileID, 'formatSpec', $A_1, \dots, A_n$) – used to directly write to the identified fileID
 - For text simply write in the desired text in formatSpec with the apostrophes. Note that `\n` is used to drop to the next line.
 - To call numbers directly from the imported table, '%s ' is used followed by the designation of the table row desired to import ($A_1, \dots, A_n$). To call a specific row and column in the table simply write the table name, followed by {column, row}.
- *for* index = values, statements, end – a loop used to repeat a specified number of times
 - used for listing individual sections along the base surface edge and printing only the desired element information to a new textfile

The following instructions are used for the MATLAB script to extract specific element data:

1. To start, the *uiimport* function is run independently from the rest of the script allowing the .mdv file to be imported into MATLAB. A new window will open displaying importing options. The “Fixed Width” option is to be selected, then Import Selection. A table name

can be set directly below column A, which will be used in later commands (MLE900, Figure C-8) Once complete a table will be imported into MATLAB containing all the data from the .mdv.

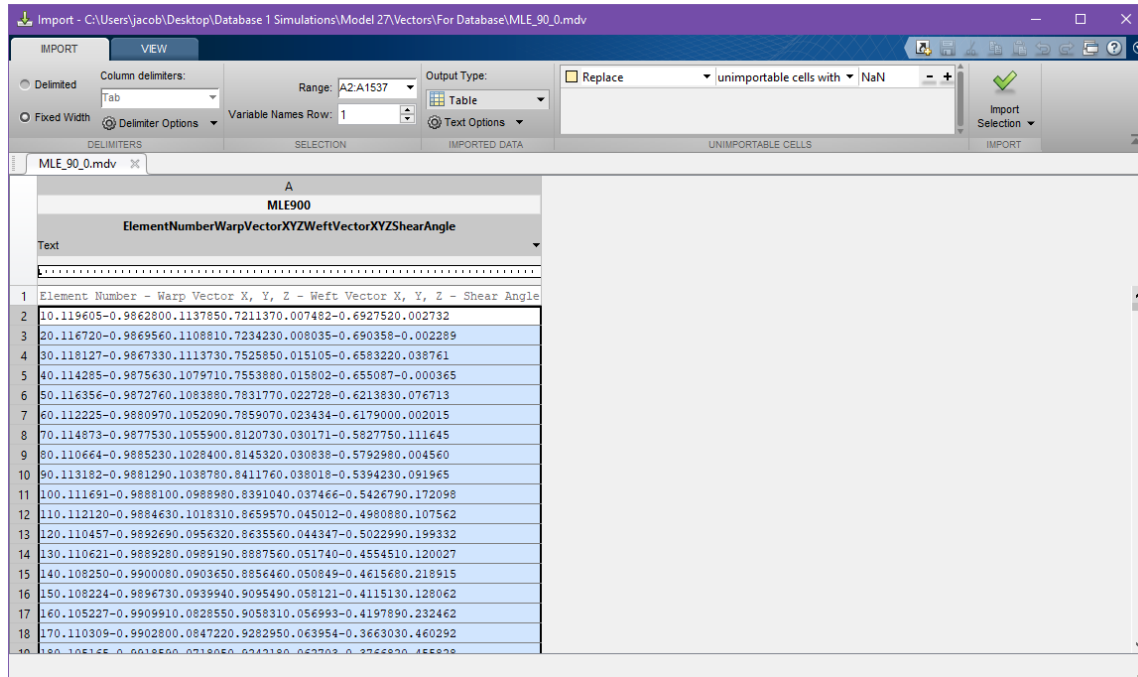


Figure C-8: uiimport command window

2. *uiimport* is then commented so that the file does not get imported again, replacing the old file and causing the script to stop running and the rest of the script is uncommented. The new extraction file is named in the *fopen* function and labeled fileID1.
3. A group of for loops are sectioned from one another, each containing a title for a specific edge. The values for z represent the starting element, increase interval, and end element (z = 1:2:11), calling rows directly from the imported table and printing them into the new textfile.
4. Once all the sections are correctly identified in the loops the script is run and a text file containing the designated edge elements and their respective data is exported.


```

1
2 -   uiimport('MLE_90_0.mdv');
3
4 -   fileID1 = fopen('EdgeExtractionExample2.txt','w');
5
6 -   fprintf(fileID1, 'DC Edges\n');
7 -   for z = 544:32:1024
8 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
9 -   end
10 -   for z = 2:2:8
11 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
12 -   end
13 -   for z = 9:2:31
14 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
15 -   end
16
17 -   fprintf(fileID1, 'CA Edges\n');
18 -   for z = 32:32:512
19 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
20 -   end
21 -   for z = 1025:2:1055
22 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
23 -   end
24
25 -   fprintf(fileID1, 'AD Edges\n');
26 -   for z = 1056:32:1536
27 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
28 -   end
29
30 -   for z = 513:2:543
31 -       fprintf(fileID1, ' %s\n',MLE900{z,1});
32 -   end

```

from Section C.3: manual element identification for edge DC

Figure C-9: MATLAB script for extracting edge element data

DC Edges

544	0.039276	-0.992012	0.119874	0.028583	-0.097784	-0.994797	-1.210444
576	0.050827	-0.992077	0.114889	0.078887	-0.095160	-0.992331	-0.893433
608	0.061886	-0.991526	0.114216	0.128833	-0.092013	-0.987388	-0.777499
640	0.073990	-0.990936	0.112122	0.178368	-0.088249	-0.979998	-0.528992
672	0.086256	-0.989590	0.115204	0.227372	-0.083925	-0.970185	-0.521699
704	0.098572	-0.987996	0.118947	0.275584	-0.078537	-0.958063	-0.527128
736	0.109470	-0.986350	0.123001	0.322762	-0.072287	-0.943716	-0.541152
768	0.120247	-0.985382	0.120677	0.368725	-0.065263	-0.927245	-0.186164
800	0.126319	-0.984402	0.122458	0.413378	-0.057573	-0.908738	-0.136943
832	0.128843	-0.983930	0.123617	0.456716	-0.049527	-0.888233	-0.127487
864	0.128746	-0.983972	0.123384	0.498769	-0.041242	-0.865753	-0.116025
896	0.126758	-0.984418	0.121872	0.539561	-0.032874	-0.841305	-0.101757
928	0.124358	-0.984974	0.119838	0.579105	-0.024534	-0.814883	-0.084321

Figure C-10: Specific edge element data extraction

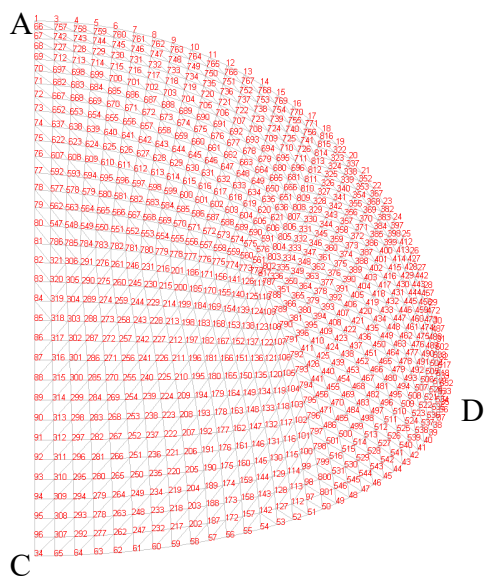
C.4 Warp and weft orientation angle calculations

The last stage of data extraction for database population is the identification and calculation of edge vectors, and warp and weft orientation angles. The spreadsheet used to make the original .ncd will be used to compile and calculate this information.

The nodes along each of the edges can be identified in a similar manner to the elements - node numbering can be enabled in the geometry tool. In a new spreadsheet, frustum and draping parameters are identified in columns A and B followed by node numbers and their respective coordinates, which are copied from the original spreadsheet. It is important that the edges are identified and organized at this stage. Once the edge nodes have all been moved into their own sheet, vectors between each pair of nodes are calculated using the Equation below, moving clockwise around the base surface. These vectors align perfectly with each of the elements along each of the edges. They are used for calculating the warp and weft orientation angles after a simulation is run.

$$\vec{V}_1 = \langle (X_2 - X_1), (Y_2 - Y_1), (Z_2 - Z_1) \rangle \quad (26)$$

Where $\vec{V}_1$ is the vector between nodes, (X_1, Y_1, Z_1) are the coordinates of first node and (X_2, Y_2, Z_2) are the coordinates of the adjacent node, moving around the surface in the following the conical frustum semi-alphabetical point labelling: A, B, D, C for four points, and either A, B, D/C or A/B, D, C for three points.



1	A	B	C	D	E	F	G	H	I	J	K	L
2	CF Parameters	Base St Edge			Nodes	X	Y	Z	Vectors	X	Y	Z
3	R1	0	DC		34	-1.144	0	0.586		0.0981	0	-0.002
4	R2	2			65	-1.316	0	0.5836		0.0979	0	-0.007
5	L	2			64	-1.218	0	0.5764		0.0974	0	-0.012
6	β	90			62	-1.121	0	0.5644		0.0967	0	-0.017
7	CR	2			63	-1.024	0	0.5476		0.0958	0	-0.022
8	Shape Parameters				61	-0.928	0	0.5311		0.0946	0	-0.027
9	Start	C			60	-0.833	0	0.4959		0.0932	0	-0.031
10	Shear				59	-0.74	0	0.4691		0.0916	0	-0.035
11	Orientatic	45			58	-0.649	0	0.4338		0.0897	0	-0.04
12					57	-0.559	0	0.394		0.0877	0	-0.044
13					56	-0.471	0	0.3498		0.0854	0	-0.048
14					55	-0.386	0	0.3015		0.0829	0	-0.053
15					54	-0.303	0	0.2489		0.0803	0	-0.057
16					53	-0.223	0	0.1974		0.0774	0	-0.061
17					52	-0.145	0	0.132		0.0743	0	-0.064
18					51	*****	0	*****		0.0711	0	-0.068
19					50	*****	0	*****		0.0677	0	-0.071
20					49	*****	0	*****		0.0641	0	-0.074
21					48	0.132	0	-0.145		0.0604	0	-0.077
22					47	0.1924	0	-0.223		0.0565	0	-0.08
23					46	0.2489	0	-0.303		0.0525	0	-0.083
24					45	0.3015	0	-0.386		0.0484	0	-0.086
25					44	0.3498	0	-0.471		0.0441	0	-0.088
26					43	0.394	0	-0.559		0.0398	0	-0.09
27					42	0.4338	0	-0.649		0.0353	0	-0.092
28					41	0.4691	0	-0.74		0.0308	0	-0.093
29					40	0.4999	0	-0.833		0.0262	0	-0.095
30					39	0.5261	0	-0.926		0.0215	0	-0.096
31					38	0.5476	0	-1.024		0.0168	0	-0.097
32					37	0.5644	0	-1.121		0.012	0	-0.098
33					36	0.5764	0	-1.218		0.0072	0	-0.098
34					35	0.5836	0	-1.316		0.0024	0	-0.098
35					2	0.586	0	-1.414				

Figure C-11: Node display in Drapelt and edge vector calculations

With all of the edge vectors calculated, the MATLAB generated edge element sheet can be copied alongside the vectors, ensuring that the elements correspond to the correct vector. A method for checking that the correct vectors are alongside their respective elements consists in toggling both element and node numbering in Drape It.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T		
1	CF Parameters		Base St	Edge	Nodes	X	Y	Z	Vectors	X	Y	Z	Elements	X	Y	Z	Warp	X	Y	Z	Shear
2	R1	0		DC	34	-1.414	0	0.586	0.0381	0	-0.002	0	-0.002	31	0.9895	0.1416	-0.029	0.1574	0.9869	-0.035	17.252
3	R2	2			65	-1.316	0	0.5836	0.0379	0	-0.007	29	0.9304	0.1145	-0.078	0.164	0.9851	-0.052	16.214		
4	L	2			64	-1.218	0	0.5764	0.0374	0	-0.012	27	0.9883	0.0853	-0.126	0.1621	0.9843	-0.07	14.657		
5	β	90			63	-1.121	0	0.5644	0.0367	0	-0.017	25	0.9833	0.0531	-0.174	0.1591	0.9832	-0.069	12.953		
6	CR	2			62	-1.024	0	0.5476	0.0358	0	-0.022	23	0.9751	0.0217	-0.221	0.1545	0.9818	-0.111	11.326		
7	Drape Parameters			61	-0.928	0	0.5261	0.0346	0	-0.026	21	0.964	-0.009	-0.266	0.1484	0.9799	-0.134	9.7606			
8	Start	C		60	-0.833	0	0.4999	0.0332	0	-0.031	19	0.9501	-0.039	-0.31	0.1412	0.9776	-0.156	8.3218			
9	Shear	0		59	-0.74	0	0.4691	0.0316	0	-0.035	17	0.9337	-0.068	-0.352	0.1333	0.9752	-0.177	6.9349			
10	Orientat	45		58	-0.649	0	0.4338	0.0297	0	-0.04	15	0.915	-0.097	-0.392	0.1254	0.9727	-0.195	5.5367			
11				57	-0.559	0	0.394	0.0277	0	-0.044	13	0.8941	-0.122	-0.431	0.1178	0.9704	-0.211	4.4918			
12				56	-0.471	0	0.3498	0.0254	0	-0.048	11	0.8704	-0.156	-0.467	0.1104	0.9682	-0.225	2.8755			
13				55	-0.386	0	0.3015	0.0229	0	-0.053	9	0.8448	-0.184	-0.502	0.1028	0.9661	-0.237	1.5782			
14				54	-0.303	0	0.2489	0.0203	0	-0.057	8	0.8177	-0.203	-0.539	0.0973	0.9647	-0.245	0.9121			
15				53	-0.223	0	0.1924	0.0174	0	-0.06	6	0.7875	-0.24	-0.568	0.0886	0.9624	-0.257	-0.87			

Figure C-12: Imported elemental data from C.3 MATLAB export

Finally, the warp and weft orientation angles are calculated using Equation 27 for each element, in radians. Once all the orientation angles are calculated, a maximum, minimum and an average

are calculated for each edge. The same is done for shear angles, and all this information is added to the database.

$$\cos\theta_{warp} = \left(\frac{\overrightarrow{V_{warp}} \cdot \overrightarrow{V_{element}}}{|\overrightarrow{V_{warp}}| |\overrightarrow{V_{element}}|} \right) \quad (27)$$

where θ_{warp} is the warp orientation angle, $\overrightarrow{V_{element}}$ is the vector along the element edge calculated from the coordinates of two adjacent nodes and $\overrightarrow{V_{warp}}$ is the warp orientation the same element. This calculation is repeated, substituting $\overrightarrow{V_{weft}}$ for $\overrightarrow{V_{warp}}$ and θ_{weft} for θ_{warp} .

T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH
		Warp		Weft				Warp		Weft		Shear		
Shear		Radians	Deg		Radians	Deg			Rad	Deg	Rad	Deg	Rad	Deg
17.252		0.1422	8.1471		1.4119	80.895								
16.214		0.1148	6.5792		1.4026	80.365		Max	0.5361	34.156	1.4119	80.895	0.3011	17.252
14.657		0.0855	4.9		1.4005	80.243		Min	0.0093	0.5315	1.0838	62.096	-0.137	-7.875
12.953		0.0532	3.0481		1.3979	80.095		Average	0.3062	17.546	1.2899	73.906	0.0008	0.0432
11.326		0.0217	1.245		1.3948	79.919								
9.7606		0.0093	0.5315		1.3912	79.708								
8.3218		0.0388	2.221		1.3868	79.457								
6.9349		0.0681	3.9016		1.3817	79.164								
5.5367		0.0984	5.6356		1.3758	78.828								
4.4918		0.123	7.0501		1.3694	78.458								
2.8755		0.1581	9.0572		1.3625	78.067								
1.5782		0.1876	10.751		1.3556	77.67								

Figure C-13: Final values from the calculations for the database

In the event that the angle between the warp and weft is set to be greater than or less than 90° during simulation, the shear angle needs to be recalculated. The exported .mdv files expresses shear angle relative to the starting angle between the warp and weft and does not consider the initial assigned change. The actual shear angle can be calculated using the warp and weft vectors and subtract the result using Equation 27, replacing θ_{warp} with θ_{shear} , $\overrightarrow{V_{element}}$ with $\overrightarrow{V_{weft}}$ and $|\overrightarrow{V_{element}}|$ with $|\overrightarrow{V_{weft}}|$, from 90° .

Appendix D

***BD100* Mould Points**

Base Surface		Vertex	Coordinates			Conical Frustum Vertex Label
			<i>x</i>	<i>y</i>	<i>z</i>	
1-1	Flat	1	0	0	0	A
		2	0	0	6	B
		14	23	0	6	D
		42	23	0	0	C
1-2	Flat	42	23	0	0	A
		14	23	0	6	B
		20	24.42	0	7.42	D
		43	24.42	0	0	C
1-3	Flat	43	24.42	0	0	A
		20	24.42	0	7.42	B
		44	30.42	0	7.42	D
		37	30.42	0	0	C
1-4	Flat	44	30.42	0	7.42	A
		20	24.42	0	7.42	B
		36	24.42	0	18.3175	D
		38	30.42	0	18.3175	C
2	Single Curvature	2	0	0	6	A
		3	0	0.3175	6.3175	B
		15	23	0.3175	6.3175	D
		14	23	0	6	C
3	Flat	3	0	0.3175	6.3175	A
		4	0	1.2075	6.7575	B
		16	23	1.2075	6.7575	D
		15	23	0.3175	6.3175	C
4	Single Curvature	4	0	1.2075	6.7575	A
		5	0	2.3175	7.8675	B
		17	23	2.3175	7.8675	D
		16	23	1.2075	6.7575	C
5	Flat	5	0	2.3175	7.8675	A
		6	0	2.4675	9.6075	B
		18	23	2.4675	9.6075	D
		17	23	2.3175	7.8675	C
6	Single Curvature	6	0	2.4675	9.6075	A
		7	0	3.4775	10.7175	B
		19	23	3.4775	10.7175	D
		18	23	2.4675	9.6075	C

7-1	Flat	7	0	3.4775	10.7175	D
		39	0	10.6275	10.7175	B
		25	28	10.6275	10.7175	A
		19	23	3.4775	10.7175	C
7-2	Flat	39	0	10.6275	10.7175	D
		40	0	15.8275	10.7175	B
		27	30	15.8275	10.7175	A
		25	28	10.6275	10.7175	C
7-3	Flat	40	0	15.8275	10.7175	D
		41	0	27.8275	10.7175	B
		29	30	27.8275	10.7175	A
		27	30	15.8275	10.7175	C
7-4	Flat	41	0	27.8275	10.7175	D
		8	0	36.4775	10.7175	B
		11	21.5	36.4775	10.7175	A
		29	30	27.8275	10.7175	C
8	Single Curvature	8	0	36.4775	10.7175	A
		9	0	37.8275	12.0675	B
		12	21.5	37.8275	12.0675	D
		11	21.5	36.4775	10.7175	C
9	Flat	9	0	37.8275	12.0675	A
		12	0	37.8275	18.3175	B
		13	21.5	37.8275	18.3175	D
		10	21.5	37.8275	12.0675	C
10	Double Curvature	21	24.1025	0.3175	7.42	A
		15	23	0.3175	6.3175	B
		14	23	0	6	D
		20	24.42	0	7.42	C
11	Single Curvature	22	24.1025	1.3175	7.42	A
		16	23	1.2075	6.7575	B
		15	23	0.3175	6.3175	D
		21	24.1025	0.3175	7.42	C
12	Double Curvature	17	23	2.3175	7.8675	A
		16	23	1.2075	6.7575	D
		22	24.1025	1.3175	7.42	C
13	Single Curvature	23	24.1025	1.4475	8.91	A
		18	23	2.4675	9.6075	B
		17	23	2.3175	7.8675	D
		22	24.1025	1.3175	7.42	C

14	Double Curvature	19	23	3.4775	10.7175	A
		18	23	2.4675	9.6075	B
		23	24.1025	1.4475	8.91	D
		24	24.2025	2.2475	12.0675	C
15	Single Curvature	26	29.6	10.3275	12.0675	A
		25	28	10.6275	10.7175	B
		19	23	3.4775	10.7175	D
		24	24.2025	2.2475	12.0675	C
16	Double Curvature	27	30	15.8275	10.7175	A
		25	28	10.6275	10.7175	B
		26	29.6	10.3275	12.0675	D
		28	31.35	15.8275	12.0675	C
17	Single Curvature	30	31.35	27.8275	12.0675	A
		29	30	27.8275	10.7175	B
		27	30	15.8275	10.7175	D
		28	31.35	15.8275	12.0675	C
18	Double Curvature	11	21.5	36.4775	10.7175	A
		29	30	27.8275	10.7175	B
		30	31.35	27.8275	12.0675	D
		12	21.5	37.8275	12.0675	C
19	Single Curvature	12	21.5	37.8275	12.0675	A
		30	31.35	27.8275	12.0675	B
		31	31.35	27.8275	18.3175	D
		13	21.5	37.8275	18.3175	C
20	Flat	30	31.35	27.8275	12.0675	A
		28	31.35	15.8275	12.0675	B
		32	31.35	15.8275	18.3175	D
		31	31.35	27.8275	18.3175	C
21	Single Curvature	28	31.35	15.8275	12.0675	A
		26	29.6	10.3275	12.0675	B
		33	29.6	10.3275	18.3175	D
		32	31.35	15.8275	18.3175	C
22	Flat	26	29.6	10.3275	12.0675	A
		24	24.2025	2.2475	12.0675	B
		34	24.2025	2.2475	18.3175	D
		33	29.6	10.3275	18.3175	C
23-1	Flat	24	24.2025	2.2475	12.0675	A
		23	24.1025	1.4475	8.91	B
		46	24.1025	1.4475	18.3175	D
		34	24.2025	2.2475	18.3175	C

23-2	Flat	23	24.1025	1.4475	8.91	A
		22	24.1025	1.3175	7.42	B
		45	24.1025	1.3175	18.3175	D
		46	24.1025	1.4475	18.3175	C
23-3	Flat	22	24.1025	1.3175	7.42	A
		21	24.1025	0.3175	7.42	B
		35	24.1025	0.3175	18.3175	D
		45	24.1025	1.3175	18.3175	C
24	Single Curvature	21	24.1025	0.3175	7.42	A
		20	24.42	0	7.42	B
		36	24.42	0	18.3175	D
		35	24.1025	0.3175	18.3175	C

Appendix E

MATLAB Scripts

E.1 MATLAB Script 1

Values used from Example 1 in Section 3.2.5. This script calculates all conical frustum parameters based on extracting corner points, midpoints and edge lengths for each edge of a base surface.

```
clc;
clear;

syms X1 Y1 Z1 X2 Y2 Z2

%Inputs points
A = [2.823; 12.608; 3.019]; %Point A
M_AB = [4.468; 11.4; -0.987]; %Midpoint AB
B = [3; 10; -5]; %Point B
C = [8.192; 0; 5.736]; %Point C
M_CD = [9.99; 0; 0.436]; %Midpoint CD
D = [8.66; 0; -5]; %Point D

M_AC = [7.096; 7.253; 5.643];
M_BD = [6.83; 5.566; -5.908];

C1 = [X1; Y1; Z1]
C2 = [X2; Y2; Z2]

%ArcLengths
Sab = 9.184 ;
Sbd = 11.982 ;
Scd = 11.345 ;
Sac = 14.911 ;

%New Coordaintes System where D is 0

A = A - D;
M_AB = M_AB - D;
B = B - D;
C = C - D;
M_CD = M_CD - D;
M_AC = M_AC - D;
M_BD = M_BD - D;

D = D - D;

%Radius of Arc on 1

%Vector between A and B for the length
AB = B-A;
magAB = sqrt(AB(1)^2 + AB(2)^2 + AB(3)^2)

%Midpoint between A and B
Wab = (A+B)/2
```

```

%Vector between B and Mab for legnth
MB = M_AB - B;
magMB = sqrt(MB(1)^2 + MB(2)^2 + MB(3)^2)

%Height at Wab to Mab
MWab = sqrt(magMB^2 - ((magAB/2))^2)

%R1 = r1 = radius of smooth arc
magR1 = MWab/2 + (magAB^2)/(8*MWab)

%%% %% For Ellipse 2 %%% %%

%Radius of Arc on 2

%Vector between C and D for the length
CD = D-C;
magCD = sqrt(CD(1)^2 + CD(2)^2 + CD(3)^2)

%Midpoint between C and D
Wcd = (C+D)/2

%Vector between D and Mcd for legnth
MD = M_CD - D;
magMD = sqrt(MD(1)^2 + MD(2)^2 + MD(3)^2)

%Height at Wcd to Mcd
MWcd = sqrt(magMD^2 - ((1/2)*magCD)^2)

%R2 = r2 = radius of smooth arc
magR2 = MWcd/2 + (magCD)^2/(8*MWcd)

%%%%%%%% %%%%%%%%% %%%%%%%%% %%%%%%%%%

syms X1 Y1 Z1 X2 Y2 Z2

eq1 = (A(1) - C1(1))^2 + (A(2) - C1(2))^2 + (A(3) - C1(3))^2 - magR1^2
eq2 = (B(1) - C1(1))^2 + (B(2) - C1(2))^2 + (B(3) - C1(3))^2 - magR1^2
eq3 = (M_AB(1) - C1(1))^2 + (M_AB(2) - C1(2))^2 + (M_AB(3) - C1(3))^2 - magR1^2

sol1 = solve(eq1,eq2,eq3);

eq4 = (C(1) - C2(1))^2 + (C(2) - C2(2))^2 + (C(3) - C2(3))^2 - magR2^2
eq5 = (D(1) - C2(1))^2 + (D(2) - C2(2))^2 + (D(3) - C2(3))^2 - magR2^2
eq6 = (M_CD(1) - C2(1))^2 + (M_CD(2) - C2(2))^2 + (M_CD(3) - C2(3))^2 - magR2^2

```

```

sol2 = solve(eq4,eq5,eq6);

x1 = round( sol1.X1 )
y1 = round ( sol1.Y1 )
z1 = round ( sol1.Z1)

C1 = [x1(1);y1(1);z1(1)]

x2 = round (sol2.X2)
y2 = round (sol2.Y2)
z2 = round (sol2.Z2)

C2 = [x2(1);y2(1);z2(1)]

Beta1 = 360*Sab/(2*pi*magR1)
Beta2 = 360*Scd/(2*pi*magR2)

R1 = A - C1;
r1 = B - C1;
R2 = C - C2;
r2 = D - C2;

N1 = cross(R1,r1);
magN1 = sqrt(N1(1)^2 + N1(2)^2 + N1(3)^2);
N2 = cross(r2,R2);
magN2 = sqrt(N2(1)^2 + N2(2)^2 + N2(3)^2);

L = C2 - C1;
magL = sqrt(L(1)^2 + L(2)^2 + L(3)^2)

Theta1 = round((radtodeg(acos(((dot(N1,L))/(magN1*magL))))))
Theta2 = round((radtodeg(acos(((dot(L,N2))/(magN2*magL))))))

R1L = cross(R1,L);
magR1L = sqrt(R1L(1)^2 + R1L(2)^2 + R1L(3)^2);
R2L = cross(R2,L);
magR2L = sqrt(R2L(1)^2 + R2L(2)^2 + R2L(3)^2);

alpha = round((radtodeg(acos((dot(R1L,R2L))/(magR1L*magR2L))))))

%% Calculate CR and Cr - same technique as calculate R1 and R2

%Radius CR

%Vector between A and B for the length
AC = C-A;
magAC = sqrt(AC(1)^2 + AC(2)^2 + AC(3)^2)

%Midpoint between A and B
Wac = (A+C)/2;

%Vector between B and Mab for legnth
MA = M - AC - A;
magMA = sqrt(MA(1)^2 + MA(2)^2 + MA(3)^2);

```

```

%Height at Wab to Mab
MWac = sqrt(magMA^2 - ((magAC/2))^2);

%R1 = r1 = radius of smooth arc
CR = MWac/2 + (magAC^2)/(8*MWac)

%Radius of Cr
%Vector between B and D for the length
BD = D-B;
magBD = sqrt(BD(1)^2 + BD(2)^2 + BD(3)^2);

%Midpoint between B and D
Wbd = (B+D)/2;

%Vector between D and Mcd for legnth
MB = M_BD - B;
magMB = sqrt(MB(1)^2 + MB(2)^2 + MB(3)^2);

%Height at Wcd to Mcd
MWbd = sqrt(magMB^2 - ((1/2)*magBD)^2);

%R2 = r2 = radius of smooth arc
Cr = MWbd/2 + (magBD)^2/(8*MWbd)

%%%% %%%% %%%% %%%%

```

E.2 MATLAB Script 2

Values used from Example 2 in Section 3.2.6. This script calculates all conical frustum parameters based on extracting corner points, midpoints and radii for each edge of a base surface.

```
clear;
clc;

syms X1 Y1 Z1 X2 Y2 Z2

%Inputs points
A = [-1;11;3]; %Point A
M_AB = [0;10.691;1.659]; %Midpoint AB
B = [0;10;0]; %Point B
C = [2;0;0]; %Point C
M_CD = [1.069;0;-0.423]; %Midpoint CD
D = [0;0;-0.5]; %Point D

%New Coordaintes System where D is 0
A = A - D;
M_AB = M_AB - D;
B = B - D;
C = C - D;
M_CD = M_CD - D;
D = D - D;

%Set Centre point variables
C1 = [X1;Y1;Z1];
C2 = [X2;Y2;Z2];

%Known Radii at A, B, C and D
magR1 = sqrt(6);
magr1 = sqrt(5);
magR2 = 2;
magr2 = 0.5;

%Setting Equations for Vectors
R1 = A - C1;
r1 = B - C1;
vM = M_AB - C1;

R2 = C - C2;
r2 = D - C2;

%Begin Method 2 Calculation, Vectors between A and Midpoint AC
AM = M_AB - A
AB = B - A
% and A and B ^

% Vector between C and Midpoint CD
CM = M_CD - C
```

```

CD = D - C
% and C and D ^

%Normal Vector for the Plane
N1 = cross (AM,AB)
magN1 = sqrt(N1(1)^2 + N1(2)^2 + N1(3)^2)
N2 = cross (CD,CM)
magN2 = sqrt(N2(1)^2 + N2(2)^2 + N2(3)^2)

%Begin Solver, Plane Equation, Vector Magnitiude Equations
eq1 = N1(1)*(X1-A(1)) + N1(2)*(Y1-A(2)) + N1(3)*(Z1-A(3));
eq2 = R1(1)^2 + R1(2)^2 + R1(3)^2 - magR1^2;
eq3 = r1(1)^2 + r1(2)^2 + r1(3)^2 - magr1^2;

eq4 = N2(1)*(X2-C(1)) + N2(2)*(Y2-C(2)) + N2(3)*(Z2-C(3));
eq5 = R2(1)^2 + R2(2)^2 + R2(3)^2 - magR2^2;
eq6 = r2(1)^2 + r2(2)^2 + r2(3)^2 - magr2^2;

%MATLAB Solve Function
sol1 = solve(eq1,eq2,eq3);
x1 = (sol1.X1)
y1 = (sol1.Y1)
z1 = (sol1.Z1)
%Yields 2 solutions!

sol2 = solve(eq4,eq5,eq6);
x2 = sol2.X2
y2 = sol2.Y2
z2 = sol2.Z2

%Check Location of Mab, Determine which solution is correct

magMAB1 = ( sqrt((M_AB(1)- x1(1))^2 + (M_AB(2)- y1(1))^2 + (M_AB(3)-z1(1))^2)
)
magMAB2 = ( sqrt((M_AB(1)- x1(2))^2 + (M_AB(2)- y1(2))^2 + (M_AB(3)-z1(2))^2)
)

magMCD1 = ( sqrt((M_CD(1)- x2(1))^2 + (M_CD(2)- y2(1))^2 + (M_CD(3)-z2(1))^2)
)
magMCD2 = ( sqrt((M_CD(1)- x2(2))^2 + (M_CD(2)- y2(2))^2 + (M_CD(3)-z2(2))^2)
)

%Check which distance is lesser
if gt(magMAB1,magMAB2)
    C1 = [x1(1);y1(1);z1(1)];
end
if gt(magMAB2,magMAB1)
    C1 = [x1(2);y1(2);z1(2)];
end
if gt(magMCD1,magMCD2)
    C2 = [x2(1);y2(1);z2(1)];
end
if gt(magMCD2,magMCD1)
    C2 = [x2(2);y2(2);z2(2)];
end

```



```

%Centre Points
C1
C2

%Caclulates Vectors R1, r1, R2, r2
R1 = A - C1;
r1 = B - C1;
R2 = C - C2;
r2 = D - C2;

%Vector Dot product to calculate Sectioning Angles Beta 1 and Beta 1
Beta1 = round((radtodeg(acos(((dot(R1,r1))/(magR1*magr1))))))
Beta1 = round((radtodeg(acos(((dot(R2,r2))/(magR2*magr2))))))

%Calculate L
L = C2 - C1;
magL = sqrt(L(1)^2 + L(2)^2 + L(3)^2)
Theta1 = round((radtodeg(acos(((dot(N1,L))/(magN1*magL))))))
Theta2 = round((radtodeg(acos(((dot(L,N2))/(magN2*magL))))))

R1L = cross(R1,L);
magR1L = sqrt(R1L(1)^2 + R1L(2)^2 + R1L(3)^2);
R2L = cross(R2,L);
magR2L = sqrt(R2L(1)^2 + R2L(2)^2 + R2L(3)^2);

alpha = round((radtodeg(acos((dot(R1L,R2L))/(magR1L*magR2L))))))

```

Appendix F

SolidWorks Macro

F.1 SolidWork macro: Point, length and radii extraction

The following macro was developed by Fabien Salles at Hutchinson Aéronautique et Industrie. It is used to extract end points, midpoints, arc lengths, and radii from mould surfaces in SolidWorks.

The macro produces a spreadsheet listing the individual point coordinates along their respective edges alongside the length of the edges the points lie on.

```
Option Explicit
Public Const LINE_TYPE As Integer = 3001
Public Const CIRCLE_TYPE As Integer = 3002
Public Const ELLIPSE_TYPE As Integer = 3003
Public Const INTERSECTION_TYPE As Integer = 3004
Public Const BCURVE_TYPE As Integer = 3005
Public Const SPCURVE_TYPE As Integer = 3006
Public Const CONSTPARAM_TYPE As Integer = 3008
Public Const TRIMMED_TYPE As Integer = 3009

'''''' Dimension to delete reference points

Public ref_type As String

'''''' Dimensions for data collection arrays

Public exApp As Excel.Application
Public sheet As Excel.Worksheet

Public tab_data(8000, 7) As Variant ' Array to write in Excel

Public e As Integer ' Edge number
Public f As Integer ' Face number
Public p As Integer ' Point number

Sub main()

Dim swApp As SldWorks.SldWorks
Dim swModel As SldWorks.ModelDoc2
Dim swSelMgr As SldWorks.SelectionMgr
Dim swSelData As SldWorks.SelectData
Dim swPart As SldWorks.PartDoc
Dim swBody As SldWorks.Body2
Dim swFace As SldWorks.Face
Dim swEnt As SldWorks.Entity
Dim bRet As Boolean
Dim vBodies As Variant

Dim modview As ModelView

Dim starttime As Variant
Dim Secondselapsed As Variant

Dim Vfeats As Variant ' Dimension for reference point
Dim ref_points As Variant
Dim swFeatMgr As SldWorks.FeatureManager
Dim status As Boolean
```

```

Dim boolstatus As Boolean

'DoEvents

'' Timer ''

starttime = Timer

'' Setting up the Solidworks working environment ''

Set swApp = CreateObject("SldWorks.Application")
Set swModel = swApp.ActiveDoc
Set swPart = swModel

'' To Fasten the macro ''

Set modview = swModel.ActiveView
modview.EnableGraphicsUpdate = False           ' Disable graphic updates

swModel.FeatureManager.EnableFeatureTree = False ' Disable feature tree update

Set swFeatMgr = swModel.FeatureManager
Set swSelMgr = swModel.SelectionManager
Set swSelData = swSelMgr.CreateSelectData
vBodies = swPart.GetBodies2(swAllBodies, True)
Set swBody = vBodies(0)
Set swFace = swBody.GetFirstFace
swModel.ClearSelection2 True
f = 1
e = 0
p = 0
Do While Not swFace Is Nothing
    Set swEnt = swFace
    If f = 1 Then
        tab_data(1, 1) = f
    Else: tab_data(p + 1, 1) = f
    End If
    ' Select using IEntity
    bRet = swEnt.Select4(True, swSelData)
    Call main2
    f = f + 1

    On Error Resume Next
    Set swFace = swFace.GetNextFace
Loop

'' Writing in Excel ''

Set exApp = New Excel.Application
If Not exApp Is Nothing Then
    exApp.Visible = True
    If Not exApp Is Nothing Then
        exApp.Workbooks.Add
        Set sheet = exApp.ActiveSheet
        If Not sheet Is Nothing Then
            sheet.Cells(1, 10).Value = "Job Number"
            sheet.Cells(2, 10).Value = "Computing time"
            sheet.Cells(1, 11).Value = "=NOW()"

            sheet.Cells(1, 2) = "Base Surface #"
            sheet.Cells(1, 3) = "Edge #"
            sheet.Cells(1, 4) = "Edge length (mm)"
            sheet.Cells(1, 5) = "Point #"

```

```

        sheet.Cells(1, 6).Value = "X (mm)"
        sheet.Cells(1, 7).Value = "Y (mm)"
        sheet.Cells(1, 8).Value = "Z (mm)"
        sheet.Range("A3:H8003") = tab_data()
    End If
End If
End If

''' Turning back on graphic updates and feature tree

modview.EnableGraphicsUpdate = True
swModel.FeatureManager.EnableFeatureTree = True
boolstatus =
swModel.SetUserPreferenceToggle(swUserPreferenceToggle_e.swDisplayReferencePoints2,
False)
''' Selection and removal of reference points

'ref_points = GetAllFeaturesByType(swModel, ref_type)
'swModel.Extension.MultiSelect2 ref_points, False, Nothing
'swModel.EditDelete

''' Computing time

Secondselapsed = Round(Timer - starttime, 2)
sheet.Cells(2, 11).Value = Secondselapsed
MsgBox "This code ran successfully in " & Secondselapsed & " seconds"

End Sub
Sub main2()

    ''' This macro is used to get the loop of edges surrounding the face

    Dim swApp As SldWorks.SldWorks
    Dim swModel As SldWorks.ModelDoc2
    Dim swPart As SldWorks.PartDoc
    Dim swSelMgr As SldWorks.SelectionMgr
    Dim swFace As SldWorks.Face2
    Dim swLoop As SldWorks.Loop2
    Dim i As Long
    Dim bRet As Boolean
    Set swApp = Application.SldWorks
    Set swModel = swApp.ActiveDoc
    Set swPart = swModel
    Set swSelMgr = swModel.SelectionManager
    Set swFace = swSelMgr.GetSelectedObject6(1, -1)
    'Debug.Print "FaceArea = " & swFace.GetArea * 1000000# & " mm^2"
    'Debug.Print "  LoopCount    = " & swFace.GetLoopCount
    'Debug.Print ""
    Set swLoop = swFace.GetFirstLoop
    Do While Not swLoop Is Nothing
        i = i + 1
        'Debug.Print "  Loop(" & i & ")"
        'Debug.Print "      IsOuter    = " & swLoop.IsOuter
        'Debug.Print "      IsSingular = " & swLoop.IsSingular
        If swLoop.IsOuter Then
            CreateTessLoop swApp, swModel, swLoop
        End If
        On Error Resume Next
        Set swLoop = swLoop.GetNext
    Loop
End Sub
Sub CreateTessLoop(swApp As SldWorks.SldWorks, swModel As SldWorks.ModelDoc2, swLoop As
SldWorks.Loop2)

```

```
' ' This macro goes in the loop, select each edge, measure the,, create 3 ref points
along the edge then use the ref points coordinates to put sketch points in 3D sketch.
So we can link the sketch points with the face.
```

```
' ' When sketch points are created, we get the coordinates
' ' We store all that data in arrays
' ' We export data to excel in main
```

```
'''''' Dimensions for edge and curve selection
Dim vEdgeArr As Variant
Dim vEdge As Variant
Dim swEdge As SldWorks.Edge
Dim swCurve As SldWorks.Curve
Dim swSketch As SldWorks.Sketch
Dim bRet As Boolean ' boolean for 3D sketch rebuild
```

```
'''''' Dimensions for curve length computation
```

```
Dim swSelMgr As SldWorks.SelectionMgr
Dim vCurveParam As Variant
Dim StartPt(2) As Double
Dim EndPt(2) As Double
Dim vStartPt As Variant
Dim vEndPt As Variant
Dim vTessPts As Variant
```

```
'''''' Dimensions for reference points and 3D sketch points creation
```

```
Dim vFeatArr As Variant
Dim swFeatMgr As SldWorks.FeatureManager
Dim swModelDocExt As SldWorks.ModelDocExtension
Dim swSpline As SldWorks.SketchSpline
Dim vFeat As Variant
Dim swFeat As SldWorks.Feature
Dim swRefPt As SldWorks.RefPoint
Dim swRefPtData As SldWorks.RefPointFeatureData
Dim swMathPt As SldWorks.MathPoint
Dim swSkPt As SketchPoint
Dim swMgr As SketchManager
Dim nStatus As Long
Dim cRet As Boolean 'boolean for entity sel
Dim boolstatus As Boolean
Dim i As Long
Dim swEnt As SldWorks.Entity
Dim swSelData As SldWorks.SelectData
Dim ref_point_status As Boolean
```

```
'''''''''''''''' CURVE COMPUTATION ''''''''''''''''
```

```
swModel.Insert3DSketch2 False
swModel.SetAddToDB True
swModel.SetDisplayWhenAdded False
Set swSketch = swModel.GetActiveSketch2
vEdgeArr = swLoop.GetEdges
```

```
For Each vEdge In vEdgeArr
    e = e + 1
    p = p + 1
    tab_data(p, 2) = e
    Set swEdge = vEdge
```

```

Set swCurve = swEdge.GetCurve
vCurveParam = swEdge.GetCurveParams2
StartPt(0) = vCurveParam(0)
StartPt(1) = vCurveParam(1)
StartPt(2) = vCurveParam(2)
EndPt(0) = vCurveParam(3)
EndPt(1) = vCurveParam(4)
EndPt(2) = vCurveParam(5)
vStartPt = StartPt
vEndPt = EndPt
'Debug.Print "CurveType = " & swCurve.Identity
tab_data(p, 3) = Round(swCurve.GetLength2(vCurveParam(6), vCurveParam(7)) *
1000, 3)
'Debug.Print ""

Dim Spline As SldWorks.Edge
Set swFeatMgr = swModel.FeatureManager
Set swSelMgr = swModel.SelectionManager
Set swSelData = swSelMgr.CreateSelectData
Set swEnt = vEdge
Set swMgr = swModel.SketchManager

swModel.ClearSelection2 True
bRet = swEnt.Select4(True, swSelData)
Set Spline = swSelMgr.GetSelectedObject6(1, -1)

vFeatArr = swFeatMgr.InsertReferencePoint(2, 1, 0, 1)
For Each vFeat In vFeatArr
    Set swFeat = vFeat
    Set swRefPt = swFeat.GetSpecificFeature2
    Set swRefPtData = swFeat.GetDefinition
    Set swMathPt = swRefPt.GetRefPoint
    Set swSkPt = swMgr.CreatePoint(swMathPt.ArrayData(0),
swMathPt.ArrayData(1), swMathPt.ArrayData(2))
    tab_data(p, 4) = p
    tab_data(p, 5) = Round(swSkPt.X * 1000, 3)
    tab_data(p, 6) = Round(swSkPt.Y * 1000, 3)
    tab_data(p, 7) = Round(swSkPt.Z * 1000, 3)
    p = p + 1
Next vFeat

swModel.ClearSelection2 True
bRet = swEnt.Select4(True, swSelData)
Set Spline = swSelMgr.GetSelectedObject6(1, -1)

vFeatArr = swFeatMgr.InsertReferencePoint(2, 1, 50, 1)
For Each vFeat In vFeatArr
    Set swFeat = vFeat
    Set swRefPt = swFeat.GetSpecificFeature2
    Set swRefPtData = swFeat.GetDefinition
    Set swMathPt = swRefPt.GetRefPoint
    Set swSkPt = swMgr.CreatePoint(swMathPt.ArrayData(0),
swMathPt.ArrayData(1), swMathPt.ArrayData(2))
    tab_data(p, 4) = p
    tab_data(p, 5) = Round(swSkPt.X * 1000, 3)
    tab_data(p, 6) = Round(swSkPt.Y * 1000, 3)
    tab_data(p, 7) = Round(swSkPt.Z * 1000, 3)
    p = p + 1
Next vFeat

Set swEnt = vEdge
swModel.ClearSelection2 True

```

```

bRet = swEnt.Select4(True, swSelData)
Set Spline = swSelMgr.GetSelectedObject6(1, -1)

vFeatArr = swFeatMgr.InsertReferencePoint(2, 1, 100, 1)
For Each vFeat In vFeatArr
    Set swFeat = vFeat
    Set swRefPt = swFeat.GetSpecificFeature2
    Set swRefPtData = swFeat.GetDefinition
    Set swMathPt = swRefPt.GetRefPoint
    Set swSkPt = swMathPt.ArrayData(0),
swMathPt.ArrayData(1), swMathPt.ArrayData(2))
    tab_data(p, 4) = p
    tab_data(p, 5) = Round(swSkPt.X * 1000, 3)
    tab_data(p, 6) = Round(swSkPt.Y * 1000, 3)
    tab_data(p, 7) = Round(swSkPt.Z * 1000, 3)
    ref_type = swFeat.GetTypeName
Next vFeat

swModel.ClearSelection2 True
Next vEdge
swModel.SetAddToDB False
swModel.SetDisplayWhenAdded True
swModel.Insert3DSketch2 True
bRet = swModel.EditRebuild3

End Sub
Function GetAllFeaturesByType(model As SldWorks.ModelDoc2, typeName As String) As Variant

    Dim swFeatMgr As SldWorks.FeatureManager
    Dim status As Boolean

    Set swFeatMgr = model.FeatureManager

    Dim swRootFeatNode As SldWorks.TreeControlItem

    Set swRootFeatNode = swFeatMgrPane_e.swFeatMgrPaneBottom)

    If Not swRootFeatNode Is Nothing Then
        Dim swFeatsColl As Collection
        Set swFeatsColl = New Collection
        TraverseFeatureNode swRootFeatNode, typeName, swFeatsColl
    Else
        Err.Raise vbError, "", "Failed to get the root node"
    End If

    If swFeatsColl.Count() > 0 Then

        Dim swFeats() As SldWorks.Feature
        ReDim swFeats(swFeatsColl.Count() - 1)

        Dim i As Integer

        For i = 0 To UBound(swFeats)
            Set swFeats(i) = swFeatsColl.item(i + 1)
        Next

        GetAllFeaturesByType = swFeats

    Else
        GetAllFeaturesByType = Empty
    End If

```



```

End If

End Function
Sub TraverseFeatureNode(featNode As SldWorks.TreeControlItem, typeName As String, feats
As Collection)

    If featNode.ObjectType = swTreeControlItemType_e.swFeatureManagerItem_Feature Then

        Dim swFeat As SldWorks.Feature
        Set swFeat = featNode.Object

        If swFeat.GetTypeName2() = "HistoryFolder" Then
            Exit Sub
        End If

        If LCase(swFeat.GetTypeName2) = LCase(typeName) Then
            If Not Contains(feats, swFeat) Then
                'swFeat.Select2 True, -1
                feats.Add swFeat
            End If
        End If

    End If

End If

Dim swChildFeatNode As SldWorks.TreeControlItem
Set swChildFeatNode = featNode.GetFirstChild()

While Not swChildFeatNode Is Nothing
    TraverseFeatureNode swChildFeatNode, typeName, feats
    Set swChildFeatNode = swChildFeatNode.GetNext
Wend

End Sub
Function Contains(coll As Collection, item As Object) As Boolean

    Dim i As Integer

    For i = 1 To coll.Count
        If coll.item(i) Is item Then
            Contains = True
            Exit Function
        End If
    Next

    Contains = False

End Function

```