The generation of 3D data for rapid manufactured textiles

This item was submitted to Loughborough University’s Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: [https://dspace.lboro.ac.uk/2134/25851](https://dspace.lboro.ac.uk/2134/25851)

Publisher: © Guy A. Bingham

Rights: This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: [https://creativecommons.org/licenses/by-nc-nd/4.0/](https://creativecommons.org/licenses/by-nc-nd/4.0/)

Please cite the published version.
The generation of 3D data for Rapid Manufactured textiles

by

Guy A Bingham

Doctoral Thesis Submitted in partial fulfilment of the requirements for the award of

Doctor of Philosophy of Loughborough University

June 2007

© Guy A Bingham 2007
Abstract

Rapid Manufactured (RM) textiles were first proposed by Freedom of Creation in 1999. Initially, these textile type structures were viewed as a novelty of additive manufacturing techniques with no specific application. However, the geometric complexity capabilities of additive manufacturing techniques means that the scope of RM textiles is far greater than initially realised. The research undertaken within this thesis regarding this novel area of textiles and Rapid Manufacturing has shown the scope of RM textiles to lie in the area of high performance and smart textiles.

Increasingly, high performance and smart textiles are seen as relevant research areas, with the incentive being the realisation of high performance and smart clothing with increased and desirable functionalities. However, the current manufacture of high performance and smart textiles has been shown to be an arduous process, involving complex manufacturing operations and techniques, fraught with design for manufacture and assembly constraints, which are currently restricting the complexity of the textile structures that can be actually achieved. This research suggests that future additive manufacturing techniques provide an elegant solution to these current manufacturing difficulties and therefore validate the novel research area of RM textiles.

This research investigates the generation of 3D RM textile geometric data essential for their manufacture by additive manufacturing techniques. It demonstrates that while the generation of planar or flat sheets of RM textile structures with high levels of geometric complexity can be efficiently created using conventional modelling techniques of Computer Aided Design (CAD) software, the generation of conformal textiles is inefficient, time-consuming and error prone. Further investigations utilising experimental textile modelling software provided an initial methodology for the efficient generation of conformal RM textiles to be established. However, the initial methodology was limited and restricted the main incentive for the creation of RM textiles, geometric complexity.
A further methodology was then presented that addresses these limitations, requiring firstly, the generation of a uniform and equidistant mapping surface mesh, and secondly, a complex geometry mapping tool capable of mapping complex geometry to such a mesh accurately and efficiently.

The research demonstrates the complexity of generating the required uniform and equidistant mapping mesh and highlights that currently available meshing techniques are incapable of generating such a mesh structure for all curved surface geometries. The generation of the required mapping mesh structure was then investigated and a novel meshing technique and algorithm developed to attain such surface mesh structures. The work then addresses the mapping of complex geometric 3D data to the surface mesh structures and again develops a technique and system capable of achieving this aim.

The research therefore culminates in a complete methodology for the efficient generation of conformal RM textile structures of an increased geometric complexity that will enable further research to be undertaken in the novel area of RM textiles.
Acknowledgements

I would like to thank all those people who have in some way or another assisted me over the course of my research. While it is not possible to mention all those that have given their assistance, I thank you, but special mention must go to the following:

Firstly to Dr. Richard Hague for giving me the opportunity to undertake this research, his guidance and assistance have been invaluable, thanks Boss.

I would also like to thank both Dr. Jonathan Crookston and Dr. Richard Buswell for their invaluable coding expertise and patience, which could always be relied upon in times of sheer debugging panic, cheers boys.

The support and friendship of members past and present from the Rapid Manufacturing Research Group, and various other people from the Wolfson School, many thanks to you all.

Thank you to the IMCRC Loughborough for the funding of several RM textile related projects that have aided the progression of my research, and to project partners, Freedom of Creation and the University of Nottingham.

To family and friends outside the echelons of academia near and afar who have encouraged and supported my endeavours, with a special mention to Truss (Simon Trussler), Cockney (Michael Lansley) and lastly but by no means least, my parents, who have remained proud of me throughout, thank you.

Finally, a very special thank you to my housemate and learned friend Dr. Candi Majewski, whose friendship, encouragement, patience, sage-like advice and proofreading skills could always be relied upon in troubled times. You have without question eased this difficult process and for that I owe you a huge debt of gratitude, thank you so very much.
Table of contents

ABSTRACT ........................................................................................................................................ 1
ACKNOWLEDGEMENTS ................................................................................................................. III
TABLE OF CONTENTS ................................................................................................................... IV
GLOSSARY OF TERMS .................................................................................................................. VIII
NOMENCLATURE ........................................................................................................................... XI
TABLE OF FIGURES .................................................................................................................. XII
LIST OF TABLES ........................................................................................................................ XV

CHAPTER 1: INTRODUCTION .................................................................................................... 1
1.1 INTRODUCTION .................................................................................................................. 1
1.2 OVERVIEW OF RESEARCH UNDERTAKEN .................................................................. 2
1.3 RESEARCH OBJECTIVES ............................................................................................... 3

CHAPTER 2: TEXTILES ............................................................................................................. 5
2.1 INTRODUCTION TO TEXTILES ....................................................................................... 5
2.2 THREE MAIN AREAS OF TEXTILE PRODUCTS ............................................................. 5
2.3 THE HIERARCHICAL STRUCTURE OF TEXTILES ............................................................ 6
2.4 FIBRES ............................................................................................................................... 6
   2.4.1 Natural fibres ........................................................................................................... 7
   2.4.2 Synthetic fibres ........................................................................................................ 7
   2.4.3 Physical attributes of fibres ..................................................................................... 9
   2.4.4 Microfibres ............................................................................................................. 11
   2.4.5 Nanofibres ............................................................................................................ 12
2.5 TEXTILE MANUFACTURING TECHNIQUES .................................................................. 12
   2.5.1 Weaving .................................................................................................................. 12
   2.5.2 Innovation in the woven textile industry ............................................................... 14
   2.5.3 Knitting ................................................................................................................... 15
   2.5.4 Innovation in the Knitting textile industry ........................................................... 17
   2.5.5 Non woven ............................................................................................................ 18
2.6 HIGH PERFORMANCE TEXTILES .................................................................................. 18
   2.6.1 High performance clothing .................................................................................... 20
   2.6.2 Other applications of functional textiles .............................................................. 21
2.7 SMART TEXTILES .......................................................................................................... 22
   2.7.1 Smart clothing ........................................................................................................ 23
   2.7.2 Limitations in the manufacture of smart clothing ................................................ 26
2.8 SUMMARY OF TEXTILES ............................................................................................... 27
   2.8.1 Final comments on textiles ................................................................................... 29

CHAPTER 3: RAPID PROTOTYPING AND RAPID MANUFACTURING ................................ 30
3.1 DEFINITION ...................................................................................................................... 30
   3.1.1 Background and application ................................................................................. 30
3.2 RAPID PROTOTYPING .................................................................................................... 30
   3.2.1 RP process classifications ..................................................................................... 33
   3.2.2 RP Resolution ....................................................................................................... 33
   3.2.3 Rapid Prototyping advantages and limitations ..................................................... 35
3.3 RAPID MANUFACTURING ............................................................................................ 36
   3.3.1 Design potential of Rapid Manufacturing ............................................................ 37
   3.3.2 Rapid Manufacturing of Metals .......................................................................... 40
CHAPTER 4: COMPUTER AIDED DESIGN

4.1 Definition
4.2 Background and Application
4.2.1 Development of CAD
4.2.2 Limitations in conventional CAD for RM
4.2.3 The interface between CAD and RP and RM
4.3 Applications of CAD in the textile industry
4.3.1 Computer Aided Apparel Design (CAAD)
4.3.2 DCC CAD
4.3.3 CAD for textiles
4.4 Summary of CAD
4.4.1 Final comments on CAD

CHAPTER 5: POTENTIAL AND INITIAL MODELLING OF RAPID MANUFACTURED TEXTILES

5.1 The potential of RM textiles
5.2 Initial Exploration
5.2.1 Problem definition
5.3 Initial research aim
5.3.1 Initial approach
5.4 Initial modelling of RM textile structures
5.4.1 Introduction
5.5 Initial modelling with conventional CAD
5.5.1 Initial modelling of flat RM textiles
5.5.2 Initial modelling of conformal RM textile structures
5.6 Initial modelling with textile CAD
5.6.1 Limitation
5.7 Discussion of initial modelling
5.7.1 Conclusion

CHAPTER 6: RESEARCH METHODOLOGY

6.1 Problem Identification
6.2 Research Aims
6.3 Research Approach and Objectives

CHAPTER 7: MESH GENERATION

7.1 Introduction
7.2 Mesh generation
7.3 Introduction to FEA
7.4 FE Pre-processors
7.5 Meshing techniques
7.5.1 Octree
7.5.2 Advancing Front
7.5.3 Delaunay triangulation
7.6 Post-processing
7.6.1 Smoothing operations
7.6.2 Cleanup operations
7.7 Surface meshing
7.7.1 Indirect surface meshing
7.7.2 Direct surface meshing
7.8 FEA surface meshing techniques capable of generating high quality mesh structures
7.8.1 Direct sphere packing
CHAPTER 11: INVESTIGATION OF A NEW COMPLEX GEOMETRY MAPPING TOOL

11.1 INTRODUCTION ............................................. 182

11.2 GEOMETRY MAPPING FUNDAMENTALS ....................... 184

11.2.1 2D mapping to a curve .................................... 184

11.2.2 3D mapping to a surface mesh .......................... 186

11.3 MAPPING METHODOLOGY .................................. 189

11.4 STAGE ONE: IMPORT CAD DATA AND SURFACE MESH DATA ......................................................... 192

11.5 STAGE TWO: CALCULATE SURFACE NORMAL AND LOCAL COORDINATE SYSTEM AT MESH NODES AND ELEMENT MIDPOINTS ......................................................... 193

11.6 STAGE THREE: ORIENTATE LINK GEOMETRY FROM GLOBAL COORDINATE SYSTEM TO LOCAL COORDINATE SYSTEM .......................................................... 199

11.6.1 2D transformation .......................................... 199

11.6.2 3D Transformation .......................................... 205

11.7 STAGE FOUR: TRANSLATION OF ORIENTATED LINK GEOMETRY TO MAPPING LOCATION .................. 208

11.8 FINAL STL CREATION ......................................... 210

11.9 CODING THE DEVELOPED MAPPING METHODOLOGY .......................................................... 210

CHAPTER 12: VALIDATION OF MAPPING OF COMPLEX GEOMETRIES TO SURFACE MESHES

12.1 INTRODUCTION ............................................. 211

12.2 GEOMETRY ORIENTATION .................................... 212

12.2.1 Surface mesh 1 ............................................. 213

12.2.2 Surface mesh 2 ............................................. 214

12.3 GEOMETRIC COMPLEXITY .................................. 215

12.3.1 Test geometry 2: Teapot .................................. 217

12.3.2 Test geometry 3: Mobius knot .......................... 219

12.3.3 Test geometry 4: Complex lattice structure .......... 220

12.4 DISCUSSION OF RESULTS .................................... 221

CHAPTER 13: FINAL CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK ................. 223

13.1 ATTAINED RESEARCH OBJECTIVES ......................... 223

13.2 CONCLUSIONS .................................................. 224

13.3 RECOMMENDATIONS FOR FURTHER WORK ............... 226

REFERENCES ................................................................ 229
## Glossary of terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition/Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D</td>
<td>Two dimensional</td>
</tr>
<tr>
<td>3D</td>
<td>Three dimensional</td>
</tr>
<tr>
<td>3D modelling</td>
<td>Creation of 3D geometric structures in CAD</td>
</tr>
<tr>
<td>3DP</td>
<td>3D printing</td>
</tr>
<tr>
<td>Additive manufacturing</td>
<td>Manufacturing by the addition of build materials</td>
</tr>
<tr>
<td>Advancing front</td>
<td>FEA meshing technique for geometry boundary</td>
</tr>
<tr>
<td>Animation</td>
<td>Capturing of motion of 3D data</td>
</tr>
<tr>
<td>CAAAD</td>
<td>Computer Aided Design and Drafting</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer Aided Design</td>
</tr>
<tr>
<td>CAE</td>
<td>Computer Aided Engineering</td>
</tr>
<tr>
<td>CAM</td>
<td>Computer Aided Manufacture</td>
</tr>
<tr>
<td>Cartesian space</td>
<td>3D space described by three axes</td>
</tr>
<tr>
<td>CFD</td>
<td>Computation Fluid Dynamics</td>
</tr>
<tr>
<td>Chain maille</td>
<td>Medieval textile armour</td>
</tr>
<tr>
<td>Circular array</td>
<td>2D circular configuration of 3D geometries</td>
</tr>
<tr>
<td>CLI</td>
<td>Common Layer Interface</td>
</tr>
<tr>
<td>Conformal</td>
<td>Adhering to a curved surface or structure</td>
</tr>
<tr>
<td>Connectivity</td>
<td>Relationship between nodes or vertices</td>
</tr>
<tr>
<td>Conventional textiles</td>
<td>Fibre-based textiles</td>
</tr>
<tr>
<td>DCC</td>
<td>Digital Content Creation</td>
</tr>
<tr>
<td>Delaunay triangulation</td>
<td>Methodology for generating triangular elements from mesh node</td>
</tr>
<tr>
<td>DFMA</td>
<td>Design for manufacture and assembly</td>
</tr>
<tr>
<td>Drape</td>
<td>A textile's ability to conform</td>
</tr>
<tr>
<td>Element</td>
<td>Geometry created by the connecting of nodes</td>
</tr>
<tr>
<td>Equidistance</td>
<td>All nodes being the same dimension apart</td>
</tr>
<tr>
<td>Equidistant mesh structure</td>
<td>Constant nodal spacing</td>
</tr>
<tr>
<td>FDM</td>
<td>Fused deposition modelling</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FEA</td>
<td>Finite Element Analysis</td>
</tr>
<tr>
<td>Fibres</td>
<td>Base component of conventional textiles</td>
</tr>
<tr>
<td>Geometric complexity</td>
<td>The complexity of geometry</td>
</tr>
<tr>
<td>Global coordinate system</td>
<td>Cartesian coordinate system</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HGPL</td>
<td>Hewlett-Packard Graphical Language</td>
</tr>
<tr>
<td>Hierarchical assembly</td>
<td>A greater structure containing main smaller structures</td>
</tr>
<tr>
<td>High performance textiles</td>
<td>Textiles with performance functionalities</td>
</tr>
<tr>
<td>Histogram</td>
<td>A Frequency plot of a sample</td>
</tr>
<tr>
<td>IGES</td>
<td>Initial Graphical Exchange system</td>
</tr>
<tr>
<td>Intersection spline</td>
<td>The boundary created by intersecting bodies</td>
</tr>
<tr>
<td>Knitted textile</td>
<td>Textile manufactured for knitted yarns</td>
</tr>
<tr>
<td>Lagrange interpolation</td>
<td>Polynomial determination of an equation of graph points</td>
</tr>
<tr>
<td>Linear array</td>
<td>2D configuration of 3D geometries</td>
</tr>
<tr>
<td>Links</td>
<td>Singular interlink base component of textile structures</td>
</tr>
<tr>
<td>LMI</td>
<td>Layer Manufacturing Interface</td>
</tr>
<tr>
<td>Local coordinate system</td>
<td>A coordinate system at a location on a surface</td>
</tr>
<tr>
<td>LOM</td>
<td>Laminated object manufacture</td>
</tr>
<tr>
<td>Mesh</td>
<td>A structure contain nodes and elements</td>
</tr>
<tr>
<td>Natural fibres</td>
<td>Fibres created from animal fur or plants</td>
</tr>
<tr>
<td>Node</td>
<td>A location of a geometry described by a mesh</td>
</tr>
<tr>
<td>Non-woven textile</td>
<td>Textile manufactured from fibres</td>
</tr>
<tr>
<td>NURBS</td>
<td>Non-rational B-spline Surfaces</td>
</tr>
<tr>
<td>Octree</td>
<td>FEA meshing technique</td>
</tr>
<tr>
<td>Orientation</td>
<td>The intended out-facing direction of a body</td>
</tr>
<tr>
<td>Orthogonal</td>
<td>Two ventricular vectors to a third</td>
</tr>
<tr>
<td>PIC</td>
<td>Picture format</td>
</tr>
<tr>
<td>Quadrilateral surface mesh</td>
<td>A mesh made of quadrilateral elements</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Rapid Manufacturing (RM)</td>
<td>Additive manufacture of end use components</td>
</tr>
<tr>
<td>Rapid Prototyping (RP)</td>
<td>Additive manufacture of prototype components</td>
</tr>
<tr>
<td>Rendering</td>
<td>Shading and colouring of 3D data</td>
</tr>
<tr>
<td>Resolution</td>
<td>Build feature dimensional capability</td>
</tr>
<tr>
<td>RM textiles</td>
<td>Textile type structures additively manufactured</td>
</tr>
<tr>
<td>RPI</td>
<td>Rapid Prototype Interface</td>
</tr>
<tr>
<td>RVE</td>
<td>Representative Volumetric Element</td>
</tr>
<tr>
<td>SLA</td>
<td>Stereolithography</td>
</tr>
<tr>
<td>SLC</td>
<td>Stereolithography Contouring</td>
</tr>
<tr>
<td>SLS</td>
<td>Selective laser sintering</td>
</tr>
<tr>
<td>Smart clothing</td>
<td>Intelligent garments</td>
</tr>
<tr>
<td>Smart textiles</td>
<td>Textiles with intelligence</td>
</tr>
<tr>
<td>Sphere packing</td>
<td>Packing of sphere tangentially</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>A measure of dispersion in a sample</td>
</tr>
<tr>
<td>STEP</td>
<td>Standard for the Exchange of Product Model Data</td>
</tr>
<tr>
<td>STL</td>
<td>Stereolithography language</td>
</tr>
<tr>
<td>Surface normal</td>
<td>A vector ventricular to a surface at a specified location</td>
</tr>
<tr>
<td>Synthetic fibres</td>
<td>Manmade fibres</td>
</tr>
<tr>
<td>Transformation</td>
<td>The orientation for one coordinate system to a second</td>
</tr>
<tr>
<td>Translation</td>
<td>Relocation from one location to a second</td>
</tr>
<tr>
<td>Triangular surface mesh</td>
<td>A structure containing nodes made of triangular elements</td>
</tr>
<tr>
<td>Uniform mesh structures</td>
<td>Constant node to element ratio</td>
</tr>
<tr>
<td>Unit vector</td>
<td>A vector with a length of one unit</td>
</tr>
<tr>
<td>Vertex</td>
<td>Corner geometry of a triangle</td>
</tr>
<tr>
<td>Woven textile</td>
<td>textile manufactured from woven yarns</td>
</tr>
<tr>
<td>Yarns</td>
<td>Tread manufactured from fibres</td>
</tr>
</tbody>
</table>
### Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_x$</td>
<td>Polynomial equation</td>
</tr>
<tr>
<td>$x_*$</td>
<td>X or Y axis recorded coordinate</td>
</tr>
<tr>
<td>$f(x_*)$</td>
<td>Z axis recorded coordinate at X or Y axis value</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Longitude angle</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Colatitude angle</td>
</tr>
<tr>
<td>$x_0$</td>
<td>x axis sphere centre coordinate</td>
</tr>
<tr>
<td>$y_0$</td>
<td>y axis sphere centre coordinate</td>
</tr>
<tr>
<td>$z_0$</td>
<td>z axis sphere centre coordinate</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>$x$</td>
<td>Value recorded for standard deviation</td>
</tr>
<tr>
<td>$\bar{x}$</td>
<td>Mean of all values</td>
</tr>
<tr>
<td>$n$</td>
<td>Number of values</td>
</tr>
<tr>
<td>$</td>
<td>a \times b</td>
</tr>
<tr>
<td>$</td>
<td>a</td>
</tr>
<tr>
<td>$i$</td>
<td>x component of vector</td>
</tr>
<tr>
<td>$j$</td>
<td>y component of vector</td>
</tr>
<tr>
<td>$k$</td>
<td>z component of vector</td>
</tr>
<tr>
<td>$\hat{a}$</td>
<td>Unit vector of vector a</td>
</tr>
<tr>
<td>$X$</td>
<td>x component of global coordinate system</td>
</tr>
<tr>
<td>$X'$</td>
<td>x component of local coordinate system</td>
</tr>
</tbody>
</table>
Figure 7-9: (left) Targeted surface geometry for meshing and (u-v) representation, (right) 2D elements mapped to (u-v) representation

Figure 7-10: (left) 3D mesh elements mapped to 3D target surface, (right) Inverted target surface showing mesh elements

Figure 7-11: (left) Target surface, (right) initial sphere placement using advancing front technique

Figure 7-12: (left) Target surface filled with spheres, (right) mesh creation from centre points of spheres

Figure 7-13: Completed surface mesh showing high quality elements

Figure 7-14: Platonic solution for N points on a sphere, N = 3 points (left) & N = 6 points (right)

Figure 7-15: Platonic solution for N points on a sphere, N = 8 points

Figure 7-16: Platonic solution for N points on a sphere, N = 12 points (left) & N = 20 points (right)

Figure 7-17: (left) Non-overlapping spheres, (right) Sphere covering [168]

Figure 7-18: Distribution of points for changing values of s [176]

Figure 7-19: Near uniform distribution of points on torus [176]

Figure 8-1: (Left) Planar surface for sphere pack meshing, (right) advancing front initial sphere placement

Figure 8-2: (Left) Full coverage of non uniform spheres, (right) mesh creation from sphere centres

Figure 8-3: Final non uniform or fully equidistant mesh structure

Figure 8-4: Sphere packing meshing of a surface where the boundary is a function of the spheres

Figure 8-5: Uniform and equidistant mesh structure created due to surface boundary configuration

Figure 8-6: (Left) Planar square surface, (right) initial sphere placement central to surface

Figure 8-7: (Left) Full coverage of uniform spheres, (right) mesh creation from sphere centres

Figure 8-8: Fully uniform and equidistant mesh structure

Figure 9-1: Initial sphere placement

Figure 9-2: Valid location creation

Figure 9-3: Mesh growth

Figure 9-4: Inaccurate sphere packing from random mesh growth

Figure 9-5: Sphere to sphere intersection showing both sphere to surface intersections

Figure 9-6: First sphere placement

Figure 9-7: Intersection curve created from sphere-surface intersection

Figure 9-8: Second sphere placement, centre point constrained to intersection curve

Figure 9-9: Valid and invalid intersection locations

Figure 9-10: Duplicated intersection sites

Figure 9-11: Invalid sites outside the surface boundary

Figure 9-12: Target surface completely packed with spheres, only invalid intersections now exist

Figure 9-13: Numbered known sphere centres

Figure 9-14: Schematic of sphere centres, (right) triangular element creation

Figure 9-15: Complete uniform and equidistant mesh

Figure 9-16: Flow diagram of planar sphere packing methodology

Figure 9-17: Target curved surface geometry

Figure 9-18: Initial sphere placement

Figure 9-19: Second sphere placement and intersection spline

Figure 9-20: Valid and invalid site definition

Figure 9-21: Initial glance

Figure 9-22: Perspective and close up of valid sites

Figure 9-23: Points created in relation to sphere centres

Figure 9-24: Inaccurate site creation

Figure 9-25: Sphere to sphere intersection spline shown as black

Figure 9-26: Intersection of sphere-sphere splines

Figure 9-27: Intersection points above and below the surface

Figure 9-28: Clustered sites creation

Figure 9-29: Triangulation of clustered sites

Figure 9-30: Erroneous sites created inside the red circles

Figure 9-31: Target surface fully packed with spheres

Figure 9-32: (left) Triangulation of nodes/sphere centres, (right) final mesh structure

Figure 9-33: Surface creation in CAD

Figure 9-34: Spline generation in CAD

Figure 9-35: Surface generated in CAD, (right) YZ and XZ splines used for its construction
Figure 9-36: XZ spline of CAD generated surface showing control points
Figure 9-37: Polynomial curve plotted for x =0-100
Figure 9-38: Surface plot of developed surface Equation 9.8
Figure 9-39: Sphere representation
Figure 9-40: Sphere-surface intersection
Figure 9-41: Sphere surface intersection
Figure 9-42: Z coordinate of sphere and surface for \( \theta = 0^\circ \) and \( \alpha = 0^\circ \)
Figure 9-43: Sphere coordinated below target surface
Figure 9-44: Sphere-surface spline showing intersection sites
Figure 9-45: 3D splines projected to mapping plane creating 2D curves
Figure 10-1: Spline creation for quadratic surfaces, showing control point locations
Figure 10-2: Surface 1 and construction data
Figure 10-3: Surface 2 and construction data
Figure 10-4: Surface 3 and construction data
Figure 10-5: Surface 4 and construction data
Figure 10-6: Surface 5 and construction data
Figure 10-7: Mesh structure generated from 1 unit sphere
Figure 10-8: Histogram of mesh 6 surface 1
Figure 10-9: Standard deviation against sphere radius
Figure 10-10: Gambit generated 5 unit mesh
Figure 10-11: Patran generated 5 unit mesh
Figure 10-12: Intersecting-sphere generated 5 unit mesh
Figure 10-13: Mesh structure generated from 1 unit sphere
Figure 10-14: Histogram of mesh 6 surface 2
Figure 10-15: Standard deviation against sphere radius
Figure 10-16: Gambit generated 5 unit mesh
Figure 10-17: Patran generated 5 unit mesh
Figure 10-18: Intersecting-sphere generated 5 unit mesh
Figure 10-19: Mesh structure generated from 1 unit sphere
Figure 10-20: Histogram of mesh 6 surface 3
Figure 10-21: Standard deviation against sphere radius
Figure 10-22: Gambit generated 5 unit mesh
Figure 10-23: Patran generated 5 unit mesh
Figure 10-24: Intersecting-sphere generated 5 unit mesh
Figure 10-25: Mesh structure generated from 1 unit sphere
Figure 10-26: Histogram of mesh 6 surface 4
Figure 10-27: Standard deviation against sphere radius
Figure 10-28: Gambit generated 5 unit mesh
Figure 10-29: Patran generated 5 unit mesh
Figure 10-30: Intersecting-sphere generated 5 unit mesh
Figure 10-31: Mesh structure generated from 1 unit sphere
Figure 10-32: Closer examination of the mesh structure, (top view) without surface geometry
Figure 10-33: Histogram of mesh 6 surface 6
Figure 10-34: Example spheres constraint to identical surfaces
Figure 10-35: Region of surface encapsulated by both spheres
Figure 10-36: Apparent surface curvature experienced by both spheres
Figure 11-1: Structure of FE meshes
Figure 11-2: Link mapping to nodes (left) and nodes and element midpoints (right)
Figure 11-3: Complex link mapped to node only
Figure 11-4: Simple 2D mapping example, (left) isometric view, (right) frontal view
Figure 11-5: Inaccurate mapping of geometry
Figure 11-6: Normals of curve at the nine mapping locations
Figure 11-7: Accurately mapped geometry matching curve normals at the mapping locations
Figure 11-8: Inaccurate surface mapping
Figure 11-9: Surface normal of mapping surface at a specified location
List of tables

Table 9.1: Lagrange Interpolation terms for 5 control points ................................................................. 137
Table 10.1: Standard deviations recorded from surface 1 ........................................................................ 156
Table 10.2: Standard deviation recorded from surface 2 ........................................................................ 161
Table 10.3: Standard deviation recorded from surface 3 ......................................................................... 167
Table 10.4: Standard deviation recorded from surface 5 ........................................................................ 172
Chapter 1: Introduction

1.1 Introduction

In 1999, Freedom of Creation (FOC) [1], an industrial design partnership exploring the use of Rapid Prototyping for product design, developed and patented the idea of using Rapid Prototyping techniques to manufacture textiles [2]. These textile type structures were later to become known as Rapid Manufactured (RM) textiles.

FOC established that by interlinking simple geometries into a larger hierarchical assembly, the resulting manufactured structure incorporated free movement and drape characteristics, properties that make them directly comparable to conventional manufactured textiles. The designs of these RM textiles were not based on fibres or yarns, as used in conventional textiles, but on geometries such as tori as used in the manufacture of chain maille armour [3]. This is demonstrated in Figure 1-1.

![Initial RM textile design](image1.png)

Figure 1-1: Initial RM textile design
By utilising a fundamental advantage of RP, the capability to manufacture free moving assemblies in one manufacturing process, and by taking inspiration from chain maille armour, FOC were able to create a portfolio of RM textile designs and several RM textile based consumer products, such as handbags. Although some initial development by FOC was initiated, this centred on the manufacture of flat sheets of RM textiles based on chain maille designs at the macro scale.

However, the scope of textile structures manufactured using additive techniques goes well beyond that of flat chain maille and RM textiles present a novel application for current RP and future RM technologies. However, if this process of manufacturing textiles by additive manufacturing techniques is to be seriously considered, the first issue that requires resolving is whether there is actually any requirement for RM textiles. While it is possible to use additive technologies to manufacture a structure that has properties comparable to conventional textiles, the potential benefit in using these manufacturing techniques needs to be established. If indeed there is a benefit in manufacturing textile structures by RP or RM technologies, their potential applications need to be identified.

1.2 Overview of research undertaken

In order to address the fundamental issues of RM textiles, a review of current textiles, their manufacture and applications was required. Therefore the first topic of investigation in the literature review is the area of conventional textiles to establish whether RM textiles could provide any benefit. Once it had been ascertained that RM textiles do provide an advantage in geometric complexity for the manufacture of high performance textiles, the second topic of investigation was the manufacturing process itself. RP and RM technologies and their capabilities were investigated to establish if the advantages offered by RM textiles will eventually be attainable through future RM systems. The final topic of investigation in the literature review was CAD and its capabilities for generating RM textile data.
Initial experimental modelling of 3D RM textile structures formed the basis of a further chapter of work, exploring the limitations in this process and finalising a research direction. The research direction was then finalised into specific research objectives. The remaining body of chapters document the research undertaken addressing the research direction and attaining the specific research objectives. To aid the reader's understanding of the developed direction of the research and thesis structure, a flow diagram of the thesis is presented in Figure 1.2

1.3 Research objectives

The research objectives of this research are developed through the literature review and initial experimental modelling of 3D RM textile structures, and are documented accordingly in Chapter 6: Research Methodology. However, for clarity, the specific research objectives are presented here in two research issues:

**Research Issue 1:**
1. Extended literature review of current surface meshing techniques and related issues
2. Investigation of uniform and equidistant meshing
3. Potential testing and validation of any existing or newly developed meshing technique

**Research Issue 2:**
1. Investigation of complex geometry mapping
2. Potential testing and validation of any existing or newly developed mapping tool

The successful conclusion of the research objectives for both these research issues will therefore be the creation of a new methodology for the generation of conformal RM textile data of a high geometric complexity suitable for manufacture and the completion of this research.
Rapid Manufactured Textiles

Conventional Textiles ➔ Rapid Prototyping and Rapid Manufacturing ➔ Computer Aided Design

Rapid Manufactured textiles, their potential, initial research aim and modelling

Research Methodology

Research Objectives

Uniform and equidistant mapping mesh generation

Mesh Generation ➔ Equidistant point generation

Investigation of a new meshing algorithm

Complex geometry mapping

Investigation of a new mapping tool

Experimental testing and validation of new mapping tool

Complete methodology for the generation of complex and uniform RM textile data

Conclusions and Further Work

Legend

- Thesis subject
- Core chapters
- Literature topics
- Research objectives
- Investigation
- Experimental testing
- Flow arrow

Figure 1-2: Flow chart of thesis
Chapter 2: Textiles

2.1 Introduction to textiles

The manufacture of textiles for clothes can be considered nearly as old as mankind itself, where it is widely accepted that our earliest ancestors manufactured simple clothing for protection against harsh environmental conditions experienced during the Ice age. From these very simple early applications of creating garments for protective purposes, using such materials as animal furs and skins, the textile industry has evolved to become an integral component of the modern industrial world. From such humble beginnings, the modern textile industrial machine has become responsible for the very first working factories, workers unions and health and safety regulations so apparent in today's manufacturing industries [4,5].

Whilst in the past, the textile industry was mainly thought of as the manufacture of fabrics used in the apparel sector, the following text will show that the textile manufacturing industry has become extremely diverse. It is important to understand that all textiles fall into three categories: woven, knitted and non-woven. Each type of textile offers different aesthetic and mechanical properties and has very different applications. The process of manufacturing each of the cited categories will be discussed in further detail in later sections.

2.2 Three main areas of textile products

The textile industry is split into three major sectors, each producing entirely different products all grouped under the generic name of textiles.

1. The apparel sector, where the manufacture of textiles is solely based on the needs of the fashion and performance clothing industry.

2. The interior furnishings sector, relating to the textiles used in furniture, carpeting, bedding and decorative furnishings.
3. The industrial sector, where textiles are used to create luggage, buildings, bandages, filters and so on.

The current textile market sales are divided approximately 40% apparel, 40% interior furnishings and 20% industrial [4,5].

2.3 The hierarchical structure of textiles

All textiles possess a hierarchical structure starting with the fibre. Fibres are the smallest part of any textile and singularly are extremely fine, hair-like structures. The next component in the hierarchy of textiles is the yarn or thread. Most textile materials contain yarns, where the actual yarns are continuous thread-like strands composed of many fibres that have been twisted, spun or crimped together. Yarns are available in many forms, offering different mechanical and aesthetic properties [6]. In the construction of any yarn based textile, the opportunity exists to use yarns constructed of several different types of fibres effecting the properties of the yarn and therefore the finally derived textile [4,6]. The exception to this process is for non-woven textiles, which can be manufactured directly from fibres.

Having created the fibres and then sequentially the yarns, the completion of the textile requires the weaving of the yarns in woven textiles or knitting of the yarns in knitted textiles to complete the hierarchy and therefore the textile.

2.4 Fibres

As previously stated, fibres are the smallest component of any textile. Fibres have a comparatively high length to width ratio producing the flexibility necessary within the final textile structures [6]. All fibres currently used in the construction of textiles can be categorised into two groups: natural and synthetic [4,6]:..
2.4.1 Natural fibres
Natural fibres are obtained from animals or plants [4,5,6]. Generally, plant or vegetable fibres are produced using either the stem (such as flax, hemp, jute and ramie) the leaves (for example sisal or abaca) or the seeds, (like cotton or kapok). Animal fibres used in the construction of apparels serve insulation purposes as they do with animals. Commonly used animal fibres such as wool, cashmere, mohair and vicuna offer incredible heat insulation properties with stretch and drape properties not yet matched by manufactured fibres. Silk is also considered an animal fibre, although, this is not entirely accurate as the silk fibres are taken from the cocoon of the silk worm as demonstrated in Figure 2-1 and not the fur of an animal.

![Silk worm and silk worm cocoon](image)

Figure 2-1: Silk worm and silk worm cocoon

2.4.2 Synthetic fibres
Synthetic fibres are produced from chemical or polymeric solutions that are extruded through a die called a spinnerette, as demonstrated schematically in Figure 2-2 [6]. Similar to a shower head, fine liquid streams of polymeric solution extrude from the die and harden into continuous strands known as filament fibres. The size, shape and number of the holes within the die vary according to the type of filament fibre and the materials being processed. A small spinnerette may only have ten holes where a large spinnerette could have up to ten thousand [6].
Different techniques are used in the hardening of the fibres as they exit from the die; this is entirely dependent on the type of polymeric solution used [6]. The most common forms of hardening fall into three categories:

- **Dry spinning:** The fibre solution is mixed with a solvent before being extruded through the die. When the fibre exits from the die, warm air is applied that causes the solvent to evaporate. Once evaporated, the fibre solutions begin to harden into a continuous filament fibre. Acetate and Modactylic fibres are manufactured using this process [4,6].

- **Wet spinning:** The fibre solution is extruded through the die and into a liquid solution bath. When immersed in the solution, the extruded fibres begin to harden into continuous filament fibres. Acrylic and Viscose Rayon fibres are produced using this process [4,6].

- **Melt spinning:** This process is directly linked to any normal extrusion process. Solid material is melted into a liquid form and extruded through the die; during exit, the cool air causes the extrudate to harden producing continuous filament fibres. Glass, Nylon, Polyester and Olefin fibres are all produced using this method [4,6].
2.4.3 Physical attributes of fibres

The structure of fibres contributes to the performance characteristic of the final textile and any products derived from it [4,6]. Some fibre characteristics contribute favourably, whereas others can have a detrimental effect on the end use application. Fibre properties are determined by their physical attributes, chemical composition and molecular orientation [6].

Fibres can vary enormously in length, ranging from 25mm to over one mile [4]. The standard measurement units of fibre length in the textile industry are inches and yards. In order to categorise between such long and short fibres, the industry refer to fibres measured in inches as staple fibres and those measured in yards as filament fibres as demonstrated in Figure 2-3. Silk is the only naturally occurring filament fibre with usual lengths of 1,600 yards, while all other natural fibres are classed as staple [4,6]. In comparison, all manufactured fibres are produced as filaments and later cut or broken to produce staple fibres. Several manufactured fibres are maintained as filaments to produce specific properties in the end use application, for example spandex in the use of apparels is one such filament fibre example.

![Figure 2-3: Filament and staple fibres](image)

Figure 2-3: Filament and staple fibres
Fibre shape or cross section is also responsible for controlling various physical properties of the final textile [6,7] (Figure 2-4). The shapes of several different fibres viewed with the naked eye will all appear extremely similar. However, when viewed under magnification the differences in shape can be extensive. It is this difference in fibre shape and the surface construction that has a direct effect on the bulk or texture of a textile when woven or knitted [4,6,7].

![Figure 2-4: Representation of variation in fibre cross section, adapted from [4]](image)

Fibre surface conditions also have a direct effect on the properties of textiles [6,7]. The surface of fibres can be smooth, rough, slightly grooved, deeply channelled or wrinkled [4,6,7].

The longitudinal configuration of the fibre also has a direct effect on the properties of a manufactured textile. Fibres vary enormously longitudinally; they may be twisted, straight, coiled or crimped. The inclusion of these variances will affect resiliency, elasticity and abrasion resistance [6,7]. Crimped or crimp refers to the bends and twists along the length of the fibre. Where greater crimp is achieved the derived textile will have increased resiliency, bulk, warmth, elongation, absorbency and skin comfort [4,6].

The skin comfort results from the fibre’s ability to stand off the skin so a fabric will not cling to the wearer which produces a cold sensation. Crimp also generates the ability for a fibre to withstand being bent back on itself many times without breaking. This property is only inherent in natural wool fibres and is responsible for most of wool’s desirable
properties. Although crimp is not a natural feature of any manufactured fibres, it can be added after the extrusion and hardening process by heat setting in a crimped configuration [4].

2.4.4 Microfibres

Microfibres, also known as micro-denier fibres, are manufactured fibres that have a much finer and decreased diameter than conventional fibres [6,7]. Not all materials are suitable for this type of manufacture. Successful microfibre materials include Acrylic, Polyester and Rayon. Invented in Japan during the 1980’s [4,6], microfibres are produced using two main manufacturing techniques. The first involves normal extrusion from a spinnerette or die. During exit of the die the extruded fibres are drawn off at an accelerated rate, thus reducing the diameter. With careful control, these processes can produce filament fibres with a diameter of 10 microns [6]. The second method involves the use of two chemically incompatible polymers carefully mixed before being subjected to the same extrusion process. After extrusion, the two incompatible polymers are encouraged to split in a finishing process, producing up to four microfibres, as demonstrated in Figure 2-5.

Figure 2-5: Microfibre manufacture

The physical and chemical properties of fibres in microform have exactly the same behaviour as would be found in a normal size fibre configuration, except that certain characteristic are improved in the micro format [6]. A fibre in micro format will have
increased drape qualities due to improved flexibility; a softer texture as a result of its reduced diameter and the moisture transportation effect can be significantly improved, in some cases by up to 50% [4].

2.4.5 Nanofibres
Nanofibres are defined as fibres that have a diameter of less than 100 nanometres [8]. As with the fibres in the microfibre format, the properties of fibres can be improved upon in the nano format. In contrast to microfibre production, nanofibres are manufactured using electrospinning [8]. Through electrospinning, nanofibres can be created from a variety of materials including polymers, composites, and ceramics. Potential applications of nanofibres include medical, industrial, high tech, fuel cell and information technology [8].

2.5 Textile manufacturing techniques
2.5.1 Weaving
The exact date and origin of weaving is unknown and can only be speculated upon by historians [9]. However, the dawn of woven textile manufacture as it is recognised today is believed to have begun with the invention of the loom at some point during the height of the Egyptian civilisation. This simple device enabled primitive woven cloth to be produced for the construction of apparels and the like. During its conception this manufacturing technique was seen as an art or craft, where extremely skilled operators could produce finer, more detailed textiles in shorter times, which would command a higher price. These very first looms consisted of simple wooden frames that would enable warp yarns to be placed in a parallel adjacent series running vertically; the operator, using a shuttle tied to a weft yarn, would carefully weave the weft yarn horizontally over and under adjacent warp yarns and back creating the textile. The structure of a woven textile is demonstrated in Figure 2-6.
Modern weaving technology is still based upon this primitive technology and developments through the thousands of years since the conception of the loom have concentrated mainly on the speed of the process and the complexity of the weave achievable [4,11,12].

The woven textile industry received a huge injection of innovation during the British industrial revolution (1730 – 1900) where great steps were taken to automate the entire process of textile manufacture. In fact, the start of the industrial revolution was attributed to the invention of the flying shuttle (loom improvement) in 1733 [10]. During this period of innovation, the design of the loom received particular attention with ever more complex configurations being produced [4,12]. A major innovation was the development of the Jacquard loom. The Jacquard loom enabled the warp yarns of textiles to be controlled and moved independently. By moving the warp yarns the weaver could determine whether the yarn would appear at the front or back of the textile, altering its weave structure.

The Jacquard loom was the first machine to use punch cards to control a sequence of operations and led to the idea of using a stored program. This idea is considered to be an important step in the history of computer hardware [10]. The punched cards related to the position of any individual warp yarn within the textile. A hole in the card would signify that the warp yarn should appear at the front of the textile while a blank would signify the yarn appearing at the back. The ability to change the pattern of the loom's
weave by simply changing cards was an important conceptual precursor to the development of computer programming [10].

This capability of moving the individual warp yarns independently also enabled the creation of increasingly complex weave patterns. These new configurations were not only intended to produce new textile aesthetic properties, but also new physical properties. It was quickly established that altering the warp and weft configurations in a woven fabric would ultimately affect the physical and aesthetic properties and this realisation in turn lead to the creation of several new textiles. The development of these new textiles was only made possible by the automation of the weaving process and the invention of the Jacquard head, introducing consistency and repeatability previously unachievable [4]. This automation of the woven textile industry also introduced much greater speed in the production process, enabling greater productivity, thus giving birth to the first factories and the development of the textile industry as seen today [4].

Recent innovations within the woven textile industry relate to the development of the loom, where speed, automation and the integration of computers are the main aims. However, it is not the intention of this thesis to document all the innovations within the woven textile industry, nor is it the intention to provide a summary of the entire woven industry with regard to weaves, textiles and their specific applications. Rather it is appropriate to document weaving as a highly developed textile manufacturing technique steeped in history and that has the capability of producing woven textiles from any currently manufactured yarn. Further information is available through published works [4,11,12].

2.5.2 Innovation in the woven textile industry
Recent innovations in the woven textile industry include the development of three-dimensional (3D) weaving [11,12,13]. The development of this technique was not through the traditional textile weaving industry, but the preform composites textile industry, where preform textile structures are used for the manufacture of textile-reinforced composites [13]. In order to manufacture a complex 3D composite structure
with uniform mechanical properties, the textile based reinforcement must also be 3D and incorporate uniform properties [13]. Initially 3D textile reinforcements were manufactured from flat woven textiles and stitched into the desired complex 3D shape. This method however develops discrete boundaries within the textile preform structure, where the interface or seam between the individual textile sheets contributes to non-uniform mechanical properties in the eventual composite structure. To address this problem a complex 3D weaving process was developed for the creation of net shape textile preforms [13] as demonstrated in Figure 2-7. The 3D weaving process is based on a modified Jacquard loom, controlling the warp and weft yarns to create a 3D structure. However, this new emerging technology is only being utilised within the composites industry, and due to the complexity of the process, no commercial garment orientated system has yet been established [13].

Figure 2-7: Woven net shape textile [14]

2.5.3 Knitting
As for woven textiles, the exact origin of knitting is unknown and remains the speculation of historians. However, the first dedicated knitting machine documented was developed in 1589 and marked the beginning of the knitted textile industry [15]. In a similar manner to the weaving industry, knitting made huge technological advances during the industrial revolution. During this period the knitting machine was redesigned producing the first
Knitting is another process for creating textiles from yarn. This can be achieved from a single yarn using a weft knitting technique or the use of several yarns using a warp knitting technique. Both techniques rely on the principle of creating loops in the yarn that are passed through one another (Figure 2-8). Weft knitted products can be achieved by hand knitting or machine knitting, where warp knitting is mainly achieved by machine knitting.

Knitted textiles exhibit very different mechanical properties to woven textiles. Through knitting, the yarns of the textile are in continuous loops meaning no straight lines of yarn are achieved. This looping generates elasticity within the textile in all directions, whereas a woven textile only achieves elasticity in the bias or diagonal direction to the fibres of the textile.

Progress since the industrial revolution has been made in the number of materials that can be used in a knitting process, the knit configurations, speed, and applications of knitted products. The knitting industry is now comparable with the woven market as a commonly used method of textile production [4,15].
Again, it is not the intention of this thesis to document all aspects of the knitting industry, relating to: textiles produced, knit types, knitting machines and applications of knitted textiles. It is however; appropriate to document knitting as a highly developed textile manufacturing technique that can create elasticity in textiles made from fibres that do not exhibit elastic properties. Further information is available through published works [4,15,16,17].

2.5.4 Innovation in the Knitting textile industry

The most recent developments in the knitting industry have been seamless garments. During the last centuries of knit wear production, seamless technology has been primarily used for the manufacture of socks, gloves and hosiery [17,18], now seamless garments have entered all aspects of the clothing industry and the technique for creating seamless garments is relatively simple [18]. A circular knitting machine creates the garment to the contoured shape of the body instead of producing flat or contoured section that must be cut and sewn later completing the garment [15]. Given that this process is able to produce complete garments straight from yarn, manufacturers have managed to eliminate several steps from the manufacturing process, including any sewing or additional cutting and knitting. The types of yarns that are currently used in the production of seamless apparels are micro fibre yarns, Lycra and cotton. Applications of these yarns have enabled seamless garment production to compete with the more traditional woven apparel industry and produce softer skin like garments [15,17]

Other interesting innovations include the WHOLEGARMENT technology [18,19]. WHOLEGARMENT technology is based on flat bed knitting. In contrast to circular knitting techniques, the modified flat bed knitting apparatus has the capability of producing a completed garment in one manufacturing process. This means that complete garments can be manufactured that require minimal post processing. Examples of garments manufactured using this technique can be found at the process website [19].
2.5.5 Non woven

The most recent innovation in textile manufacture is the advent of non-woven textiles, first developed in 1942 [4,5]. The non-woven industry showed slow growth until the 1960s and since that period has developed rapidly [4,5]. Non-woven textiles are defined as textile materials made directly from fibres and held together as a textile by adhesion [4].

![Figure 2-9: Representation of non woven structure](image)

The adhesion may take the form of heat fusion if thermoplastic fibres are used or entanglement, if non-thermoplastic fibres are used. Generally these materials are flat, flexible, porous sheet structures with high surface to weight ratios. Non-woven textiles can also be engineered to include a wide range of physical properties by incorporating different fibre materials depending on the application, which may include various industries from construction to agriculture [5]. Examples of non-woven textiles include, toilet paper, cleaning wipes and geological membranes.

2.6 High performance textiles

Increasingly, functional or high performance textiles are seen as relevant research areas [20,21], where the introduction of innovative technology in contemporary textiles and advances in material science have produced high performance specialised textiles capable of protecting against ballistic attack, for example Kevlar™ [22]. Others can be self cleaning and antibacterial, control and transport moisture from the body, for example Gore-Tex™ [23]. Yet more provide thermal insulation and many other functionalities. The research surrounding high performance textiles relates to the
materials that can be utilised in fibre production, the weave or structure of the resulting textiles and potential coatings and treatments to enhance functionality.

In general, high performance textiles used in the manufacture of garments can be categorised into four main areas:

**Moisture transport and comfort cooling**
The main mechanisms here are 'push-pull' textile configurations where hydrophilic and hydrophobic fibres are used in conjunction. Through the matrix of the weave, the hydrophobic fibres are placed next to the skin and do not absorb moisture. Instead, the fibres push the moisture through the textile to the hydrophilic fibre, which absorb and evaporate the moisture. Therefore, the hydrophobic fibres push and the hydrophilic fibres pull creating a push-pull textile [24].

**Waterproof and breathable**
Waterproof textiles can be created using several techniques and fibre arrangements [25,26]. The ability of textiles to be waterproof can be characterised by the degree at which they act: water resistant, highly water resistant and highly water repellent. Mechanisms that achieve these properties consist of regular waterproof coated textiles, micro denier or micro fibre textiles and laminated textiles using a breathable membrane such as Gore-Tex™.

Breathable textiles work using the same mechanism as push-pull textiles, although in reverse; air can be transported through the textile, while moisture can be transported in the opposite direction.

**Thermal insulation**
High performance insulating textiles are generally based on textiles systems combining dead air entrapment, moisture dissipation and resilience, where the resilience of the textiles is the ability not to mat down (become thin) after prolonged use. More technical methods of thermal insulation include the use of specifically manufactured fibres
incorporating a hollow core. The hollow core of the fibre is used to trap dead air, therefore increasing the insulating properties. Examples include Thermastat™.

Textiles and systems that provide special features
Progress in the development of high performance textiles has resulted in products and clothing that offer special and unique properties, either through the addition of a topical finish to the textile [27] or the addition of a broad spectrum agent during actual fibre manufacture [28]. One example is the introduction of antibacterial properties [29,30,31].

Genetically modified materials are another example of current high performance textiles and include the ‘Spider silk’ being researched at Shinshu University [41]. The research is concerned with the introduction of spider genes into the chromosomal DNA of silk worms. The research aims are for genetically modified silkworms to create silk with the strength of spider webs. Similar research has been by undertaken by Nexia [32] for the production of their spider-silk material BioSteel ®. The Nexia program is however based on transgenic goats, were spider genes are introduced to the chromosomal DNA of the goats so their milk can be harvested and the BioSteel ® then created. More recent research as resulted in the generation of spider silk in laboratory conditions [33]. While the process still uses spider genes, this research utilise a genetically engineered insect-infecting virus which is grown in a culture of insect cells to create the spider silk proteins which can then be processed [33].

2.6.1 High performance clothing
High performance or functional clothing is a term given to clothing that performs to a set of specified criteria, for example an item that is waterproof, windproof and breathable. To achieve all of these performance characteristics in a single garment, high performance clothing includes several different functional textiles in a laminated textile structure, as demonstrated in Figure 2-10. Each layer of the laminated textile structure therefore delivers a specific functionality creating the overall high performance garment. Examples include Sympatex ® [34], and ActiveLayer from DuPont™ [35].
The manufacture of high performance clothing is therefore a sequential process starting with the manufacture of the individual layers. Once completed the individual functional layers are then combined to create the laminated high performance textile that performs to the specified criteria. Once this has been achieved the resulting laminated textile is then ready to be used in the garment construction process. Due to the laminated structure of high performance textiles, clothing manufactured using such textiles are created using the cut and sewn method of construction from flat sections of high performance textiles.

2.6.2 Other applications of functional textiles
In addition to the use of high performance textiles for apparel manufacture, their desirable properties have been utilised in other sectors [4]:

- Construction: Inflatable buildings for sport, domes and stadiums, bridges
• Military and space applications: Spacesuits, heat shields, protective clothing for biological warfare
• Filtration: Hot gases, liquids and air filters
• Safety: Anti-inflammable uniforms that protect fire fighters, antistatic materials, bullet proof vests, parachutes
• Medical: Artificial arteries, special bandages
• Transportation: Car brake linings and airbags, boat sails, rigid composite structures for air craft components

2.7 Smart textiles

In contrast to functional or high performance textiles, smart textiles were derived from a concept first defined in 1990 concerning intelligent or smart materials [36,37]. Here the emphasis is not on the measurable functionality or performance but the ability to sense stimuli from the environment and the inherent capability to react and adapt to them by the integration of functionalities within the textile structure. The stimulus or response of the textile can have an electrical, thermal, chemical, magnetic, or other origin [37]. The principle of smart textiles is that two components are included within the textile structure: a sensor and an actuator, with the additional possibility of complementing them with a processing unit. These three components therefore provide the textile with the ability to sense, think and act. Classification of smart textiles falls into three distinct categories:

• **Passive smart textiles**: [38] The textile can only sense the environment rather than adapt and react. Fundamentally, they are wearable sensors.
• **Active smart**: [39] The textile can sense the environment and react to it. The textile is not only a sensor but incorporates an additional actuation function
• **Very smart textiles**: [40] The textile can sense and act; in addition the textile can adapt its behaviour to particular circumstances.

At the forefront of the smart textile research is the creation of electrically conductive textiles or 'electronic textiles', where Electromagnetic Interference (EMI) shielding and
static dissipation are examples of commercially available products. Here the emphasis is the ability for textiles to act as potential circuitry. The conductivity of the textile can be generated in several ways, through the material used in the manufacture of the fibres, the inclusion of a conductive material in the manufacture of the yarns, or the coating or post processing of a finished textile article [41].

Another area of interest in the field of smart textiles include Shape Memory polymers (SMP) [41]. With SMPs the intelligence is created during the phase change of the fibre base material. DiAPLEX™ [41,42] is one example. The DiAPLEX™ fibre utilises shape memory polyurethane and the properties exhibited by the material above and below the materials glass transition to become more breathable at higher temperatures and to become insulating at lower temperatures, as demonstrated in Figure 2-11.

![Figure 2-11: DiAPLEX™ (left) below the glass transition acting as insulation, (right) above the glass transition acting as breathable](image)

2.7.1 Smart clothing

Almost all the current research concerning smart textiles is aimed at the creation of smart clothing [37,43,44,45], with the driving factor being society demanding the integration of intelligence into our local and daily environment. Clothing therefore provides the perfect vehicle for the integration of such intelligence, as it is an integral part of our personal environment, being adjacent to the body and worn almost anywhere.
and at anytime. It therefore offers a very unobtrusive and natural interface between humans and potentially integrated desirable technology.

Through the utilisation of such textiles that incorporate conductive fibres and the additional inclusion of standardised circuitry components, the vision becomes electronic clothing that integrates such items as the mobile phone and Personal Digital Assistants (PDA), giving initial realisation to the goal of truly wearable computers and the first step towards truly smart clothing [37].

The first generation of smart clothing was the incorporation of existing technology into conventionally manufactured textile garments. Examples include the ICD + line at the end of the 1990s, shown in Figure 2-12, which was the result of co-operation between Levi's and Philips [43]. The garment's textile architecture was adapted in such a way that the existing technology could be concealed within it. The concealed technology included a microphone, earphone, remote control, mobile phone and an MP3 player.

Figure 2-12: Levis & Phillips smart jacket

However, due to the traditional manufacturing techniques employed, all of the electrical components had to be removed prior to cleaning the garment.

One of the most advanced and well documented examples of smart clothing is the smart shirt developed at Georgia Tech [46,47,48] demonstrated in Figure 2-13. The smart shirt is made of plastic optical fibres and conductive fibres woven into a textile. These optical
and conductive fibres allow the shirt to wirelessly communicate with other devices, transferring data from the sensors embedded in the shirt.

Figure 2-13: Georgia Tech smart shirt

The smart shirt was originally designed for soldiers in combat, so that medical personnel could find the exact location of a bullet wound remotely. To pinpoint the location of bullet penetration, a light signal is continually sent from one end of the optical fibre to a receiver on the other end. The fibre is also connected to a personal status monitor worn on the hip. If the light from the emitter does not reach the receiver inside the monitor, this signals that the soldier has been shot. The light signal then bounces back to the point of penetration, which helps doctors find the exact location of the bullet wound.

Wearers of the device attach sensors to their body, put the shirt on and connect the sensors to the shirt. The shirt also tracks vital signs, such as heart rate, body temperature and respiration rate. These measurements are monitored in two ways: through the sensors integrated into the shirt and the sensors on the wearer’s body, both of which are connected to the monitor on the hip. With the capability to monitor these vital signs, the shirt is being marketed as a way to prevent sudden infant death syndrome (SIDS). Other applications may involve athletes, who may be interested in tracking their body’s performance during training and competition.
Other examples of smart garments include WEALTHY, developed by SMARTEX of Italy [49]. This wearable monitoring system fully exploits the possibility of textiles with strain fabric sensors, piezoresistive yarns and fabric electrodes. The system is capable of recording vital signs such as heart and respiratory rates and the activity pattern of the wearer. The garment works in collaboration with an external processing unit which captures and processes the data.

The Lifeshirt™ system [50] is another example. Again, this system is capable of measuring heart and respiratory rates in addition to the activity and posture of the wearer [51]. The complete Lifeshirt™ system consists of three components, the shirt, a data recorder and a PC based analysis software.

However, more commercially orientated and less technologically advanced examples have also been produced recently. A collaboration between Apple and Nike has seen the introduction of the ipod+trainers system [52], and a collaboration of Adidas and POLAR, has resulting in the adiStar fusion apparel [53]. The adiStar example uses heart rate sensors in the construction of the garment and a detachable transmitter. Once the transmitter is attached to the garment, electrocardiogram readings of the wearer are recorded during exercise. The data stored can then be later downloaded from the transmitter and used as training data.

2.7.2 Limitations in the manufacture of smart clothing
While the research into smart textiles is evolving at an accelerated pace with the creation of ever more intelligent materials, the research and examples of smart clothing are not. While it may be possible to create smart textiles, or textiles that have the capability to be smart, the creation of smart clothing is still a difficult operation and this is reflected by the lack of current smart clothing examples.

The current manufacture of the limited number of electronic smart garments requires a complicated manufacturing process chain starting with the manufacture of the textile itself. As with any other product, textiles have to be constructed with the manufacturing
process in mind and therefore are fundamentally limited by the need to consider Design for Manufacture & Assembly (DFMA) criteria. If this point were extrapolated further to smart conformal textiles articles such as "cut and sewn" clothing constructed from flat sections of smart textiles, then the whole process would suffer from DFMA criteria twice, once during the manufacture of the textile itself, and secondly, through the manufacture of the garment from several flat smart textile sheets. Having created a smart garment the final stages would involve the integration of circuitry components providing the smart functionality to complete the garment. Even considering this simplified manufacturing process chain, it becomes obvious why the number of electronic smart garments is limited and the result is often cumbersome and uncomfortable [5,37,45].

2.8 Summary of textiles

The literature shows that the applications of textiles are extremely vast, with the key to this diversity of applications being the inherent capability of textiles to provide form, function and aesthetics for a variety of simple and specialised needs. The structure of textiles is hierarchical, with the base component of the hierarchy being the fibre which ultimately determines the properties of the textile. The range of achievable properties that textiles deliver is not only due to the material used in the manufacture of the fibre but also to its scale, geometric complexity and the method in which the textile is created from it.

The micro level design of the fibres can have a dramatic effect on the macro level properties of any textile derived from it. An example is the geometric manipulation of polymeric fibres through crimping. Through the considered design of the fibre and its resulting geometric complexity, the favourable properties of a wool textile derived from natural fibres can be mimicked. These properties can be further enhanced by the reduction in scale of the modified polymeric fibre.

The literature also shows that 3D conformal textiles can be readily created. Automated knitting machines have had the capability of producing seamless conformal 3D textiles
since their conception, but weaving, traditionally a flat sheet textile manufacturing technique, can now produce net shape woven textiles, albeit for limited specialised applications.

The limitations of textile manufacture only become obvious when considering the manufacture of high performance and smart textiles. While it may be possible to create various properties in a textile through careful choice and micro level design of the fibre, if a series of different functionalities are required, for example for a textile to be both breathable and windproof, the textile manufacturing industry’s solution is the creation of laminated textile structures. This highlights the limitation of conventional textile manufacturing techniques and their capability of only being able to produce one complexity of textile and therefore a singular high performance functionality within a textile in one manufacturing process. In addition, laminated textile structures can currently only be produced in 2D flat sheets, meaning that the manufacture of high performance conformal garments can only be achieved through cut and sewn operations and no true 3D conformal high performance garments currently exist. This limitation of current textile manufacturing techniques is further reinforced when considering smart textiles and smart clothing.

The literature has shown that smart textiles and smart clothing are current research topics where there is an increasing demand for such items. The lack of smart clothing examples compared with concepts of possible smart clothing applications highlights the limitation of conventional textile manufacturing techniques to deliver such products. Where intelligence is required within the textile structure, this can currently only be achieved through the lamination of textiles and the inclusion of electrical components, such as micro processors. Again, due to limitations and the complicated manufacturing process chain required for their manufacture, no conformal smart garments currently exist and the limited number that are available are often cumbersome and ill-fitting.

Fundamentally textiles suffer from DFMA constraints, as a result of their hierarchical structure. Micro level design of fibres can produce a vast array of potential properties.
However, to physically create a textile exhibiting such properties, the fibre must be capable of being processed into a flexible yarn that must then have the capability of being woven or knitted into a textile. This sequential manufacturing process therefore means that the finished article suffers from two sets of DFMA constraints, limiting the complexity achievable in the hierarchical structure of textiles and therefore any potential functionality derived from it.

2.8.1 Final comments on textiles
If an additive approach is taken for the manufacture of high performance textiles, then the current DFM limitations surrounding their manufacture can be broadly solved. This elegant solution to the current paradigm of high performance textiles indicates that the potential for RM textiles is considerable, albeit with a significant amount of required research.

The literature review of textiles has developed a potential application for RM textiles: the manufacture of 3D conformal high performance textiles. However, certain issues remain that require addressing before RM textiles can be fully validated as a new textile manufacturing technique. These issues relate to manufacturing process itself, where complexity of geometry, build materials and geometric resolution will be the important factors. Therefore the next topic of investigation in this thesis is the area of Rapid Prototyping and Rapid Manufacturing.
Chapter 3: Rapid Prototyping and Rapid Manufacturing

3.1 Definition
Rapid Prototyping (RP) is a generic term for a classification of technologies that enable the manufacture of prototype components directly from a three-dimensional (3D) Computer Aided Design (CAD) data without the application of conventional tooling [54,55].

3.1.1 Background and application
Commercially available RP systems, also known as Additive Manufacturing (AM), Solid Freeform Fabrication (SFF) and Layered Manufacturing (LM), began to appear in the USA during the late 1980's [54,55,56]. Some 20 years later, the US remains the dominant player in the global market of RP systems [55,56] but alternative systems have been developed worldwide, notably in Japan, Europe, Russia, China and more recently Israel [55,56].

To many organisations new to RP technology, the exact nature of the intended use of the technology may remain unclear. Currently RP technology has been utilised for the manufacture of concept models, enabling the greater communication of design ideas and verifying design concepts. The technology is also used for the production of functional or semi functional components when creating complex working prototypes, the manufacture of master patterns for the manufacture of production tooling and direct tooling and, increasingly, the manufacture of end use parts [55,56].

3.2 Rapid Prototyping
Conventional manufacturing processes are generally subtractive or formative in nature, where the manufacturing element of the production process refers to the removal or moulding of material when creating the desired product geometry [56]. Conventional
milling and turning are examples of subtractive manufacturing techniques, while casting and injection moulding are examples of formative manufacturing. By contrast, RP technologies are additive manufacturing techniques [54,55,56]. RP geometries are built up gradually in layers until the final geometry is realised. The methods by which these layers are produced, and the materials in which the components are built, vary significantly between the different available processes within the single classification of technology [56].

The initial point for all RP processes is a 3D CAD model, prepared and exported to meet the input requirement for a given additive technology. Various digital inputs are available that can be used to create RP components; including Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) data [57]. However, the de facto standard for RP process data input remains the STL (Stereolithography, Standard Triangulation Language or Surface Tessellation Language) file format developed by 3D systems [57,58].

STL file creation and the implications of CAD in RP will be discussed in detail in later sections but for clarity, an STL file is a tessellated triangular 3D approximation of a 3D geometry. Having created the STL file input from the 3D CAD model, discrete 2D slices are taken that correspond directly to the build slices that will be used in the layer by layer manufacture. Reproducing the discrete 2D slices of the 3D CAD data using the additive technique therefore reproduces a physical representation of the 3D CAD model. This can be more accurately described in a pictorial step by step progression as follows in Figure 3-1:

**Stage one:** Creation of desired geometry in 3D CAD  
**Stage two:** Conversion of CAD data into STL data and exported to the RP system  
**Stage three:** Sliced version of the STL file, created by RP system  
**Stage four:** Geometry built by RP process
Stage four of the sequence highlights the issue of stair stepping common to all RP system. While exaggerated in this example, RP systems build parts through consecutive layers creating the stair stepping effect shown. This issue can be reduced with the reduction of the layer thickness used when manufacturing the desired geometry.

Stage one: Geometry creation
Stage two: STL format creation
Stage three: Slices taken from STL
Stage four: Additive Manufactured geometry

Figure 3-1: Methodology of additive manufacturing
3.2.1 RP process classifications

Current RP technologies that are commercially available fall under the following five categories:

- Curing processes: where a photosensitive polymer under controlled conditions can be exposed to a light source capable of curing the polymer. Example, Stereolithography (SLA).
- Sheet processes: the utilisation of sheet material which can be cut to shape and stacked upon each other. Example, Laminate Object Manufacture (LOM).
- Dispensing processes: where a material may be heated until melting point and deposited either as a hot filament or individual droplets of the material. Example, Fused Deposition Modelling (FDM).
- Binding processes: the utilisation of a liquid binder, usually adhesive, can be deposited upon a powdered material, creating a solid. Example, 3D Printing (3DP)
- Sintering & Melting processes: where a powdered granular material can be sintered or melted together using a heat source (typically a laser). Example, Selective Laser Sintering (SLS).

The intention of this section is not to provide a critique of all the available systems but to introduce the technology and summarise the main manufacturing processes utilised in RP. Further information relating to additive manufacturing can be found in various published works [55,56,59].

3.2.2 RP Resolution

The resolution of RP geometries, or the minimum feature size which can be produced, is dependent not only on the process used for manufacture but the build material utilised. The resolution can also vary depending on the orientation of the part within the build envelope. For example, when using Stereolithography, the minimum possible increment in the z axis direction is approximately 25 μm [56]. However, due to the diameter of the laser spot, and energy leakage into the resin surrounding the laser spot, the resolution in the x-y plane is substantial less, with the minimum feature size being approximately...
100 μm. In general it can be stated that the minimum feature size achievable for most currently available, commercial systems, falls between 100 and 500 μm, although some specifically modified systems can show slightly better resolutions [60].

Whilst these resolutions were acceptable when RP was first developed, there has recently been increased interest in using these technologies for more advanced and demanding applications, which in turn often require better resolution and accuracy of geometries. Current problems with the use of RP systems to produce small components include lack of definition of features such as small holes, which may have to be re-drilled after the part has been built, and difficulties with manual surface finishing or removal of supports. These concerns, when coupled with the current manufacturing trend towards miniaturisation of consumer products, for example mobile telephones or MP3 players, have led to increasing amounts of research being undertaken in the area of RP resolution.

The best resolutions have been produced from purpose-built processes, based on the Stereolithography (SLA) system. Often, these systems have replaced the laser with some form of masking system to allow instantaneous irradiation of the entire build area [61]. This dramatically reduces the build time for parts, and also provides much better resolution compared with the use of a conventional laser based system. Minimum feature sizes using modified SLA system have been recorded as small as 1 μm [62].

Figure 3-2 shows a scale model of a car, produced on one of these systems. The model was built in 5 μm layers, and is approximately 3 mm long. Even at this minute scale, a substantial amount of geometric detail can be observed on the part.
Figure 3-2: Scale model of a small car, made of 673 layers of 5 μm each [61]

Although these major improvements in the achievable resolution of parts are potentially extremely useful for many manufacturing industries, there are various limitations to be overcome before these resolutions are achievable practically. At such small scales, the removal of any support material from a part can become extremely difficult, if not impossible, especially for fragile or fine-featured parts [63]. Many small systems often rely on the use of gravity effects to level out the surface of the resin before curing a cross-section, which necessarily increases the time of the build process [64]. As an example, the model shown in Figure 3-2 required a build time of three hours.

However, as further research into this area is undertaken, it is likely that the resolution of RP parts will continue to improve, and micro or even nano-scale Rapid Prototyping may become a genuine manufacturing reality.

3.2.3 Rapid Prototyping advantages and limitations

The advantage of utilising Rapid Prototyping in the design and development of new products is well documented. It can reduce lead times to market, enable the production of increased design iterations and reduce development cost due to the lack of required tooling. However, the principal advantage of RP is the ability to manufacture parts of virtually any complexity of geometry entirely without the need for tooling [55,56]
Currently available RP systems are capable of producing components from a range of materials. These materials include: metals, plastic, paper, cellulose, ceramics and even water [55,56]. However, components produced by Rapid Prototyping techniques have inherently very different mechanical properties to those produced by more conventional manufacturing techniques such as computer numerical controlled (CNC) machining, casting techniques or injection moulding. The comparison of these properties from components manufactured through RP technologies depends on the materials and process used for manufacture, but may include reduced mechanical properties, high levels of porosity and the possibility of a reduced component shelf life due to the continual degradation of the build material. Although the materials and process used in the manufacture of RP components varies between different systems within the classification of the technology, several key limitations apply to typically all RP systems:

- Surface roughness of the component and stair stepping
- Dimensional accuracy of the manufactured component
- Mechanical properties of the component
- Speed of manufacturing the component
- Build envelope of the RP process

The cited limitations of the technology have contributed to the lack of industries adopting RP as a true manufacturing process [56]. In order for RP technologies to develop into Rapid Manufacturing technologies, these limitations must be overcome. Continuing research is being undertaken globally in all the areas highlighted [55].

3.3 Rapid Manufacturing

Rapid Manufacturing (RM), as described by the Wohlers Report 2006, defines the term RM, as "the direct production of finished products or parts using additive fabrication techniques". It is widely believed that current RP technologies will develop into RM technologies when all of the current limitations of RP cited previously have been addressed and overcome, or new and true RM systems are developed that totally afford the definition of RM to the prospective user [55]. RM is widely anticipated throughout
manufacturing industries and the expected impact that RM has to offer is such that it has been heralded as the next industrial revolution or the digital revolution [59].

The benefits that RM should afford to manufacturing industries encompass all the advantages of current RP technologies and overcome those limitations that currently diminish from its attractiveness. With RM, tooling is eliminated, eradicating substantial time and costs associated in their production and manufacture. Other advantages include design freedom (discussed in detail later); no longer will the constraints of manufacture and the process of Design for Manufacture (DFM) be an issue. Heterogeneous or functionally graded materials, customised products, just-in-time production and the decentralisation of manufacture could all be realised through the application of RM [54,56].

Certain commercial avenues are being exploited through the utilisation of RP technologies in the production of end use parts, although this is not in itself RM, more the application of RP systems to low volume production and intermediate tooling operations. Examples of end use parts being manufactured from current RP technologies include components for the international space station and the entire space shuttle fleet, custom lamp designs, orthodontic aligners, custom fitting hearing aid shells, and air ducting components utilised in both formula one and aerospace industries [55].

In essence RM is predicted to change virtually all aspects of manufacturing. However, it is not just the method of manufacturing a particular component that will change, the actual products manufactured will have the potential be entirely different to those available commercially today [65,66].

3.3.1 Design potential of Rapid Manufacturing
The principle advantage to be gained by utilising a Rapid Manufacturing approach (including most but not all of the currently available RP techniques) is their ability to manufacture geometry of virtually any complexity entirely without the need for any tooling [54,55]. This need for tooling in conventional manufacturing represents one of
the most restrictive factors for today's product design and development. The absence of this requirement when utilising Rapid Manufacturing processes means that many of the restrictions of 'Design for Manufacture & Assembly' (DFMA) [67] that are essential in a modern manufacturing environment are no longer valid [68]. For example, with injection moulding, the need to consider the extraction of the part from the (usually expensive) tool takes an overriding precedence in the design process of the part. Therefore, designers have been educated to develop designs with manufacturing considerations in mind (DFMA); creating restricted geometries so that parts can be made easily. This therefore limits the original design intent of the product.

With Rapid Manufacturing, the design potential offered means that complexity of geometry will no longer be a limiting factor on the design intent. Therefore highly complex designs can be manufactured in lower volumes than would be required when utilising conventional tooling-based manufacturing techniques. The mass production of the design would be required to offset the associated tooling cost of such complex geometry. When utilising an additive manufacturing technique, the potential exists for the economical manufacture of one unit and therefore each part that is produced could in theory be customised [69]. Without the need for tooling or necessity to consider any form of conventional DFMA, the possibilities for design are only limited by the imagination of the designer [70].

The ability to produce any complexity of geometry that can be created within CAD packages means that designers can enter a new dimension of 'Manufacture for Design' rather than the more conventional 'Design for Manufacture' philosophy [71]. This freedom of design is one of the most important features of RM and extremely significant for producing parts with complex or customised geometries, which could result in reducing the lead-time and ultimately the overall manufacturing costs for such items. RM will affect manufacturers and customers alike. For manufacturers, costs will be dramatically reduced as no tooling is required. For customers, complex, individualised products will be produced cost-effectively and configured for personal use, thus giving the potential for much greater customer satisfaction [66]. The elimination of tooling and
the subsequent removal of many DFMA criteria will realise significant benefits in the design and manufacture of a product or components. Cited benefits will include [54]:

**Design complexity / optimisation**
An example of design complexity is the manufacture of surface macro textures [72]. Utilising additive manufacturing techniques, geometrically complex surface textures can be manufactured on products and components for a range of potential application where traditional tooling or subtractive manufacturing would prove impossible.

**Parts Consolidation**
Examples of part consolidation include the US navy using RM to produce end-use components for Trident submarines. Utilising additive manufacturing techniques it became possible to reduce one particular component from an assembly involving 25 separate pieces, to a single component manufactured in one RM process [73].

**Body-fitting Customisation**
An example of body-fitting customisation is the research undertaken at Loughborough University in collaboration with MG Rover. Individuals' body-scan data was used in conjunction with additive manufacturing to produce customised car seats with potentially increased levels of comfort [54].

**Multiple free moving assemblies manufactured as one**
An obvious example of this particular benefit is the manufacture of RM textile structures developed by FOC and the research contained in this thesis [1,70].
3.3.2 Rapid Manufacturing of Metals

Another rapidly-expanding area of RM is that of metals processing, with several companies focussing on the production of end-use metal components. EOS GmbH have developed the DirectPart process, for the production of prototypes, tooling, and end-use components [74]. Using the EOSINT M 270 machine, parts can be produced in a wide range of metals, including bronze, steel, stainless steel, cobalt chrome, and titanium. An example of the use of this is the production of personalised knee implants as demonstrated in Figure 3-3 [75].

![Figure 3-3: Cobalt chrome knee implant produced on EOSINT M 270 machine](image)

Another example, the ProMetal process [76], involves the printing of a liquid binder onto metal powders, including stainless steel, tool steel, bronze and gold, in order to produce porous parts, which are then infiltrated to near-full density.

US army mobile parts hospitals utilised to repair vehicles and equipment in the field, have made use of Optomec's LENS77 process to produce replacement parts for vehicles in Iraq [78], leading to huge reductions in spare parts inventories, and reductions in down-time for essential equipment. The process uses a high power laser to fuse metal powder into solid parts, and can currently process steels, titanium and nickel-based super-alloys.
Other applications for RM of metals include the laser sintering of gold for jewellery production [79], and the production of moulds and dies for a wide range of moulding and casting operations [80].

3.3.3 Rapid Manufacturing of functionally graded materials
Another of the more exciting prospects from the perceived advantages of RM relates to heterogeneous or functionally graded materials. Multiple materials within a component are not essentially new to current manufacturing techniques. Combinations of materials within a single product can be currently achieved through overmolding techniques used in injection molding [54]. However, when combining two materials utilising this technique, a discreet boundary exists between the two materials and the operation required to produce such a component is costly and time consuming [65]. In spite of these considerations, combining the properties of two different materials is very appealing and provides added functionality to a component. RM will greatly enhance this type of capability, providing the ability to grade multiple materials throughout the geometry of a component increasing both design and aesthetic possibilities with variable mechanical properties [55,65,73].

3.4 Direct-Write Technologies
A related classification of technology to both RP and RM is Direct Write. The term Direct Write can be used to classify a range of technologies capable of manufacturing 3D electronic circuitry on 3D components. Whilst these processes are generally not yet competitive for very small-scale components, they are expected to have several benefits in terms of speed and complexity [81], particularly when looking at the printing of 3D circuits as opposed to traditional 2D circuit boards.

Some success has been shown in the production of encased 3D circuitry by incorporating conventional RP/M techniques with the deposition of conductive fluids, which are then solidified during the build process [82]. Other processes have been developed using electron beam nanolithography [83], or by a combination of laser
processing and micromachining [84]. In addition to producing simple electronic circuits, such techniques have also proved successful for the production of conformal antennae [85], and in the production of batteries [86].

It is considered entirely possible that, with further research, it may be possible to produce fully functional assemblies [87] and systems [88] using combinations of Direct-Write technologies and RP/M techniques. One vision for the future is the production of a machine which is capable of producing both electronic and mechanical components in one process, for example to create robots incorporating power cells, actuators and sensors, without the requirement for multiple assembly operations or human intervention [89].

3.5 Summary of RP, RM

Rapid Prototyping has matured during the last decade to develop into a key engineering tool in the development process of product designs. Advances have been made in the mechanisms used, the build materials and the resolution of the processes to create a prototype manufacturing technique capable of creating geometry of unrivalled complexity and increasingly, in engineering grade materials. This principle advantage of geometric complexity without the requirement of tooling has led to the development of Rapid Manufacturing where the aim becomes the true additive manufacture of end use components.

However, while there are examples of RP processes being utilised in the manufacture of end use components, there are now true examples of RM. While improvements are still to be made in build materials and build envelopes in which components can be manufactured, end use metal products created from additive manufacturing processes are being utilised for specialist applications. However, functionally graded materials have yet to be developed to the required standard for engineering and consumer product applications.
It is widely accepted that future RM techniques will be developed from current and future RP processes. Therefore it follows that current RP systems can be used as a benchmark for the capabilities that RM will eventually offer and in particular, if they have the capabilities required to produce 3D conformal high performance textiles.

The literature shows that RP and therefore future RM systems do have the capability to produce the level of geometric complexity required by a high performance textile application. Even when considering the micro level, the literature shows that RP process resolution is a current topic of research and the level of geometric complexity achievable can be maintained at this smaller scale.

The literature also shows that build materials available for current RP processes are extremely varied and new introductions are being consistently added to the portfolio. While the build materials may not provide an exact match to those being currently used in the manufacture of textiles, the previous review highlighted the importance of micro level design and the dramatic affect this can have on the macro level properties of the textile. In addition, the materials currently used for the manufacture of fibres are a current constraint of the fibre manufacturing process, therefore RP and future RM systems enable the use of potentially new base materials for the manufacture of textiles.

3.5.1 Final comments on RM
The review of textiles highlighted a limitation in the current manufacture of high performance textiles that could be solved utilising an additive manufacturing approach. The review of RP and RM has shown that this is indeed true and additive manufacturing does have the inherent capabilities required for this application. The literature also shows that in order to manufacture any geometry by RP or RM then a 3D model of the target geometry is required. Therefore the next topic of investigation in this thesis is the area of CAD and CAD tools used within the conventional textile industry.
Chapter 4: Computer Aided Design

4.1 Definition

Computer Aided Design (CAD) can be defined as the use of computers and software to assist users in the creation of 2D and 3D design data [90].

4.2 Background and application

CAD software, or the ability to utilise computers in the design process was first conceived in the 1960s with the creation of a Computer Aided Drafting system, Sketchpad, developed at MIT [91]. However, CAD as, it is recognised today, was developed by the automotive and aeronautical engineering industries to assist the design and manufacture of their respective products. Since these early specialised applications, the use of CAD now ranges from the large scale road systems, buildings and bridges to mid scale cars, aircraft and shipping. At a smaller scale CAD has applications in mechanical components, textiles and consumer products and at the micro scale, miniature or micro components and the molecular modelling of materials. The applications of CAD have grown in accordance with computing power and availability, removing it from a supercomputer-only application to a software system that may be run on a home personal computer.

CAD was once thought of as a specialised drafting tool for engineering schematics, but now the area of CAD has evolved to incorporate several other applications of computer integration with engineering, manufacturing and simulation. These areas include [90]:

- **CADD** - Computer Aided Design and Drafting: the use of computers in the creation of product plans and schematics.
- **CAE** - Computer Aided Engineering: the use of computers to assist with all stages of engineering design work. Similar to computer aided design, but also involving the conceptual and analytical design steps.
• **CAM** - Computer Aided Manufacturing: the process of utilising computers to control and monitor tools and machinery in manufacturing.

• **DCC** - Digital Content Creation: the use of computers in the creation of digital content such as animations, media and renderings.

### 4.2.1 Development of CAD

The development of CAD since its conception in the 1960's has seen increasing functionality being provided to its users. Starting as a 2D drafting system, CAD systems developed into 3D, where users could fully visualise their design intent in a virtual environment. The development of 3D modelling started in the 1970's [92]. During this early period of 3D design work limitations were prevalent in the creation of 3D geometry and its rendering, making the creation of complex geometries extremely difficult. Model visualisation was restricted to wire frame representations, demonstrated in Figure 4-1a and due to the heavy computational requirements of such early 3D work, the applications of this 3D modelling were highly limited. The next evolution in 3D design work was the development of surface and solid modelling using Constructive Solid Geometry (CSG) and later Boundary Representation (B-Rep), demonstrated in Figure 4-1b.

CAD as it is recognised today was first introduced with the release of Pro-Engineer in 1987 which was the first system to incorporate a functional Graphical User Interface (GUI) and parametrics [93]. The introduction of the GUI greatly increased the speed and efficiency of solid modelling and marked a turning point in the design of CAD software and systems. With continued research into 3D curves, splines, NURBs (Non Uniform Rational B splines) surfaces, rendering, GUI's and the addition of increased computational power since this period, the capabilities of CAD have been transformed into an extremely powerful tool for designers and engineers alike. CAD now offers the capability of freeform surface modelling techniques and solid modelling operations that allow the user to create almost any complexity of geometry required and visualise that geometry in a photo realistic rendered virtual environment as demonstrated in Figure 4-1c.
4.2.2 Limitations in conventional CAD for RM

Interestingly, the development in CAD packages and GUI’s that enable the user to design components has not kept pace with that of developing RP and RM technologies. The current design of CAD systems and GUI’s reflect the mainstream manufacturing techniques that they have been created to support and not the potential capabilities of emerging manufacturing technologies. It follows therefore that currently it is possible to manufacture components by RP and RM that cannot be efficiently represented through conventional CAD systems. Klein bottles and Mobius surfaces are such examples [54].

Research utilising additive manufacturing techniques for the manufacture of product designs has cited several limitations when using conventional CAD systems [94]. The research followed a conventional design process where the starting point was the generation of concept design sketches of the product to be manufactured. The first limitation cited was the inability to translate the concepts held within the design sketches to 3D CAD data. The research suggested that translating design ideas through to 3D features in CAD was extremely time consuming and labour intense. This difficulty was attributed to CAD systems being expert systems that require extensive training and the complication of the GUI. It was also cited that certain features of the concept design sketches would have to be ignored altogether in the final manufactured design, which was attributed to the lack of design freedom within the CAD system used. Another issue
highlighted was the time taken to complete the CAD model in comparison to the time required for the actual manufacture of the product. Here the time taken to complete the CAD phase of the design process was far greater than that of the manufacturing phase. When considering a more traditional manufacturing process such as injection molding, the manufacture of the tool required for producing the actual component is the longest part of the entire manufacturing process. When considering RM as the manufacturing technique, the longest part of the manufacturing process becomes the translation of design concept to 3D data or more concisely the CAD phase of the process [94]. Importantly, CAD systems currently provide the only meaningful tool for the creation of 3D geometric data required by RP and RM processes for manufacturing. Due to the complexity of CAD software being expert systems, this has the effect of limiting those who can, and want to, utilise RM technologies for manufacturing [95].

With the advent of RM, CAD systems need to evolve to accommodate the developing capabilities this manufacturing technique presents, not only in the geometric complexity achievable through RM but with the potential introduction of functionally graded materials. CAD systems require the capability to model the complex interaction between different materials within a single structure and translate this data to RM systems for manufacture. Current research into this issue is looking at the potential of voxels (3D pixels) to model the interaction of different materials within a single structure [96]. However, this method of describing 3D geometry requires significantly more data than more conventional 3D representations [97]. Continued research in this field has led to the development of an experimental CAD system that enables the modelling of graded materials, Innerspace [97]. However, this system is still being developed and the functionality is limited.

4.2.3 The interface between CAD and RP and RM
As previously stated, to produce geometries via additive manufacturing firstly requires a 3D CAD model of the component. To enable the RP/RM system to read the data, the 3D data must be converted to an appropriate file format such as the STL format. Once converted, the RP system has the capability of reading and slicing the data ready for
manufacture. It can therefore be stated that RP systems never actually build the geometries created from the CAD system but a sliced version of a STL approximation [98,99].

An STL file is a polyhedral representation with triangular facets of the intended geometry for manufacture [57,98,99]. STL files are generated from the CAD model using a process known as tessellation which produces triangles to approximate the CAD model surface [98,99]. STL files can be created in two different versions, either ASCII or binary [98,99]. Binary STL files are simply line after line of binary information (ones and zeros) that describe the STL approximation of the CAD geometry. ASCII STL files in contrast produce the coordinates of the individual triangle facets; this increases the file size (Bytes) dramatically but enables the user to read the data. Within the STL file, triangular facets are described by a set of X, Y and Z coordinates for each of three vertices and a unit normal vector to indicate the side of the facet that resides outside the geometry [98].

Disadvantages of the STL conversions relate to the generation of high degrees of redundancy; this is due to the duplication of vertices and edges for every triangular facet. STL files usually require unnecessarily high computer storage space for complex geometries, as well as high computer resources to process. Furthermore, errors are prone in the tessellation process, resulting in defects such as holes or cracks, non-manifolds objects, overlapping facets and incorrect normals [99]. Checking for and the repairing of these defects can be time consuming and algorithms to facilitate the repair of these defects have been developed [99]. In addition, the STL format does not provide efficient information storage compared to the higher-level representations such as NURBS (Non Uniform Rational B Spline) surfaces. As a result, the STL file contains thousands of triangles to represent a model, whereas, using a more mathematically precise format, only a few spline or NURBS surfaces may be sufficient to represent the same model [98].

Due to these inefficiencies, alternative file formats have been researched to provide a more efficient interface between CAD packages and RP systems [99]. Existing file
formats that have utilised for this purpose include: Initial Graphical Exchange System (IGES), Stereolithography Contouring (SLC), Hewlett-Packard Graphical Language (HPGL), Virtual Reality Modelling Language (VRML) and Standard for the Exchange of Product Model Data (STEP) [98]. All of these formats have been utilised with relative success, providing a more efficient and mathematically perfect representation of the initial CAD model. However, due to mainstream RP and RM developer's lack of commitment to these formats, wide scale adoption has not been achieved. Other file formats have been specifically developed to combat the problems of inefficient approximation and larger file size, these include: Layer Manufacturing Interface (LMI), Common Layer Interface (CLI), Rapid Prototype Interface (RPI) and Picture format (PIC) [98].

The most recent development in this area, PIC format, offers a perfect representation of the desired geometry, increased accuracy from the manufactured part when compared with STL builds of the same component and reduced file size [98]. The recent research cited that for non-curved geometry the PIC file format offered very little advantage over STL conversions, when considering file size and shape approximation. However, when considering highly curved geometry, for example a sphere or torous, the file size reduction was sufficient enough to consider over STL file conversions [98].

4.3 Applications of CAD in the textile industry

CAD and the use of computers have been used in the development of textiles since the 1980s [4]. The first utilisation was CAD/CAM applications to replace Jacquard loom punch card data for the creation of woven textiles.

The adoption of CAD/CAM technology has increased the speed and accuracy of developing new textile products, reducing the manpower required to complete the development stage [100]. After its introduction for the first time, textile designers could create a new design relating to the actual weave or knit construction, visualise the design and then export the data to the relevant manufacturing process. These first
simple CAD/CAM packages reduced the design phase of a textile from a potential 2 weeks to as little as 24 hours [4]. More recently CAD had been used in the area of textile print design through automated textile ink jet printing systems [101]. Current developments in the area of CAD within the textile industry help build on the traditional CAD/CAM functions seen in past decades and offer a new way of visualising designs and product development [102].

4.3.1 Computer Aided Apparel Design (CAAD)
The development of Computer Aided Apparel design (CAAD) has seen less development than the CAM aspects of the textile industry. CAAD systems have concentrated on the visualisation of completed garments and the efficient communication of design concepts and not the actual design of the garment itself. A screen capture of Lectra CAAD software is demonstrated in Figure 4-2.

![Figure 4-2: Modaris 3D Fit, Lectra's 3D Virtual Prototyping and Visualisation software][103]

Due to the number of third party software producers and their respective software packages each with different capabilities, further information about their capabilities can be found on their respective websites [103, 104, 105, 106].

Areas in which current CAAD packages require further development falls under these main categories [107]:

---

[103]
• The 3 dimensional specification of the entire garment: Providing the designer with all the tools necessary to realise his design ideas as a fully rendered realistic 3D model

• Accurate simulation of garment: The capability to accurately simulate the characteristics of the garment incorporating all the seams and textile mechanical properties during wear

• The creation of 2 dimensional cutting patterns from the 3D data: Utilising the 3D models to extract all the information required for manufacture.

• Compatibility with current CAD/CAM systems: The compatibility with existing CAD/CAM systems being utilised in textile manufacturing.

The functionality required by textile design houses and apparel producers has led to investigations into other areas of CAD more associated with the game development and animation industries. The use of conventional engineering CAD has also been investigated, where current research is being undertaken into its use in apparel design [107,108].

4.3.2 DCC CAD

Digital Content Creation (DCC) software is a powerful blend of three different applications [109,110]:

• 3D modelling: DCC offers complex freeform surfacing and solid modelling techniques for the creation of organic and complex structures and characters that are only matched by the current very high-end conventional engineering CAD systems.

• Animation: DCC systems have the capability to create life like movement and motion of complex 3D geometry utilising real world dynamic physical properties, such as gravity.

• Rendering: DCC offers a host of features to make 3D data appear life like and photo realistic, including texture mapping and extensive scene lighting.
The primary use of current DDC systems is for the development and creation of computer games graphics, animated movies, and special effects used in film and television [109]. The 3D freeform modelling and photo realistic rendering capabilities of DCC software are demonstrated in Figure 4-3.

![Image](image-url)

Figure 4-3: Freeform modelling and photo-realistic rendering capabilities using 3D Studio Max [111]

Due to its use in these industries, DCC software has received the added advantage of increased funds for research and development to enhance its capabilities further, advantages that dedicated apparel software has not received due to its low demand and niche markets [109].

Recent innovations in DCC software relate to the realistic modelling of apparels and textiles structures by approximating the structure as 3D surface geometry [108,109,110]. In order to realistically model the physical behaviour of textiles and garments by a 3D surface approximation, a selection of variables are available within DCC packages, including: elasticity, friction, buoyancy, compressibility, damping, fold stiffness, spread angle and air resistance [108]. The capability to more accurately model, simulate, animate and render garments and textiles better than any CAAD package currently available provides an interesting paradigm to the apparel industry. If the close links and compatibility of DCC with current engineering CAD/CAM packages are considered, the incompatibility experienced between currently available CAAD systems and current
CAD/CAM solutions being utilised in textile manufacture becomes instantly solvable. Due to the potential benefits DDC offers, research is currently being undertaken into its use and potential other CAD systems for the application of apparel design [108].

4.3.3 CAD for textiles

Other textile based CAD packages include textile CAD systems that have been developed by the textile reinforced composites engineering sector. These systems have been developed for the efficient modelling of the actual 3D geometric structure of the textiles reinforcement for further analytical analysis [112, 113, 114, 115] as demonstrated in Figure 4-4. The 3D structure is required to accurately predict the mechanical behaviour of textile reinforced composites through Finite Element Analysis (FEA), where the 3D geometric structure of the textiles composite is required for meshing.

Figure 4-4: Textile weave structure

The modelling of a complex 3D textile structures can be extremely difficult using conventional CAD software [114], although such complex models can be achieved the results are often time consuming, error-prone and inefficient [115]. As the actual structure of the textile reinforcement becomes more complex, with increasingly complicated weave structures, the problems of utilising conventional CAD to generate the complex 3D structure become exacerbated [115]. To address this situation, research engineers in the field of composites have developed in-house textile CAD packages specifically for this application [112, 114, 115].
When considering the FE (Finite Element) analysis of complex 3D geometry, an often implemented practice is that of geometry simplification. This aids the computational efficiency of the process by reducing the resulting complexity of the FE mesh, which has a direct effect on computational time. As a result, the CAD systems designed for this purpose have been developed to provide an idealised geometric representation of the fibres within the textile reinforcement and not an exact representation. To achieve this idealisation of the geometry, current textile CAD packages represent the individual fibre structure of textiles reinforcement in a vector format. The volume or body of the fibre is then created by sweeping a cross sectional shape along the vector, as demonstrated in Figure 4-5. As the geometries of the fibres are being idealised in current textile CAD systems, the cross sections are currently limited and examples include circles, ellipses and tori [114].

Another common feature of textile CAD packages is the micromechanical concept [115] and the Representative Volumetric Element (RVE). The micromechanical concept used in the FE analysis of textile structures assumes that the overall mechanical properties of a textile structure can be predicted from the mechanical properties associated with a much smaller sample of the same material. The smaller sample is represented by the RVE of the textile. The RVE is the smallest possible tessellating volumetric element of the textile reinforcement and often the only part of the textile structure to be geometrically modelled, also demonstrated in Figure 4-4. Using this system it becomes possible to model the RVE and simply repeat the tessellating unit within the volume that
the final designated textile structure would occupy, producing the complete 3D geometry. Currently most of the textile CAD packages in operation are beta versions used primarily by the people who designed and programmed them. Due to the sequential application of FEA, the systems are designed to export the 3D geometry into various CAD formats making them suitable for potential other applications. Commercially available systems are virtually non-existent although much research is being undertaken in this area [112,113,114,115].

4.4 Summary of CAD

Since its first conception, CAD has developed into an extremely powerful tool to assist in the creation of design data for a magnitude of different applications. The current capabilities of CAD allow the creation of photo-realistic 3D design data to be visualised fully in a virtual 3D environment for the validation of design intent. Originally thought of as computer aided drafting for the creation of engineering schematics, the modern era of CAD has evolved to incorporate several other computer based applications in the design and manufacture process of products and components, including CAM and CAE.

The literature has shown the application of CAD in the textile industry to be varied, where the general trend shows the textile industry in all its forms is moving away from traditional techniques and embracing the digital age. Whether this may be in the form of a simple raster CAD/CAM program used to drive weaving looms and knitting machines, or computer aided apparel design and manufacture systems. The desire by the textile industry to utilise CAD techniques in the design and manufacture of its products has not been matched by the CAD tools specifically developed for them. Research is being undertaken in the use of DCC software for the simulation and animation of textile designs as these capabilities are currently beyond dedicated commercial apparel CAD systems. Technical aspects of textile manufacture have also forced the development of dedicated textile CAD system capable of efficiently modelling the geometric structure of textiles.
CAD has the capability to support mainstream manufacturing techniques, where anything that can be created using such manufacturing techniques can be visualised in a CAD environment. However, the literature has also shown that the design potential and geometric complexity achievable by additive manufacturing has moved beyond CAD's currently capabilities. Research using CAD systems in the design process of manufacturing products using additive manufacturing techniques has highlighted the restrictive nature it can have on design intent. These limitations are further reinforced when considering the potential of RM to create functionally graded materials, where experimental CAD software currently provides the only possible solution. CAD remains the only meaningful tool for the creation of the 3D design data necessary for additive manufacturing techniques. However, if CAD is going to unlock the full potential of RM, advances are required in the design of the systems, and also the GUI.

4.4.1 Final comments on CAD
The initial review of textiles highlighted the limitations of conventional textile manufacture when considering high performance textiles. The subsequent review of RP and RM suggests that a solution to this issue would be the use of additive manufacturing techniques. The literature surrounding CAD and CAD in the textile industry shows that a range of tools are available to assist in the creation of 3D design data. Where limitations do occur in this process, alternative CAD based solution are being utilised or developed. DCC systems have been utilised for textile simulation and animation and textile CAD has been developed for the actual modelling of the hierarchal geometric structure of textiles. The literature review also shows that while CAD is an extremely powerful tool for the creation of 3D design data, its current capabilities are being stretched by the geometric complexity achievable by RP and RM systems.

In order to determine the capabilities of CAD for the generation of RM textile 3D data, the initial experimental work will involve the modelling of RM textiles structures. However, the next topic to feature will be a discussion of the potential of RM textiles.
Chapter 5: Potential and initial modelling of Rapid Manufactured textiles

5.1 The potential of RM textiles

As discussed in the introduction, Rapid Manufactured textiles are fundamentally different in their construction to conventional fibre based textiles. While they are still hierarchical structures, the hierarchy consists only of linkable geometries, as shown in Figure 5-1.

Figure 5-1: single link, the base component of the hierarchy (left), RM textile structure, complete hierarchy (right)

The free movement and drape characteristics achieved in RM textiles are not created from yarn type structures using fibres, as conventional textiles are, but by the interlinking of linkable geometries or links. This fundamental difference means the manufacturing constraint limiting fibre complexity for the sequential manufacturing of yarns in conventional textiles can be completely removed. The shift from fibres to links therefore enables a substantial increase in the level of geometric complexity achievable for the base structure of any RM textile, and therefore the complexity of the textile itself.

The level of geometric complexity that can be achieved in the manufacture of fibres for conventional textiles is ultimately restricted to what can be achieved from the basic cylindrical fibre format, a current constraint of the fibre manufacturing process.
Sequentially, the fibres must have the capability to be manufactured into a yarn, potentially further limiting their level of geometric complexity. With RM textiles, the links are removed from any sequential manufacturing stages and the only constraint becomes the capability of interlinking with adjacent links within the RM textile structure. Therefore, as long as the geometry fulfils this simple constraint, the links are free to incorporate any level of geometric complexity. The design freedom achievable by additive manufacturing means that the link design could potentially include fully optimised smart structures, flexible linkages promoting elastic properties, hollow structures for thermal insulation, enclosed pockets for buoyancy and other potential structures that can only be manufactured using additive manufacturing techniques. In addition, because an additive approach is being taken, the opportunity exists to incorporate different designs of links in different locations within the final RM textile structure. If each different design were to offer a different functionality, therefore multiple functionalities can be created within a single textile structure using only one manufacturing technique.

An important point for consideration is the micro level design of current textile fibres. While the level of geometric complexity in fibres has been restricted, the diversity in mechanical properties and functionalities that can be created in current textiles is immense. If we consider the design freedom capabilities of RM, and apply this to the links, then the potential of RM textiles becomes considerable, removing all of the DFMA constraints currently experienced by conventional manufactured textiles and introducing the concept of creating textile functionality through design. The functionality of the link could be further enhanced with the use of functionally graded materials.

While conventional high performance textiles can only be manufactured in 2D laminated structures, the advantage of utilising an additive manufacturing approach is the capability of creating conformal high performance RM textiles. Using an additive approach, there is no longer any necessity to manufacture high performance textiles in flat sheet configurations. If a conformal RM textile is required, the textile could be modelled in its conformal state during the CAD phase of the manufacturing process, resulting in the manufacture of truly conformal high performance textiles.
The level of complexity achievable and the capability of creating conformal high performance textiles validate the potential use of additive manufacturing for the creation of textiles. Through additive manufacturing techniques, incorporating their immense design freedom and unrivalled ability to manufacture geometry of increasing complexity, the concept of designing textiles entirely for specific applications with no conventional DFMA criteria becomes a manufacturing reality. The ability to manufacture textiles that are tailored to specific applications could enable the generation of new and hybrid textiles for a multitude of current applications and new applications only realised through the manufacture of functional specific RM textiles. In addition, the application of additive manufacturing can also enable the manufacture of fully finished customised items of clothing, new high-tech smart textiles capable of executing specifically designated tasks, components that transition from solid to textile, such as optimised footwear, and the potential to give textiles added functionality through design. If the paradigm of functionally graded materials develops as predicted, the generation of smart clothing with integrated circuitry, sensors and actuators may become possible in a single manufacturing process.

Importantly, no true RM system currently exists that can be used to manufacture RM textiles for research purposes. However, it is widely accepted that RM systems will be developed and that they will be an evolution of currently available RP systems. While RM systems will offer a range of advantages over existing RP systems, for example higher resolution and a wider portfolio of build materials, the level of geometric complexity achievable will be comparable. Therefore, for the RM textile research, it is appropriate to use currently existing RP processes to manufacture RM textile designs as these processes will simulate the capabilities of future RM systems.
5.2 Initial exploration

5.2.1 Problem definition

The literature review and previous discussion regarding the potential of RM textiles have provided an argument that justifies the further exploration of the manufacture of textile structures by additive techniques for the creation of 3D conformal high performance RM textiles. The main argument in favour of RM textiles is the current inability of conventional textile manufacturing techniques to efficiently produce conformal high performance textiles, an issue that can be potentially resolved if an additive-manufacturing is utilised.

However, before the novel application of RM can be fully explored and further research undertaken regarding its various intricacies and potential applications, the fundamental issue surrounding RM textiles is the development of a robust methodology that allows the 3D geometric data required for manufacture to be efficiently generated. As no such methodology currently exists, developing such a methodology that efficiently generates conformal RM textile 3D data will therefore provide the first step in the manufacture of textiles by additive manufacturing techniques.

The development of such a robust methodology, which will be fundamental to further research in the new area of RM textiles, therefore provides the research direction of this doctoral thesis.

In order to address this fundamental issue, and therefore develop the required methodology capable of efficiently generating conformal RM textile 3D data, it is important to embark on some initial modelling work to refine the broad research issue into specific research problems that can be successfully addressed, creating the final required methodology.
5.3 Initial research aim

The development of a robust methodology for the generation of conformal RM textile 3D data required for manufacture.

5.3.1 Initial approach

The approach required for developing the required methodology must begin with the initial experimental modelling of RM textile structures. Only by creating such geometric structures can the requirements of the methodology be fully understood and established. Having completed the initial modelling of RM textile structures, the general research issue of developing a robust methodology can be refined and a more detailed research methodology developed to resolve the redefined issues.
5.4 Initial modelling of RM textile structures

5.4.1 Introduction
The application of additive manufacturing for the manufacture of textiles is a completely novel research area in which no prior in-depth research exists. The initial development that has been undertaken prior to this research centred on the creation of flat sheets of RM textiles based on historical chain maille examples.

The main requirement of RM is to have a robust 3D CAD model of the target geometry to be manufactured. This initial stage represents the fundamental process when creating all additively manufactured geometries. Therefore, the purpose of this initial modelling work was to determine the limitations associated with the creation of conformal 3D RM textile CAD data which will aid in the development of a methodology for creating conformal RM textile structures.

While future RM systems will have the added advantage of increased resolution capabilities enabling the manufacture of micro level RM textiles, current RP processes are limited to a resolution considered to be at the macro level. The purpose of this initial experiment of this work was to understand the limitations of modelling RM textile structures rather than their design and actual manufacture; therefore the RM textiles used were modelled at the macro level.

The first design exercise was to ascertain the level of complexity achievable for the link design in conventional CAD. The second was to determine the capability of creating conformal RM textile structures using test geometries of a cylinder and a hemisphere.

5.5 Initial modelling with conventional CAD
The 3D modelling of RM textiles requires each individual link with the textile hierarchy to be geometrically modelled, which is necessary for the subsequent creation of the RM
textile STL data required for manufacture. Initial modelling work was undertaken using conventional engineering based CAD software, Solidworks, [116].

5.5.1 Initial modelling of flat RM textiles
The initial modelling of RM textile structures showed positive results; individual simple link geometries could be modelled efficiently and an array created that resulted in the creation of a complete RM textile structure as demonstrated in Figure 5-2.

![Figure 5-2: Simple RM textile, design 1](image1)

The configuration of the array could also be easily modified resulting in the creation of an alternative RM textile as shown in Figure 5-3.

![Figure 5-3: Simple RM textile, design 2](image2)
This particular configuration as shown in Figure 5-3 exhibits a triangular hierarchy and can easily be achieved through conventional CAD. The next stage in the experimental process was to increase the level of geometric complexity of the actual link structure.

Figure 5-4 and Figure 5-5 show a marked increase in the level of geometric complexity of the individual link structures. Again, using a simple array technique within conventional CAD, the complex links can be arranged into a flat complex RM textile structure.

Figure 5-4: RM textile, design 3

Figure 5-5: RM textile, design 4
Figure 5-5 also shows an increase in the hierarchy of the RM textile, where two different geometries have been utilised for the creation of the final RM textile structure. Again, this modelling situation can easily be achieved through conventional CAD for the creation of planar RM textiles.

5.5.2 Initial modelling of conformal RM textile structures

The next stage in the experimental process was the attempted creation of conformal RM textile structures. The initial link design chosen for this exercise was the simple RM textile configuration as shown in Figure 5-2. This configuration exhibits the highest level of free movement between individual link geometries and is therefore the most likely RM textile link structure to conform to a curved geometry.

The initial curved geometry selected for this exercise was a cylinder. The main aim of this exercise was to ascertain if a cylindrical RM textile of any dimension could be generated within conventional CAD. It should be noted, that for the creation of a complete perfect cylindrical RM textile structure, the link spacing would obviously be a function of the circle describing the cylinder as shown in Figure 5-6.

![Circular array of RM textile links](image)

Figure 5-6: Circular array of RM textile links

Figure 5-7 demonstrates that the creation of cylindrical RM textile was possible using conventional CAD software. The construction method relied on the generation of a suitable circular ring of links also shown in Figure 5-7 using a circular array. Once
created the ring of links was simply repeated using a linear array to create the final cylindrical RM textile structure, also shown in Figure 5-7.

![Figure 5-7: Cylindrical RM textile construction (left) and complete cylindrical RM textile structure](image)

The next area of investigation was the generation of a hemispherical RM textile structure, exhibiting a substantial increase in the level of complexity, as the surface of a hemisphere possesses curvature in all three Cartesian axes, whereas a cylinder only possesses curvature in one axis. The results of generating a conformal hemispherical can be seen in Figure 5-8. It was quickly established that the generation of a conformal RM textile structure was not possible using the link configuration used for the creation of the cylindrical structure.

![Figure 5-8: Attempted hemispherical RM textile structure, error shown as red links](image)

The link configuration used in this experiment was based on quadrilateral patterning as shown in Figure 5-2. Initially, this was cited as the restricting factor which prevented the
generation of a hemispherical RM textile, as the uniform mapping of quadrilaterals or squares over a hemispherical surface is impossible without shearing. Therefore a second attempt was made using the triangular based RM textile configuration as shown in Figure 5-3. The reasoning for the use of the triangular link configuration was that a triangular configuration possesses greater tessellating capabilities, where the mapping of triangles on a hemispherical surface should be possible.

However, it was again established that this was impossible using a uniform link configuration. Whereby the uniformity describes the relationship between individual links within the RM textile structure, with all the links separated by the same dimensional value. By removing this uniformity and therefore altering the dimensions of a proportion of the individual links to accommodate the irregular spacing created, it does become possible to create a hemispherical RM textile structure using conventional CAD, as demonstrated in Figure 5-9.

![Triangular based RM textile hemisphere](image)

Figure 5-9: Triangular based RM textile hemisphere

However, the conformal structure created is no longer uniform and this will therefore affect the eventual mechanical properties of the resulting RM textile. Additionally, the generation of the conformal RM textiles required an extensive amount of manual manipulation of the individual links, resulting in a laborious modelling task. This was due to the absence of any array function describing the hemisphere.
Reconsidering the generation of planar and simple cylindrical RM textile structures, it should be noted that these were efficiently generated through the use of linear and circular array functions. The array function within any CAD system provides set points in Cartesian space using dimensional values set by the user to which geometry can be copied and repositioned, as demonstrated in Figure 5-10.

Figure 5-10: The generation of planar RM textiles structures using a linear array

Figure 5-10 shows a planar RM textile structure and its creation from one set of repeating links using a linear arraying function. If the structure of the array locations is considered, as demonstrated in Figure 5-10 (right), it can be seen that the actual structure of the array locations can be compared to that of a surface mesh. In addition, if the generation of a cylindrical RM textile is considered, as demonstrated Figure 5-11, using a combination of circular and linear arrays, again the actual array locations can be compared to the structure of a cylindrical surface mesh.
The efficient generation of RM textile structures using array function can therefore be considered as a process of mapping repeating geometry to a surface mesh, where the surface mesh describes locations in Cartesian space where RM textile link geometry can be copied and repositioned creating the complete RM textile structure.

Returning to the generation of hemispherical RM textile structures, it was demonstrated in Figure 5-9 that the creation of such a structure was only achievable through the manual manipulation of individual link geometries as no array function was able to fully describe the underlying hemisphere structure. If the array function is considered as a surface mesh once more, the generation of a suitable uniform hemispherical mesh would therefore provide locations in Cartesian space to which RM textile link geometries could be positioned, thus allowing the efficient generation of hemispherical RM textile structures. The absence of any array function or hemispherical surface mesh means that the creation of conformal RM textile structures within conventional CAD is limited and the only plausible methodology of creating such structures through conventional CAD requires the manual manipulation of link geometries. While this may be acceptable when using macro level sized links and a comparatively small hemisphere, if micro level sized links are considered, combined with a proportionally much larger geometry (such as a human form), then the manual manipulation of individual links would become unfeasibly time consuming.
To further emphasise these limitations, a CAD model of the world’s first 3D conformal seamless RM textile garment was produced [70,117], demonstrated in Figure 5-12.

Figure 5-12: World’s first 3D conformal RM textile garment, CAD model (Left), actual garment (Centre & Right) [70]

The CAD modelling of this RM textile garment required two months to construct utilising the link configuration shown in Figure 5-2. The time required to complete the CAD model was solely due to the extensive manual manipulation of the individual link geometry. However, if a suitable uniform mesh of the underlying surface geometry was created and the link geometries mapped to it, the manual manipulation of link geometries could be removed from the process, allowing the efficient generation of conformal RM textile structures.

It can therefore be concluded that while conventional CAD systems have the capability to provide the level of geometric complexity required for the individual link geometries of RM textiles, they do not have the capabilities for the efficient generation of uniform conformal RM textile structures, as no automated mapping of the link geometry by an array function can be achieved over curved surfaces. However, the use of the array function has highlighted a potential methodology for the creation of conformal RM textile structures, by mapping repeated geometry to a suitable uniform mesh of the underlying...
geometry that is to be ultimately achieved. This realisation provides an important step forward in the creation of a methodology for the generation of conformal RM textile structures. While the process of mapping to a mesh could be achieved in conventional CAD, where a suitable uniform mesh could be established that provided the link geometry locations, again this would require the manually positioning of the link geometry to create the final conformal RM textile structure. Therefore, in order to create conformal RM textiles in a reasonable time frame and therefore efficiently, an alternative method of producing the CAD data is required, where the automated mapping of link geometries to a suitable surface mesh could be achieved.

5.6 Initial modelling with textile CAD

An alternative option for the geometric modelling of RM textile structures is the use of dedicated textile CAD system, as discussed in the literature. Textile CAD has been developed to effectively model the actual complex geometric structure of conventional textiles and provides an alternative to conventional CAD. While no commercial textile CAD systems are currently available, an example of this type of system, TexGen, is being developed at the University of Nottingham [118].

The TexGen software was developed for the accurate and comprehensive Finite Element (FE) modelling of the textile component of fibre reinforced composite structures. The purpose of the software is to generate FE models for mechanical analysis of dry textiles and textile composites, as well as Computational Fluid Dynamics (CFD) to model infusion of textiles during composite manufacture. The software can also be used to export the textile FE models as 3D CAD geometry. The TexGen system uses the concept of the Representative Volume Element (RVE) as discussed previously. The RVE is the smallest possible repeating volumetric element and the only part of the textile structure to be geometrically modelled. Once the RVE has been established, the TexGen system then requires an FE created volumetric mesh of the intended final net shape of the textile reinforcement. Using commercially available FE pre-processing software, the 3D geometry approximating the final intended textile structure can be
imported and meshed. The mesh can then be imported into TexGen where the system populates the RVE within the enclosed sections of the volumetric mesh, matching the normal, rotation and scaling factor to create the final geometric textile structure.

In order to evaluate the TexGen system for its potential use in the generation of conformal RM textile structures, collaboration was initiated with Nottingham University to adapt their TexGen software to model RM textile link geometries that could be then mapped to a suitable surface mesh for the creation of conformal RM textile structures.

The first modelling exercise using the adapted TexGen software was the creation of a hemispherical RM textile structure, where conventional CAD had previously shown limitations. Using the TexGen system, the first requirement was the creation of a RVE of the RM textile to be used, demonstrated in Figure 5-13.

![Figure 5-13: Basic RM textile (Left) and the associated RVE (Right red)](image)

Having established the RVE, the next requirement was the creation of surface mesh approximating the final intended shape of the hemisphere. The hemispherical mesh was created using Fluent's Gambit pre-processor [119], shown in Figure 5-14. Once the mesh had been imported to TexGen, the system was then used to map the RVE to the enclosed section of the quadrilateral surface mesh, the result of which can also be seen in Figure 5-14.
While the RM textile produced using this system is not fully uniform, the system required no intervention and was fully automated once the surface mesh had been imported and the RVE fully established. The adapted TexGen system therefore offers a completely automated process for the efficient generation of conformal RM textile data and provides another important step forward. This approach was further investigated and a methodology was established for the efficient generation of conformal RM textile structures, as follows:

- 3D surface incorporating the final net shape of the desired RM textile created in CAD
- Finite Element (FE) mesh created of the conformal surface and imported to TexGen
- RVE established within the textile CAD
- RVE mapped to the enclosed sections of the surface mesh
- Complete conformal RM textile structure exported as STL data
- STL uploaded to RM system for manufacture

5.6.1 Limitation

The use of the adapted TexGen software and highlighted methodology was exceptionally fast compared to conventional CAD, having the capability of creating the RM textile hemisphere within a two minute time frame. However, there are limitations with this approach. TexGen requires a vectorial description of the individual RM textile links for the creation of the RVE, from which smoothed paths are generated using simple parametric curves, (e.g. Bezier curves) as demonstrated in Figure 5-15.
The 3D geometry of the RVE was then generated by sweeping a cross section along these paths creating the individual RM textile links, as demonstrated in Figure 5-16. This system is therefore severely restrictive when considering the level of geometric complexity required for RM textiles. Due to the vectorial geometry description the system at present is limited to the level of geometric complexity shown within the example.

A second limitation of the system is the requirement of using quadrilateral FE meshes. Currently, the meshes that are created within conventional FE software are not uniform in their structure, due to the limited capability of a quadrilateral mesh to uniformly tessellate over conformal surfaces, as shown in Figure 5-17.
Where deformation occurs in the FE created mesh, this is reflected in the final RM textile structure, as TexGen manipulates the RVE to fit in the enclosed quadrilaterals of the mesh, as demonstrated in Figure 5-18.

Figure 5-18 shows a conformal surface that has been meshed using conventional FE software and used by TexGen to map the RVE. The resulting deformation and inconsistency within the RM textile structure would therefore potentially adversely affect the resulting properties and performance of the eventual RM textile.

5.7 Discussion of initial modelling

Through the use of conventional CAD it becomes possible to create virtually any level of geometric complexity a user would desire for the design of individual RM textile link
structures. However, when contemplating conformal RM textile structures, the issue is not the creation of a single or linear and circular array of complex geometries, but the automated creation of a uniformly distributed collection of complex geometries that ultimately creates the hierarchy of conformal RM textile structures. This is where conventional CAD showed limitations, as curved geometry cannot be accurately described by any array function and no automated mapping systems exist. The initial modelling work using conventional CAD systems did however highlight the efficiency of mapping geometry to a mesh compared with the manual manipulation of link geometries.

The use of textile CAD as an alternative does provide the capability of automatically creating a conformal RM textile structure by mapping geometry to a mesh. However, the limitation with textile CAD is the inability to create the level geometric complexity that will ultimately be required.

The methodology employed by TexGen also relies on the generation of a suitable quadrilateral FE created surface mesh. As previously discussed, the RM textile geometry is then mapped to the individual elements of the mesh creating the conformal RM textile structure. However, it was seen that any distortion apparent in the surface mesh would automatically reproduce distortion within the finally created RM textile structure itself. When considering potential future applications of this type of textile manufacture, any distortion in the complex RM textile geometry would almost definitely produce an adverse effect on the textiles performance or functionality. Secondly, due to the manufacturing process itself, if the distortion were to be so severe that individual RM textile geometries began to interfere with one another and occupy the same three-dimensional space, then this would be reproduced during the manufacture stage, again affecting functionality and performance. However, the creation of a uniform surface mesh would resolve this issue and the methodology of mapping to a surface mesh is extremely effective at automatically creating a conformal RM textile structure and provides the only plausible methodology at present.
5.7.1 Conclusion
The initial experimental work regarding the modelling of RM textiles, using currently available methods has shown that there are severe limitations in this process. Additionally, the initial modelling work shows that no currently available system has the inherent capabilities required for producing uniform conformal RM textile structures to the level of geometric complexity that will ultimately be required.

The use of conventional CAD allows the creation of RM textiles with a high level of geometric complexity, although only in planar configurations or in very limited, manually-created conformal configurations. While this was not an exhaustive study of current CAD systems, the Solidworks software utilised is indicative of the capabilities of current CAD systems. In contrast, the use of textile CAD allows the automatic creation of conformal RM textile structure, although these can only be created using a very limited level of geometric complexity. When considering the complete design freedom and geometric complexity that Rapid Manufacturing can achieve, where the future potential truly offers conformal high performance textiles incorporating highly complex micro-level geometry, then it becomes obvious that a major research issue surrounding RM textiles is the efficient generation of uniform conformal RM textile data of a high geometric complexity.

Therefore, in order to truly create uniform and conformal RM textile data that will enable the full potential of RM textiles to be explored with further research, a new dedicated RM textile CAD tool is required that incorporates the level of geometric complexity associated with conventional CAD and the automated mapping capabilities of textile CAD. The investigation of a new CAD tool specifically for RM textiles therefore provides the redefined direction for this doctoral thesis and research.
Chapter 6: Research methodology

6.1 Problem identification

The conclusion of the initial modelling work has shown a requirement exists for the development of a new dedicated RM textile CAD tool. This is due to the limitations inherent with the current generation of 3D data required for manufacture. These limitations are one of the fundamental research issues that are currently preventing further research being undertaken in the novel area of conformal RM textiles. The development of such a tool that allows 3D conformal data to be generated efficiently is therefore a significant research issue and the direction of this thesis.

The investigation of a dedicated RM textile CAD tool is a significant task but inspiration can be taken from the initial experimental work for the actual working methodology it could utilise. Recognising the methodology of mapping to a mesh that both CAD systems utilised, these systems provide an elegant solution of generating conformal RM textile structures efficiently. Secondly, by mapping to a uniform and equidistant surface mesh, the resulting RM textile would therefore exhibit a conformal uniform hierarchy. The research direction for this thesis can therefore be split into two separate aims:

6.2 Research aims

- **To investigate uniform and equidistant surface meshing**
  A detailed investigation of current surface meshing capabilities, followed by the investigation of a new uniform and equidistant meshing algorithm that would ultimately enable the uniform conformal mapping of complex RM textile geometries.

- **The investigation of a complex geometry mapping tool**
  To create a new mapping tool that has the capability to accurately map any level of geometric complexity specified by the user to a surface mesh generated by the previous work and export the geometric data required for manufacture.
Achieving both of these research aims would therefore result in a complete methodology for the creation of true conformal RM textile structures incorporating high levels of geometric complexity.

6.3 Research approach and objectives

The two research aims, whilst inherently linked by the requirement of conformal RM textiles are actually very separate research issues. The first research aim will require an extended literature review of current surface meshing techniques and potentially related issues. This initial investigation will be required to ascertain the current capabilities of meshing techniques to determine if a new meshing algorithm is required. If this is true, the remainder of this research would involve the development of such an algorithm and its subsequent testing and validation. The second research issue will require the development of a mapping methodology which can be tested for accuracy and used in collaboration with the uniform and equidistant mesh algorithm. The approach for the research can therefore be separated into two research issues with specific research objectives thus:

**Research Issue 1:**
1. Extended literature review of current surface meshing techniques and related issues
2. Investigation of uniform and equidistant meshing
3. Potential testing and validation of any existing or newly developed meshing technique

**Research Issue 2:**
1. Investigation of complex geometry mapping
2. Potential testing and validation of any existing or newly developed mapping tool

The successful conclusion of both these research issues will therefore be the creation of a new methodology for the generation of conformal RM textile data of a high geometric complexity suitable for manufacture.
Chapter 7: Mesh generation

7.1 Introduction

The purpose of this second extended literature review is to investigate currently available meshing techniques for their potential application as a mapping mesh required by the methodology for creating uniform conformal RM textiles.

The main requirement of the surface mesh, when used as a means of providing mapping locations over curved surfaces for the accurate positioning of RM textile link geometries, (as highlighted in the initial experimental work) is for the mesh to have a uniform structure and equidistant point spacing. For a surface mesh to exhibit such qualities, then the only possible structure the surface mesh can be generated from, are triangular elements. To clarify, if a section of a uniform and equidistant example surface mesh is considered, as demonstrated in Figure 7-1 the structure of the mesh can be described through a system of nodes and elements. The nodes of the mesh, shown as red spheres in Figure 7-1 are locations from the underlying geometry that are described by the mesh. The elements of the mesh, shown as grey cylindrical geometries in Figure 7-1, are described as the geometry connecting the nodes of the mesh, completing the surface mesh structure.

Figure 7-1: Uniform and equidistant mesh structure
For the surface mesh to be uniform, each node contained within the mesh must be connected to a maximum of 6 other nodes in a triangular, hexagonal configuration. Also, for the mesh to be equidistant, the distance observed between each node, shown as (x) in must all be constant, and therefore, the elements of the mesh must all be equal.

In addition to the structure of the mesh required by the conformal RM textile generation methodology, a second requirement is the control of the actual element dimensions, and therefore the equidistant spacing of the nodes observed in the mesh. The control of this dimension will be required when contemplating the design of RM textile link structures, as one design will require a certain dimensional spacing value compared with an alternative value for a different design. The capability to define the dimensional spacing of nodes within the mesh will therefore enable a customised surface mesh for a particular RM textile link design to be generated. The creation of a suitable uniform and equidistant surface mesh with user defined node spacing over a curved geometry could then be used to define mapping locations for RM textile link geometries.

7.2 Mesh generation

The meshing of 3D geometry, either volumetric or surface configurations, is a prerequisite for Finite Element Analysis (FEA). However, the meshing of surface geometry is a requirement for many different applications other than FEA. These other applications include: computer graphics [120], data visualisation, RP STL creation [121,122] and computational geometry [123]. While such applications have generated methods of producing surface meshes, the results are often unsuitable for FEA as the meshes created may include very thin (poor aspect ratio), elements and a lack of any discrete element size control. These limitations also make them unsuitable for the requirement of RM textile link geometry mapping, as the generation of high quality uniform and equidistant mesh structures with the additional capability of controlling element size is ultimately required.
While numerous specialised meshing techniques have been developed for the purpose of surface meshing, it is not the intention of this thesis to document and explain each one in detail. It is however appropriate to document the main techniques utilised in surface meshing and in particular, to highlight those that have the capability to create uniform and equidistant mesh structures suitable for RM textile link geometry mapping.

Of all the applications of surface mesh structures cited, the field of FEA has received the most intensive mesh generation research and development for the creation of high quality mesh structures. The generation of surface meshes for FEA have been developed from techniques utilised for the primary requirement of volumetric mesh generation. It is therefore appropriate to review FEA and the main volumetric meshing techniques utilised before addressing the specialised surface meshing techniques developed for FEA.

7.3 Introduction to FEA

From its conception during the 1940s until around a decade ago, FEA had been performed exclusively by specialised analysts in the subject, who had devoted their careers to the discipline. In the past decade the field of FEA has seen great development, with an increase in the number of computer packages available to all levels and all types of engineers [124].

FEA involves the use of a complex system of points, called nodes, that form a grid (or mesh) of elements, across and within a targeted geometry. The user assigns nodes at a particular density throughout the usually volumetric geometry, depending on many considerations, that may include the anticipated stress levels of a certain area and the detail required in the final results. The generated mesh is then assigned data on material and structural properties, which allows the prediction of how the geometry will react to certain load conditions. In essence, finite element analysis is a numerical method used to solve engineering problems that involve stress analysis, heat transfer, electromagnetism, and fluid flow [124].
7.4 FE pre-processors

FE pre-processors, or meshers, are dedicated software systems that now have the capability to automatically generate the mesh necessary for FE analysis, removing the requirement of the user to manually create a suitable mesh of the geometry to be analysed, as demonstrated in Figure 7-2. However, the type of mesh used and the method of its creation are very different for different applications. For example, tetrahedral meshes work well when calculating stresses in solids, whereas quadrilateral meshes are better at calculating strain in stamped metals. When investigating fluids or CFD analysis, hexahedron meshes are generally preferred [124].

![Figure 7-2: FE meshed exhaust manifold][125]

The next section in this chapter relates to the main volumetric meshing techniques currently utilised in commercial FE pre-processor packages.

7.5 Meshing techniques

Triangular and tetrahedral (a platonic solid containing 4 equilateral triangles) meshes are by far the most common forms of mesh types available for FEA. However, quadrilateral, pyramid, prism and hexahedron elements can also be utilised [126].

While numerous specialised meshing techniques have been developed, it is the intention of this thesis to document and explain the main techniques utilised in the
creation of FE meshes and highlight those that have the potential to create uniform and equidistant meshes suitable for mapping RM textile link geometries. Further information regarding FE meshing techniques are available in published works [129-163].

Volumetric meshing techniques currently available in commercial FEA packages can fit into one of three main categories:
- Octree
- Advancing Front
- Delaunay Triangulation

7.5.1 Octree
The oldest meshing methodology in operation, the Octree or Quadtree meshing technique [127, 128], relies on the mapping of cubic cells to the 3D geometry required for analysis. Figure 7-3 shows an example geometry; once the cubic cells are mapped over the entire 3D geometry, the cells are continually subdivided until the required mesh resolution is created, as demonstrated in Figure 7-4. Having reached the required mesh resolution, irregular cells are created where the mapped cubes intersect with the target geometry surfaces.

Figure 7-3: Simple 3D geometry required for meshing by Octree

Figure 7-4: (left) Mapping and subdivision of cubic cells, (right) mesh creation with irregular cells
The Octree method continues by creating tetrahedrons (pyramids) from the irregular cells on the geometry boundary and the internal irregular cells. The surface mesh of the geometry is then created from the tetrahedrons intersecting with the surface of the 3D geometry, creating a mesh of random element sizes. Due to the random mesh elements created, mesh smoothing and clean up operations (covered in section 8.5) can be employed to maintain quality within the mesh. As this method requires a significant number of surface intersection calculations to complete the mesh, the overall processing and completion times for the mesh are higher than those of Delaunay and Advancing Front methods [126].

7.5.2 Advancing Front

The advancing front or moving front method [129,130,131,132] utilises tetrahedrons for volumetric meshing that are created progressively inward from the boundary of the geometry. During this inward progression, an active front is maintained where new tetrahedrons are formed. Using a simple 3D example geometry, as demonstrated in Figure 7-5, triangles can be formed on all of the geometry's boundaries, as shown in Figure 7-6; as the algorithm progresses, the inner domain of the geometry becomes filled with triangular elements.

Figure 7-5: Target geometry for Advancing front meshing

Mesh creation at all geometry boundaries

Figure 7-6: Advancing front meshing, detailing boundary meshing
Each triangular element produced will therefore provide an ideal location for the fourth node of a 3D tetrahedral element. The algorithm then determines whether to create this new node or use an existing node from other tetrahedrons already existing in the mesh, before executing an intersection check on tetrahedrons advancing from opposing boundaries of the geometry. A sizing function can then be employed to help maintain element sizes within the mesh, although this value is used as a guide and not an accurate dimension [133]. A version of the advancing front mesh algorithm is available with the ANSYS suite of mesh generation tools [134].

7.5.3 Delaunay triangulation

The Delaunay triangulation method [135] is the most commonly implemented triangular meshing technique currently available to FE users. While the Delaunay triangulation itself is not a complete meshing algorithm, it is a robust method of connecting existing nodes of a mesh to create elements. The Delaunay triangulation stipulates that for nodes to be connected, all nodes must not be contained within the circumsphere of any tetrahedron within the mesh, where a circumsphere is defined as a sphere passing through all four vertices of a tetrahedron.

This criterion can be easily demonstrated using a 2D example as shown in Figure 7-7 and Figure 7-8. Using a 2D example of the Delaunay triangulation, a circle can be formed through three existing nodes to create a triangular element as shown in Figure 7-7 (left). If two further nodes are considered for triangulation (A and B) shown in Figure 7-7 (right), the Delaunay triangulation is only valid if all four nodes of two connected triangular elements are not contained within the circle, defined by the three nodes of the original triangular element. Therefore, Figure 7-8 (left) shows the creations of two triangular elements that invalidate the Delaunay triangulation, while Figure 7-8 (right) shows two triangular elements that conform to the Delaunay triangulation, where only three nodes reside in any one created circle.
Circle created from three nodes of triangular element

Figure 7-7: (left) Circle created from initial triangular element, (right) two further nodes for triangulation

Invalid triangulation as four nodes are contained with one circle

Valid triangulation as only three nodes are contained with any one circle

Figure 7-8: (left) Invalid Delaunay triangulation, (right) valid Delaunay triangulation

The Delaunay triangulation was developed in the early 1930's, however the triangulation itself was not implemented in meshing algorithms until the late 1970's [136,137] when combined with a suitable node placement or point insertion algorithm. A typical approach for meshing using the Delaunay criterion is to create a simple boundary mesh of the geometry (similar to the first stage of the advancing front technique) and then triangulate those nodes according to the Delaunay triangulation. Using this method, nodes are inserted incrementally within the existing mesh. The method employed for the generation of the initial nodes, when combined with the Delaunay triangulation, distinguishes one Delaunay meshing algorithm from another. More information can be found in various published works regarding differing Delaunay based meshing algorithms [126,138].
7.6 Post-processing

With the automatic meshing of any target geometry, post processing of the mesh will generally be required to introduce some level of optimal element shape and mesh quality. In order to achieve this, two main categories of mesh enhancement exists: smoothing operations and clean up operations [126]. Generally, mesh smoothing includes any process that adjusts node position while maintaining element connectivity, and mesh clean up includes any process that changes element connectivity.

7.6.1 Smoothing operations

Smoothing operations generally employ an iterative process to manipulate the position of nodes within a mesh to improve element shape and quality. The techniques and methods implemented for this purpose can be categorised into three main groups:

- Averaging methods
- Optimisation methods
- Physical methods

**Averaging methods:** The simplest form of mesh smoothing algorithms involves Laplacian smoothing [139]. Due to the simplicity of the methodology, only slight modification of the technique is required to incorporate different element shapes, making it simple to implement and accounting for its wide scale adoption.

**Optimisation methods:** A more complex approach to mesh smoothing requires the use of optimisation algorithms that can be employed to improve element quality [126]. The smoothing techniques measure the quality of the surrounding elements for a particular node and attempts to optimise the elements by calculating the local gradient of the element quality with respect to the node location. The node is translated in the direction of the increasing gradient until an optimum is reached.

**Physical Methods:** A physically based method represents another alternative to mesh smoothing. Using physical principles such as force attraction or repulsion, nodes can be translated within the mesh based on an equilibrium methodology. Physically based
algorithms have been developed using spring systems [140] and sphere or bubble systems [141,142].

7.6.2 Cleanup operations
Similar to mesh smoothing, a variety of methods and techniques can be utilised to improve mesh quality by mesh cleanup operations. As previously stated, mesh cleanup involves the alteration of the mesh element connectivity. Before mesh cleanup methods can be performed, generally some criteria must be met. The general criteria can be defined as:
- Shape enhancement
- Topological enhancement

Shape enhancement: When considering triangular or tetrahedral meshes, simple shape enhancement and element quality can be achieved by swapping the diagonals of the mesh elements. A more involved approach for tetrahedral meshes utilises the Delaunay criteria to determine the best position of an element edge via swapping. In an application where a mesh may present mixed element types, the element quality of two adjacent triangles could be preferred to a single lower quality quadrilateral element, in which case a splitting algorithm can be employed.

When considering highly curved geometries and surface meshing, the resulting elements generated may deviate significantly from the target geometry surface. For a triangular mesh, edge swapping can be performed based on which edge position of the element will deviate least from the surface. In addition, a local reduction of the element sizes of the mesh within certain areas may also be considered to incorporate complex surface features.

Topological enhancement: The topological enhancement of mesh elements is concerned with the number of edge elements connected to each node within the mesh. When considering a triangular based mesh, the ideal number of edge elements connected to each node would be six; similarly, when contemplating quadrilateral
meshes the ideal number would be four. This property is sometimes referred to as the node valance or degree and it is assumed that local enhancement of this property would improve the overall element quality of the mesh. Several methods have been proposed for both triangular [143] and quadrilateral meshes [144,145].

7.7 Surface meshing

All of the meshing techniques previously discussed are directly aimed at meshing planar domains or geometric volumes, where a direct by-product of volumetric meshing is the creation of a surface mesh on the actual surface of the volumetric geometry. However, dedicated surface meshing methods have been developed using the principles of the volumetric meshing techniques discussed previously. Dedicated FE surface meshing algorithms can be categorised into two main groups; either indirect or direct.

7.7.1 Indirect surface meshing

Indirect or parametric space algorithms [146,147] use a NURBS representation of the target surface to form elements in two dimensional parametric space as the initial meshing step as demonstrated in Figure 7-9. On completion, the 2D elements are mapped back to the 3D surface using the underlying (u – v) representation of the NURBS surface, producing a conformal mesh as demonstrated in Figure 7-10. While this system is robust, the 2D elements created in parametric space do not always conform well to the 3D surfaces, often producing irregular elements and poor overall element quality. To address this shortcoming, mesh smoothing operations have been included in various meshing algorithms of this type to create better mesh elements when mapped to the 3D surface.
7.7.2 Direct surface meshing

In contrast to parametric space meshing algorithms, direct surface meshing algorithms form 3D elements directly on the geometry without regard to the 2D parametric representation of the geometry. One approach uses the advancing front technique [148,149], where surface normals and tangents are calculated to determine the direction of the advancing front. In addition, a significant number of surface projections are required to ensure that new nodes remain on the 3D surface and not deviate from it. Also of significance is the increased complexity of the intersection calculations required...
to ensure that triangles on the surface do not overlap with one another. A variation on this approach is the paving algorithm available in Fluent’s pre-processor software Gambit [150,151].

7.8 FEA Surface meshing techniques capable of generating high quality mesh structures

One approach to FEA surface meshing that is of particular interest is sphere packing or bubble packing. Sphere packing techniques for surface mesh generation have been developed in both direct surface meshing [141] and indirect surface meshing [152], the general principle of both techniques requires the close tangential packing of spheres or ellipsoids to completely cover the targeted surface geometry. The attractiveness of sphere packing based meshing algorithms is due to their inherent capability of producing high quality and mostly uniform elements [141]. Due to the advantage of high quality mesh generation many sphere packing algorithms have been produced with subtle differences, information regarding these subtle differences; can be found in published works [153,154,155,156,157,158,159,160,161]. In general, sphere packing techniques can be categorised as either direct or indirect meshing techniques.

7.8.1 Direct sphere packing

In the case of direct 3D sphere packing, the centre points of the spheres are constrained to the surface and packed using an advancing front technique so they are tangential to each other, completely filling the 3D surface domain. The process of sphere packing can be explained using a simple planar 2D example as demonstrated in Figure 7-11, Figure 7-12 and Figure 7-13.

Figure 7-11: (left) Target surface, (right) initial sphere placement using advancing front technique
Inevitably, overlapping of, or gaps between, spheres will occur with full coverage of curved 3D surfaces. To address the potential overlapping and gap formation, one example of direct 3D sphere packing has utilised a physical based smoothing operation to attract or repel spheres using inter-sphere forces \[141,162\]. Reaching an equilibrium state between all spheres therefore aids the reduction of overlapping or gap formation. The completion of the meshing technique typically requires a Delaunay triangulation method to connect the centre points of the sphere to complete the final 3D surface mesh.

### 7.8.2 Indirect sphere packing

In contrast, indirect or parametric space sphere packing techniques utilise tangentially packed ellipsoids (circular and elliptical based 2D elements) of different radii in the 2D parametric space of the 3D surface, again using an advancing front technique [133]. The radii of the ellipsoids are linked mathematically to the difference in curvature between the 3D surface and its 2D parametric domain. Therefore, when a triangulation technique,
typically Delaunay, is utilised to create the mesh from the centre points of the ellipsoids which in turn is mapped to the 3D surface, the variation in mesh element size is minimised.

7.9 Summary of FEA meshing

FEA pre-processors or meshers have evolved in the past decade to offer users the capability to automatically generate a volumetric or surface mesh of sufficient quality that will allow further FE analysis to be undertaken. Meshes can be created from a range of different element shapes and tailored to the specific analysis required using a range of developed meshing algorithms.

However, the actual control of the mesh resolution, aspect ratios of elements, and uniformity, is still limited. In addition, the meshes generated are dependent on the level of complexity of the input geometry, where a highly complex curved geometry will inevitably produce a complex non uniform mesh. To achieve the quality necessary for further FE analysis, currently available commercial automatic meshing algorithms include some form of mesh post processing or mesh clean up operation as the meshing algorithms available do not provide the sufficient mesh quality by themselves.

More experimental meshing algorithms such as sphere packing, which currently are not commercially available, do, however, have greater potential of creating high quality volumetric and surface meshes with almost uniform elements. This type of surface meshing algorithm therefore has greater potential as a mapping mesh for the generation of conformal RM textiles, although, again their functionality is limited and no true uniform and equidistant mesh can be created for all geometry types.

7.9.1 Final comments on FE meshing

When considering the intended use of meshing algorithms, the ultimate requirement is a complete mesh representation of the targeted geometry (usually volumetric) for further FE analysis and not the generation of an equidistant mesh. However, when considering
a simple uniform prismatic geometry, a cube for example, or planar surface, currently available meshing algorithms generally have this capability. When a complex curved surface is considered, current meshing algorithms simply have not been designed for, nor have the capability of producing an equidistant and uniform mesh, merely a mesh representation of the geometry to a standard sufficient for FE analysis.

The literature review of current FEA meshing capabilities has shown that while meshing techniques are extremely developed and diverse, no current algorithm has the capability to produce a uniform and equidistant mesh for all targeted geometries that would render them suitable for the requirement of mapping RM textile link structures. Due to these cited limitations, a further investigation is required to determine if other suitable tools are available that can accurately calculate equidistant points across surface geometries. An area of potential interest is the generation of equidistant points across the surface of a sphere. While this research area is not concerned with mesh generation or any geometry other than spheres, the mathematical principles and methodologies employed may provide a solution for the generation of a uniform and equidistant mesh structure required for the generation of conformal RM textiles. The next section of this chapter will therefore be a review of such tools and techniques.

### 7.10 The generation of equidistant points on spherical surfaces

The ability to uniformly distribute equidistant points on the surface of a sphere has attracted attention not only from mathematicians, but also from biologists, chemists and physicists [163]. This is due to potential applications ranging from global climate models, mapping the Earth's gravitational and magnetic fields, satellite global positioning, mobile communications, carbon fullerenes molecules, crystallography, geodesic domes, spherical codes, molecular modelling, virus modelling, computational geometry, computer graphics, rendering and FEA meshing [163,166,174,176].

The equidistant distribution of N points on any planar surface has simple and obvious solutions. Even with the addition of curvature in one axis, uniform points can be easily
distributed across the surface of a cylinder. However, when considering the uniform distribution of points across the surface of a sphere, or a surface that exhibits curvature in all three Cartesian axes, the task becomes increasing complex. In fact, the problem is so complex that absolute solutions only exist in special cases for \( N \) points on the surface of a sphere where \( N \) is equal to 1, 2, 3, 4, 6, 8, 12 and 20, [164, 165], with the absolute solution for any number of points being regarded in 1998 as 'a problem for the next century' [166].

7.10.1 Platonic solid solutions (special cases)
The generation of equidistant points for solutions for \( N \) points, where \( N = 4, 6, 8, 12 \) and 20 can all be generated from the five Platonic solids [165]. Platonic solids, also known as regular solids or regular polyhedra, are polyhedra with faces composed of regular polygons [167]. Using this construction method, exactly 5 platonic solids can be generated:

- Tetrahedron, demonstrated in Figure 7-14
- Octahedron, demonstrated in Figure 7-14
- Cube, demonstrated in Figure 7-15
- Icosahedron, demonstrated in Figure 7-16
- Dodecahedron, demonstrated in Figure 7-16

Encapsulating the platonic solids in spheres of the same radii therefore produces the special cases of uniform equidistant points as demonstrated in Figure 7-14, Figure 7-15 and Figure 7-16.

Figure 7-14: Platonic solution for \( N \) points on a sphere, \( N= 3 \) points (left) & \( N= 6 \) points (right)
Due to the number of potential applications, several different methods have been employed to provide a solution to this mathematical problem for \( N \) points greater than 20, where the distribution acquired is close to, but not exactly an equidistant distribution. Methods employed include:[168]

- Convex hull
- Voroni cells
- Delaunay triangulation
- Sphere packing
- Sphere covering
- Riesz s-energy
- Lagrange polynomials
- Interpolatory cubature
Of all the current methods employed, the Riesz s-energy and sphere packing and covering strategies have received the most intensive research. An interesting application is the use of sphere packing and covering for the attempted creation of uniform equidistant points over the surface of a sphere, and may account for its recent adoption in high quality mesh generation for FEA discussed previously.

7.11.1 Sphere packing and covering
The sphere packing method relates to the packing of a predetermined number of spheres on the surface of a greater central sphere. The problem is to find the centres of N non-overlapping spheres so their common radius is maximised and equalised, as demonstrated in Figure 7-17. The packing of spheres for this application is therefore in complete contrast to that of sphere packing mesh generation, where the aim is the packing of spheres with a user defined radius and an unknown number.

Figure 7-17: (left)Non-overlapping spheres, (right) Sphere covering [168]

The problem is often called the Tammes problem [169] after his work on pollen grains in 1930, or the Fejes Toth problem [170], sometimes referred to as 'spherical code' [171]. Another related problem is that of the Kepler conjecture [172,173].

In contrast, the sphere covering problem relates to the number of identical spheres that overlap and cover the entire surface of the greater central sphere when the radius of the covering spheres are minimised [174], demonstrated in Figure 7-17. While both methodologies generate good equidistant approximations for N points greater than 20, better approximations can be generated through energy minimisation strategies.
7.11.2 Riesz s-energy

The Riesz s-energy method relates to the standard Coulomb potential used to model electrons repelling each other on the surface of an atom [163]. The standard Coulomb potential relates to a simple inverse square law, where the repelling force observed by an electron is quartered each time its distance from another electron is doubled. If the atom is considered to be the greater central sphere, the electrons or points repel one another while constrained to the surface of the central sphere. Once settled, or when the repelling force creates equilibrium between N points, the distribution is considered to be uniform. The problem of finding a point set that minimises the Coulomb potential is known as the Thomson problem after work by J.J. Thomson in 1904 [175].

Recent work using the s-energy minimisation for N points greater than 100 and for geometries other than spheres has produced a method for generating large numbers of points that are spread with near equidistance over practically any surface of any dimension [176]. The research has investigated equilibrium cases in which the repelling force between any pair of particles is inversely proportional to the distance between them raised to a power. The formulation for achieving this is a generalisation of the inverse square law, where it describes the behaviour of forces such as electrical charge and gravity. The work has cited that the power depends on a parameter labelled 's' which is a variable of the derived equation. When 's' is small, the points act as if they are responding to a long-range force (like gravity or electromagnetism), and when 's' is large they act as if they are subject to a short-range force (such as the strong force that binds the atomic nucleus together). Figure 7-18 shows the change in distribution of points on a torus with different values of 's' [176].

![Figure 7-18: Distribution of points for changing values of s][176]
The continued research has also discovered and rigorously justified [176] that the critical value of $s$ is precisely equal to the number of dimensions of the surface to which the points are constrained. In the case of the mathematical two dimensional surface of a torus; for example, using a value of $s$ greater than or equal to '2' produces the most uniform distribution. As demonstrated in Figure 7-19

![Torus with points](image)

Figure 7-19: Near uniform distribution of points on torus [176]

**7.12 Summary of uniform and equidistant point generation**

The literature concerning the creation of uniform and equidistant points shows the problem to be considerably more complex than first expected. The problem is so complex that exact solution for the generation of uniform and equidistant points across the surface of a sphere are limited to 20 points. Continued research in this area has produced methodologies for the creation of $N$ points greater than 20, although these solutions are approximations only and a true solution for $N$ points greater than 20 has yet to be discovered.

Methodologies that have produced almost exact approximations utilising Riesz $s$–energy minimisation can now create an almost uniform and equidistant point set greater than 20. However this method is based on a defined number of points and not point spacing dimension. Therefore it may never be possible to determine a point set over a sphere or specific surface with a specific value.
7.12.1 Final comments on uniform and equidistant point generation

The literature section concerning FE meshing has shown that current meshing algorithms do not have the capability to create uniform and equidistant meshes for complex geometries and in fact have not been designed to do so. The literature shows that current FE meshing algorithms have been developed to create a complete mesh representation of the target geometry of sufficient quality for further analysis only.

The literature regarding equidistant point generation shows that creating such a point set for complex geometries is extremely difficult although good approximations can be achieved. However, while these strategies provide good approximations for equidistance, they do not control uniformity of the point set.

The aim of this second extended literature review was to determine the meshing capabilities currently available and identify any suitable techniques that would enable the creation of uniform conformal RM textile structures. What has been identified is the need to develop such a surface meshing algorithm capable of creating a uniform and equidistant mesh that, when mapped to with RM textile link structures, would create a uniform and conformal RM textile. While the mapping of RM textile link structures to a non-uniform and therefore non-equidistant mesh is a viable alternative, this is not the scope of this research.

The next chapter in this thesis is therefore a discussion of how uniform and equidistant mesh structures may be achieved based on the findings of the second literature review.
Chapter 8: Discussion of the requirements for a new meshing algorithm

8.1 Introduction
The aim of this chapter is to discuss the findings from the literature and develop a possible new methodology for uniform and equidistant mesh generation.

8.2 Discussion of potential meshing techniques
The literature reviews of mesh generation techniques and the strategies employed to create equidistant point sets has shown that, firstly, no current meshing algorithm exists that has the capability to create a uniform and equidistant mesh structure for all curved surfaces, and secondly, that the creation of equidistant points over curved surfaces is an extremely complex problem where only good approximations exist.

The development of a methodology for the creation of conformal RM textiles requires a uniform and equidistant mesh structure for mapping; therefore it is essential that a new algorithm capable of producing such a mesh structure is investigated. The methodology also requires the repeated RM textile link structure to be designed first and the dimensional spacing or the required distance between individual link structures established during this phase. To create a conformal RM textile from the individual link structures, a surface geometry that the RM textile will eventually conform to must be represented as a surface mesh, which provides the required mapping location. Mapping the individual link structures to the mesh therefore ultimately leads to the generation of a complete conformal RM textile structure. This therefore requires the mapping mesh to be created with a nodal spacing that matches the dimensional value of the tessellation requirements of the RM textile link structure.
8.2.1 Energy minimisation
The literature has highlighted that an obvious solution for this meshing issue would be to utilise an energy minimisation method for equidistant point set creation over a curved surface, and to combine this with a Delaunay triangulation algorithm to create a triangular equidistant mesh structure. However, two issues become apparent when using a combination of these systems:

Firstly, energy minimisation, while capable of providing near equidistant points across curved surfaces, does not guarantee uniformity within the point set structure. Therefore, if a Delaunay triangulation algorithm were employed to connect the points created, in order to complete the mesh structure, the mesh structure would not be fully uniform and potential combinations of node to element ratios ranging from 4:1 to 6:1 become entirely possible.

Secondly, energy minimisation does not account for point spacing as a user-defined dimensional value. The methodology of energy minimisation requires a set number of points to be established first, that are then sequentially distributed across the geometry until the energy is minimised, therefore providing equidistance. This means that the actual point or node spacing value of the mesh can never be defined.

The fact that the potential utilisation of energy minimisation combined with Delaunay triangulation will never allow this component of the mesh to be accurately specified, or the uniformity guaranteed, renders their application redundant for this process.

8.2.2 Sphere packing
Reconsidering the sphere packing meshing techniques, high quality surface meshes were achieved by the packing of spheres with a user defined radius. The radii of the spheres are then only modified to minimise overlapping or gap formation between the packed spheres when generating a complete representation of the target surface geometry. A limitation of all current sphere packing algorithms is the dependency on the advancing front system discussed previously. All of the currently available sphere
packing algorithms utilise this system as the initial meshing process, where spheres are first generated at the boundary of the target geometry and then progressively packed inward until the geometry becomes completely packed with spheres. While the advancing front system is a robust method of ensuring the ultimate generation of a complete mesh representation of the target geometry, which is a prerequisite for FEA analysis, it immediately constrains the resulting structure of the mesh created.

Even when considering the meshing of a planar square surface; due to the initial advancing front meshing process, the resulting mesh structure will be created from spheres of different radii resulting in irregular node spacing and a non-uniform structure, as demonstrated in Figure 8-1, Figure 8-2 and Figure 8-3.

Figure 8-1: (Left) Planar surface for sphere pack meshing, (right) advancing front initial sphere placement

Figure 8-2: (Left) Full coverage of non uniform spheres, (right) mesh creation from sphere centres
While the surface mesh created may be considered high quality for FEA applications, as a large proportion of the elements are uniform and equidistant, the resulting mesh is unsuitable as a mapping mesh for the creation of RM textiles as it does not fully conform to these requirements. In fact, the utilisation of the advancing front system in current sphere packing algorithms means that a uniform mesh structure cannot be created for all potential surface geometries. The generation of a uniform mesh structures using this technique can only be achieved in the boundary of the surface to be mesh is a function of the packing sphere in dimension and shape.
8.2.3 Modified sphere packing

As discussed, the requirement of the mesh for RM textile link structure mapping is not identical to that of FEA analysis. While a complete mesh representation of the surface geometry that the eventual RM textile will conform to is desirable, the actual structure of the mesh is of far greater importance. Therefore, if the initial advancing front system of the meshing process is removed and the initial packing of spheres starts at a location central to the targeted surface, the resultant mesh structure generated becomes radically different.

Reconsidering the planar square surface meshed using the principles of current sphere packing algorithms as demonstrated in Figure 8-1, Figure 8-2 and Figure 8-3. If a modified sphere packing methodology is utilised, where the advancing front system is removed and the initial sphere placement is central to the surface geometry, the results of the modified sphere packing process can be seen in Figure 8-6, Figure 8-7 and Figure 8-8.

Figure 8-6: (Left) Planar square surface, (right) initial sphere placement central to surface

Figure 8-7: (Left) Full coverage of uniform spheres, (right) mesh creation from sphere centres
While the mesh created using the modified sphere packing technique is not a complete representation of the targeted geometry, it does however exhibit a completely uniform and equidistant structure as required by the methodology for creating RM textiles and provides the only possible technique at present for achieving such a mesh structure.

8.3 Conclusion

The previous discussions have highlighted the difficulty associated with the creation of a complete uniform and equidistant surface mesh representation of targeted surface geometries with user defined nodal spacing. The creation of such a mesh structure that completely describes any potentially specified surface geometry is currently impossible using available techniques highlighted in the literature review. The constraint of mesh creation for FEA, where the ultimate goal is a complete mesh representation of the underlying geometry, dictates that a fully uniform and equidistant mesh can never be generated with an equidistant nodal spacing specified by the user. The only plausible alternative is therefore to modify the constraints of the mesh generation procedure and allow a fully uniform and equidistant mesh to be created that does not completely describe the full structure of the underlying geometry. While this meshing situation may not be ideal, it may never be possible to create a complete uniform and equidistant mesh representation of all targeted surface geometries for a specified nodal spacing.

As the uniform structure and nodal spacing of the mesh is imperative for accurate mapping of RM textile link structures, these variables of the mesh structures are of far
greater importance than that of a complete mesh representation of the underlying surface geometry. As sphere packing from a central surface position shows the most promise for achieving a uniform and equidistant mesh structure, it is therefore the intention of this thesis and research to investigate a new meshing algorithm based on these principles for the creation of a uniform and equidistant mapping mesh. While equidistance could be achieved for the planar surface example highlighted, this may not be possible for all curved surface geometries. Therefore a compromise may have to be made and a tolerance introduced for the nodal spacing of the mapping mesh structures. However, while a small range in nodal spacing can be tolerated, the uniformity of the mesh is vital to maintain the tessellating structure of RM textiles and therefore, no deviation from uniformity can be tolerated.

The next chapter of this thesis therefore focuses on the investigation and design of such a meshing algorithm.
Chapter 9: Investigation into the generation of a suitable mapping mesh

9.1 Introduction

The findings of the previous chapter showed the potential for the implementation of a modified sphere packing approach for the generation of uniform and equidistant mesh structures, required for the manufacture conformal RM textiles. The meshing of planar surfaces by the modified sphere packing technique showed positive results and was indeed able to create a fully uniform and equidistant mesh. However, this result was totally expected as the meshing of planar surface is not a complex issue. Therefore, the aim of this chapter is to investigate the meshing of curved surface geometries and to investigate the methodology required by such surface geometries by a new sphere packing approach. In order to develop the rules of the new meshing algorithm it is first appropriate to work through a manual planar sphere packing meshing sequence. The rules generated from the meshing of a planar surface will then be modified as required for the accurate meshing of curved surface geometries.

9.2 Initial meshing strategy

All of the existing direct sphere packing algorithms utilise the advancing front technique as the initial sphere placement procedure. As previously demonstrated, the removal of this technique allows the generation of a uniform and equidistant mesh to be created if a central initial packing technique is utilised. However, the removal of the advancing front system means that the growth of the mesh structure is no longer constrained.

The advancing front system, as previously discussed, provides a complete sphere-packed boundary as a starting point. Therefore by continually advancing through the target geometry from the boundary, the surface geometry becomes fully ‘packed’ with spheres. If a central position is utilised instead of the geometry boundary, problems arise in the growth of the mesh structure, as highlighted in the following example.
Using a central position for the initial location of the first sphere dictates that the valid locations of the second sphere, tangential to the first, can be described by a circle twice the radius of the packing spheres. The circle describing the valid locations of the second sphere will be central to the initial sphere, as demonstrated in Figure 9-1.

Figure 9-1: Initial sphere placement

Therefore, the second sphere can occupy any point on the circle while still remaining in a valid packing location. The valid positioning of the second sphere therefore enables two further valid locations to be defined, as demonstrated in Figure 9-2.

Figure 9-2: Valid location creation
If the meshing algorithm is allowed to randomly choose which valid locations to use for the sequential packing of spheres, the potential exists for random mesh growth as demonstrated in Figure 9-3.

![Figure 9-3: Mesh growth](image)

While this is not a concern for the meshing of planar surfaces, when considering curved surfaces, the random growth will potentially lead to incorrect tessellation of the mesh structure as the opposing mesh directions eventually meet, as demonstrated in Figure 9-4.

![Initial direction of sphere packing](image)

![Second direction of sphere packing](image)

![Yellow packed spheres linking opposing packing directions, highlighting incorrect tessellation](image)

Figure 9-4: Inaccurate sphere packing from random mesh growth

Figure 9-4 demonstrates the tangential packing of spheres in two different directions. The spheres packed in each direction are uniformly spaced and tangentially packed to two further spheres, however, when sphere packing is initiated to link the two different packing directions, demonstrated by the yellow spheres in Figure 9-4, the meshing
technique fails and inaccurate non-tangential sphere packing occurs. While the mesh growth could be controlled with the use of direction vectors or the control of the actual packing sequence itself, an alternative is to use the sphere's intersection with the target surface to define the valid locations of a sequential sphere.

The intersecting-spheres methodology still requires the initial placement of the first sphere centre point central to and on the targeted surface. An intersection spline can then be created between the sphere and the surface. The spline created can then be used to define the location of the second sphere, in a similar manner to the original technique. However, locating a sphere to the intersection curve dictates that the two spheres will now intersect. If a second sphere-surface intersection spline is created from the second sphere, the two intersection splines created will also intersect, defining two new valid locations, also demonstrated in Figure 9-5.

![Figure 9-5: Sphere to sphere intersection showing both sphere to surface intersections](image)

If the meshing methodology is developed to position spheres at these locations, before calculating any new valid sites, the growth of the mesh structure can be fully controlled to grow consistently outward from the initially packed sphere.

The intersecting-sphere methodology represents a completely new approach to sphere-packing mesh generation, and has been developed by the author for the creation of uniform and equidistant mesh structures. Having established this novel methodology
technique fails and inaccurate non-tangential sphere packing occurs. While the mesh growth could be controlled with the use of direction vectors or the control of the actual packing sequence itself, an alternative is to use the sphere's intersection with the target surface to define the valid locations of a sequential sphere.

The intersecting-spheres methodology still requires the initial placement of the first sphere centre point central to and on the targeted surface. An intersection spline can then be created between the sphere and the surface. The spline created can then be used to define the location of the second sphere, in a similar manner to the original technique. However, locating a sphere to the intersection curve dictates that the two spheres will now intersect. If a second sphere-surface intersection spline is created from the second sphere, the two intersection splines created will also intersect, defining two new valid locations, also demonstrated in Figure 9-5.

![Figure 9-5: Sphere to sphere intersection showing both sphere to surface intersections](image)

If the meshing methodology is developed to position spheres at these locations, before calculating any new valid sites, the growth of the mesh structure can be fully controlled to grow consistently outward from the initially packed sphere.

The intersecting-sphere methodology represents a completely new approach to sphere-packing mesh generation, and has been developed by the author for the creation of uniform and equidistant mesh structures. Having established this novel methodology
and technique for ensuring consistent mesh growth, it is appropriate to manually work through the entire mesh sequence required to generate a planar surface mesh. Working through the sequence will enable the rules of the meshing methodology to be established, which can then be modified for the meshing of curved surface geometries.

9.3 Complete meshing of a planar surface
To investigate the rules of the new meshing algorithm a simple planar square surface was used as the target surface. The nodal spacing of the mesh was arbitrarily selected as 1 unit, therefore requiring the radius of the packing spheres to be 1 unit.

9.3.1 Initial sphere placement
The first stage of the intersecting-sphere methodology requires the initial packing sphere centre point to be constrained to the centre of the target surface, as demonstrated in Figure 9-6.

Figure 9-6: First sphere placement
The next stage of the intersecting-sphere technique is to calculate the intersection spline between the sphere and the surface. The intersection spline therefore represents every possible location where a second sphere centre can be positioned so the distance between the two respective centres is 1 unit, as demonstrated in Figure 9-7.
9.3.2 Second sphere placement

By constraining the second sphere centre point to the initial intersection spline and creating a second intersection spline as shown in Figure 9-8, the intersection of both the intersection splines created provides two further locations where a packing sphere may be accurately located, also shown in Figure 9-8.

9.3.3 Third and nth sphere placement

Repeating the sphere packing process, two further spheres can be mapped to the valid locations defined by the intersections of the splines created. The packing of spheres to these new locations enables two further sphere surface intersection splines to be created, as shown in Figure 9-9. Calculating every possible intersection between the
created splines defines four new valid sphere locations, including the sphere centre points of the initial and second packed sphere, also demonstrated in Figure 9-9. Therefore, it becomes essential that any location defined by the intersection of the splines created must be cross referenced against existing sphere centres and those that do match existing centres must be ignored.

Figure 9-9: Valid and invalid intersection locations

Continuing the packing of spheres to the valid intersection sites produces Figure 9-10. Again, the intersection of the splines created generates six new valid sites. However, existing sphere centres are again redefined and must be ignored. In addition, two valid sites are defined three times, by the splines created from new spheres intersecting with existing splines as shown in Figure 9-10. Therefore only unique sites become valid locations to remove any possibility of repetition.

Figure 9-10: Duplicated intersection sites
Continuing the sphere packing process at the unique valid sites produces Figure 9-11. At this stage of the mapping process, not only are existing sphere centres redefined but intersections between sphere surface splines are created that are outside the target surface boundary. As these new sites are outside the surface boundary they are defined as invalid, highlighted in Figure 9-11 as black dotted circles.

![Diagram showing valid and invalid intersection sites](image)

**Figure 9-11:** Invalid sites outside the surface boundary

Using the sphere packing rules developed so far, where sequential spheres are packed to valid sites only and intersection splines created, the packing of spheres continues until no further valid intersection can be defined, as demonstrated in Figure 9-12.

![Diagram showing target surface packed with spheres](image)

**Figure 9-12:** Target surface completely packed with spheres, only invalid intersections now exist
For clarity, all of the internal sphere intersections have been removed from Figure 9-12. When no further valid intersections exist or can be defined, no further spheres can be added to the surface, completing the sphere packing phase of the meshing technique. The centre points of the existing packed spheres therefore represent the uniform and equidistant nodes of the desired surface mesh. The next phase of the meshing process therefore becomes the triangulation of the nodes to create the elements of the mesh, completing the uniform and equidistant mesh structure.

9.4 Triangulation of nodes: Element creation

Figure 9-13 demonstrates seven packed spheres whose centre points are known and numbered 1-7. The triangulation process works by isolating a single known node or sphere centre, (for this example, number 1), highlighted in Figure 9-13 with a red circle.

![Figure 9-13: Numbered known sphere centres](image)

The algorithm must then identify which other nodes are a radius of the packing sphere (R) distance from this isolated point, in this example, numbers 2, 4 and 3, as demonstrated in Figure 9-14 with dotted blue lines.
By working sequentially through all the nodes identified, the algorithm must then determine if node 2 is a radius distance from node 4 (dotted red line). If this is the case, triangular element 1 can be created and recorded, as demonstrated in Figure 9-14 right.

Figure 9-14: schematic of sphere centres, (right) triangular element creation

The algorithm must then continue through the identified nodes by determining if node 2 is a radius length from 3. As the distance observed between 2 and 3 is greater than the radius, no new element is created. Continuing with the sequence, node 4 is then compared with node 3. As the distance between these nodes is equal to the radius, triangular element 2 can be created. As all of the nodes identified as being a distance (R) from the isolated sphere centre (1) have been mutually compared, the algorithm can then isolate one of the remaining known nodes and repeat the triangulation process.

As the algorithm isolates each known node individually, triangular elements will obviously be duplicated throughout the mesh. For example, from Figure 9-14, element 1 can be defined from node 1, 2 and 4. However, when isolating node (2) and repeating the triangulation process, the same element can be defined from nodes 2, 1 and 4. Therefore, before a triangular element can be fully verified, it must be compared to existing recorded elements. The algorithm must therefore have the capability to recognise that one triangular element generated from nodes 1, 2 and 4, is identical to a triangular element generated from nodes 2, 1 and 4, and to ignore the duplicate element. Repeating this process for every node centre therefore generates all of the
triangular elements required to complete the uniform and equidistant mesh structure, as demonstrated in Figure 9-15.

Figure 9-15: Complete uniform and equidistant mesh

Figure 9-15 shows the completion of the meshing procedure. To aid the reader's understanding of the rules that the meshing methodology complies with to create a planar uniform and equidistant mesh, a flow diagram of the sphere packing methodology has been included and demonstrated in Figure 9-16.
Create sphere with centre point constrained to surface geometry

Record sphere centre coordinates

Create sphere surface intersections spline

Record sphere surface intersection spline

Create sphere with centre point constrained to intersection spline

Enter loop stage

Create sphere surface intersections spline

Unique spline to spline intersection sites

Are sites existing sphere centres?

Yes

Create sphere with centre point constrained to valid site

Yes

(N) Valid sites

No valid sites available, END

Triangulation procedure

No

Invalid site

No

Are sites within the surface boundary?

Yes

Have (N) valid sites been used?

Yes

No

No

No valid sites available, END

Triangulation procedure

No

Invalid site

No

Are sites within the surface boundary?

Figure 9-16: Flow diagram of planar sphere packing methodology
Having established a new sphere packing methodology for the uniform and equidistant meshing of planar surfaces, the next section of this chapter investigates the meshing of curved surface geometries.

### 9.5 Meshing of curved surfaces

The aim of this section is to investigate the meshing of curved surfaces by the new intersecting-sphere packing approach investigated in the previous section. As the sphere packing approach was developed for the meshing of planar surfaces, modification will obviously be required to account for the curvature now present in the target geometry. Therefore, by working through the methodology investigated for planar surface meshing, the limitations and modifications necessary can easily be identified.

The target surface geometry used for this example is demonstrated in Figure 9-17. The radius of the packing spheres will again be 1 unit. The large curvature, in comparison to the packing sphere size has been chosen to further emphasise any limitations associated with meshing of curved geometries by the intersecting-sphere technique developed.

![Figure 9-17: Target curved surface geometry](image)

#### 9.5.1 Initial sphere placement

Using the same methodology set out in Figure 9-16, the first stage of the meshing process requires the initial sphere centre point to be constrained centrally to the surface of the target geometry, as demonstrated in Figure 9-18. Once achieved, a sphere-surface intersection spline can be created, also demonstrated in Figure 9-18.
Identical to the meshing of planar surfaces, the intersection spline created describes every possible location where a second packing sphere can be positioned, where the two respective sphere centre points are constrained to the surface and are a distance of one radius apart.

### 9.5.2 Second sphere placement

Locating a second sphere at any point on the intersection spline and generating a second intersection spline allows the intersection of the two splines to be calculated, generating two new valid packing sites, as demonstrated in Figure 9-19.
9.5.3 Third and nth sphere placement

At this stage, the meshing technique has entered the loop stage of the algorithm as described in the flow chart of Figure 9-16. As neither site created from the spline to spline intersection are existing sphere centres or outside the boundary of the surface, both points highlighted with red circles in Figure 9-19 are defined as valid. Therefore a sphere centre point can be constrained to the valid sites and subsequent sphere surface intersections splines created.

Packing spheres to both valid sites and creating the respective intersection splines then allows the methodology to calculate all of the spline to spline intersection sites. At this stage in the meshing process, eight sites are returned. As in the previous example however, four sites are existing sphere centres and therefore invalid and ignored (highlighted by blue circles in Figure 9-20). The remaining four sites which are not existing sphere centre or outside the boundary of the surface are therefore defined as valid and are highlighted in Figure 9-20 with red circles.

Continuing with the methodology developed, additional spheres can be constrained to the valid sites demonstrated in Figure 9-20 and intersection spline created. However, it is at this stage that the curvature observed in the target geometry has a direct effect on the accuracy of the developed meshing technique.
9.5.4 Clustered site creation

Figure 9-21 demonstrates the packing of spheres to the valid sites defined and highlighted in Figure 9-20.

Figure 9-21: Initial glance

From an initial glance at Figure 9-21, the meshing methodology appears to be working correctly. However, when a more detailed observation is undertaken, as demonstrated in Figure 9-22, it can be seen that the intersections of the splines created from packing of spheres to the valid sites shown in Figure 9-20, produce a cluster of three new valid sites where only one would have been expected in the ideal situation.

Figure 9-22: Perspective and close up of valid sites
This increase in the number of valid sites is due to the curvature observed in the geometry, causing the sphere-surface intersection spline to intersect at three different locations. When considering the meshing of planar surfaces, the intersection of the sphere-surfaces spline at this stage of the meshing process would all intersect at the same location and as only unique sites are defined as valid, the duplicated sites created from the splines intersecting would be ignored.

A closer examination of the three sites created is demonstrated in Figure 9-23.

![Figure 9-23: Points created in relation to sphere centres](image)

Figure 9-23 shows that the three sites created are defined as equidistant from at least two existing sphere centres. Where point 1 can be defined as equidistant from spheres A and B, point 2 can be defined as equidistant from spheres B and C and point 3 can be defined as equidistant from spheres A and C. While each site is defined as valid because it is not an existing sphere centre or outside the boundary of the target surface, the mapping of sequential spheres to these sites would ultimately lead to the meshing
process to fail, as inaccurate valid sites will be created from the resulting intersection of the sphere-surface splines created, as demonstrated in Figure 9-24.

Figure 9-24: Inaccurate site creation

Therefore it becomes essential to modify the intersecting-sphere meshing methodology to calculate a single valid site equidistant from all three packed sphere centres, in order that a further sphere can be constrained and the packing process continued. Closer inspection reveals that an equidistant point from all three sphere centres can be described by the generation of sphere to sphere intersection splines between all three spheres (ABC) as demonstrated in Figure 9-25.

Figure 9-25: Sphere to sphere intersection spline shown as black

The three sphere-sphere splines created intersect at two possible locations as demonstrated in Figure 9-26, as point 1 and 2.
However, while these two locations are equidistant from the centre of spheres (ABC), the points created are no longer constrained to the surface of the target geometry. Using the viewing plane shown in Figure 9-26, point 1 can be shown to be a perpendicular distance $X$ above the surface geometry, and point 2, a perpendicular distance of $Y$, both demonstrated in Figure 9-27.

This observation dictates that the intersecting-sphere meshing techniques cannot created a true conforming uniform and equidistant mesh representation of the underlying geometry, only an approximation of it. It was highlighted in the previous chapter that the use of centrally based sphere packing did not have the capability to create a complete
mesh representation of the underlying geometry, where the surface boundary remained unmeshed. This work has shown that a true mesh representation of curved surface geometry is also not achievable using the intersecting-sphere technique.

While deviation of the centre point of the sphere from the surface could allow a true equidistant point to be generated, the deviation also causes further concern for the remaining growth of the mesh structure. To accommodate the deviation from the underlying geometry in the mesh structure, further deviation would ultimately be required. To remain uniform and equidistant, the mesh structure would now continually deviate above and below the target geometry causing the mesh to crease. Also, as the meshing structure is grown, the potential of the mesh creasing to a point where it becomes unsuitable as a mapping mesh would be probable.

The work has demonstrated that the generation of a uniform and equidistant mesh is not possible whilst constraining the packing sphere centres to the surface of the target geometry. As deviation away from the target surface is not ideal and prevents a true conforming mesh representation from being achieved, a mesh structure of dimensionally variable sphere centre point spacing cannot be avoided.

As a true conforming mesh is more desirable over a deviating alternative, a tolerance-based mesh structure is proposed. Therefore it becomes necessary to calculate a single valid site that will allow a uniform mesh structure of variable centre point spacing to be generated that conforms to the targeted geometry.

9.5.5 Averaging method
Accepting that in order to create a true conforming mesh representation of the target geometry, variable nodal spacing becomes unavoidable for any specified surface, it now becomes necessary to triangulate the clustered sites demonstrated in Figure 9-28, to calculate a central location on the surface that can act as the new valid site for continued sphere packing.
As the intersections sites are known, it is possible to triangulate the three sites as demonstrated in Figure 9-29.

Once triangulated, the midpoints of the triangular element can be linked to an opposite corner of the triangular element. The intersection of the three lines created therefore describes the centre of the triangular element. This point can then be mapped back to the surface to provide a single valid site that allows the packing of a sphere to continue the meshing process.
9.5.6 Erroneous site creation

As an averaging method is now being implemented, the intersection of sphere-surface splines will now create erroneous sites within the existing mesh structure as demonstrated in Figure 9-30.

![Erroneous site created inside the red circles](image)

**Figure 9-30**: Erroneous sites created inside the red circles

The erroneous sites will be created and clustered around existing sphere centres. While the sites are within the boundary and not existing sphere centres, the methodology will define them as valid. However, packing additional spheres to these sites will ultimately cause re-packing of spheres to a proportion of the geometry that has already been successfully packed with spheres. It therefore becomes essential to identify these points and define them as invalid. The growth of the mesh structures defined by the intersection of sphere-surface splines means that new valid sites will always be created as a radius distance from at least two existing sphere centres. Therefore the erroneous sites created at a distance less than the specified radius from an existing sphere centre can be defined as invalid and ignored.

Having specified a system of calculating a single valid site from clustered sites and a system for removing erroneous sites generated from the implemented procedure, the
intersecting-sphere meshing technique can now complete the sphere packing of the target surface geometry, as demonstrated in Figure 9-31.

Figure 9-31: Target surface fully 'packed' with spheres

The last procedure of the intersecting-sphere meshing methodology is the triangulation of the sphere centres or nodes, necessary for the creation of mesh elements, completing the mesh structure, as investigated in section 9.4, and demonstrated in Figure 9-32.

Figure 9-32: (left) Triangulation of nodes/sphere centres, (right) final mesh structure
While the mesh structure created using the newly developed intersecting-sphere meshing technique demonstrated in Figure 9-32 is not completely equidistant, the structure is uniform and therefore becomes suitable as a mapping mesh.

9.6 Key mathematical principles of the intersecting-sphere meshing algorithm

In order to test the capabilities of the new meshing methodology investigated, it is necessary to code the methodology into an experimental meshing system. Producing an experimental meshing system therefore enables meaningful results to be efficiently generated that can then be examined to ascertain the capabilities of the intersecting-sphere technique developed. To achieve this aim, it was decided that the rules of the methodology investigated in this chapter should be coded into a complete meshing system using the Matlab software [177].

To code the experimental algorithm, it therefore becomes necessary to define the key mathematical principles utilised by the new algorithm. These key principles are:

- Target surface creation
- Sphere to surface intersection
- Spline to spline intersection

As the aim of the experimental testing is an investigation of the capabilities of the new algorithm and not the generation of a fully functional standalone meshing tool, the key mathematical principles will investigated accordingly. While the generation of standalone meshing tool may require the use of NURB surfaces for example, the testing of the algorithm will be based around analytical surfaces that can be readily described by a single surface equation.
9.7 Target Surface creation: meshing domain

For the experimental testing of the developed intersecting-sphere algorithm, an obvious component required is a target surface geometry. In order to test the capabilities of the algorithm, a system of surface generation is required that will allow the creation of a range of surfaces with variable curvature and complexity. As the most probable application at present for conformal RM textiles is that of high performance and smart clothing, the level of curvature that the new algorithm would be expected to successfully mesh can be defined as the curvature and complexity observed in the human form. While a complete human form as a surface geometry cannot be described by a single equation, an obvious solution is the utilisation of quadratic surface patches. Using this system, the curvature and complexity of various areas of the human form can be replicated. The use of quadratic surfaces therefore enables the generation of a range of surfaces where the curvature and complexity can be easily modified and an equation of the surface can be developed.

9.7.1 Development of surface equation

A quadratic surface is a surface whose mathematical representation is given by a quadratic polynomial:

\[ f(x, y, z) = 0 \]  

Equation 9.1

Where the algebraic representation is given by the equation:

\[ Ax^2 + By^2 + Cz^2 + 2Dxy + 2Eyz + 2Fzx + 2Gx + 2Hy + 2Jz + K = 0 \]  

Equation 9.2

Where \((x, y, z)\) are coordinates in Cartesian space

A surface patch can be created in CAD systems from the creation of two splines and a sweep function, as demonstrated in Figure 9-33. The sweep function is a standard CAD operation that allows the creation of geometry by utilising a profile region and a path
curve, spline, geometry edge or line. In the case of surface geometry creation, the profile of the surface geometry will be a spline and the path will also be a spline.

In the case of surface geometry creation, the profile of the surface geometry will be a spline and the path will also be a spline.

\[
Z(\text{surface}) = F(X,Y)
\]  
Equation 9.3

Therefore, using the same system that is incorporated in conventional CAD systems, a surface patch can be described as the function of two splines, one in the XZ domain and the second in the YZ domain. Where the surface \( Z \) will be a function of \( X \) and \( Y \):

However, spline generation within CAD is not the result of a function of two Cartesian axes as a standard graph curve would be and is therefore not explicit. Splines within CAD are the result of control points and blending operators, for example those used in NURB splines (Non-uniform rational B splines) and are therefore implicit, as demonstrated in Figure 9-34.
As no explicit definitions of the splines are available in conventional CAD systems, no simple method of developing the surface equation exists. In order to develop the surface equation, an explicit definition of both splines is required. In order to develop the explicit definition of the splines it therefore becomes necessary to fit a curve to the control points of the spline by Lagrange interpolation [178].

### 9.7.2 Curve fitting – Lagrange interpolation

The principle of Lagrange interpolation is that a function $f(x)$, whose values are given as a collection of points, is assumed to be approximately represented by a polynomial $P(x)$ that passes through each and every point [178]. The polynomial is called the interpolation polynomial and it is of one degree less than the number of points given. For two data points the interpolating polynomial is taken to be linear, for three the polynomial is quadratic and for four points the polynomial is cubic and so on [178].

LaGrange interpolation formulae [178] can be represented as:

$$P_x = \sum \frac{(x - x_{n+1})}{(x_{n} - x_{n+1})} f(x_{n})$$

Equation 9.4
Where:

\[ P_x = \text{Polynomial equation} \]
\[ x = \text{X coordinate} \]
\[ x_n = \text{X axis recorded coordinate} \]
\[ f(x_n) = \text{Y axis recorded coordinate at X axis} \]

When considering a surface generated in CAD, created by a sweep function using the two identical splines XZ and YZ, as demonstrated in Figure 9-35, the equation describing the surface can be generated by calculating the polynomial of each spline respectively and combining them into one equation.

Considering each spline individually, Figure 9-36 demonstrates the XZ spline used in the construction of the surface shown in Figure 9-35, and the recorded (z,x) control points used in the construction of the spline in CAD.

Figure 9-35: Surface generated in CAD, (right) YZ and XZ splines used for its construction
The five sets of \((x,z)\) coordinates recorded in Figure 9-36, can be defined as terms of the Lagrange interpolation equation as demonstrated in Table 9.1.

<table>
<thead>
<tr>
<th>Control point</th>
<th>(x) = (x) coordinate</th>
<th>(f(x)) = (y) coordinate</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_0)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(x_1)</td>
<td>25</td>
<td>50</td>
</tr>
<tr>
<td>(x_2)</td>
<td>50</td>
<td>25</td>
</tr>
<tr>
<td>(x_3)</td>
<td>75</td>
<td>50</td>
</tr>
<tr>
<td>(x_4)</td>
<td>100</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 9.1: Lagrange Interpolation terms for 5 control points

Assuming LaGrange interpolation:

\[
P_x = \sum \frac{(x-x_{a+1})}{(x_{a}-x_{a+1})} f(x_a)
\]

Equation 9.4

Expanding LaGrange interpolation for five control points:
Substituting values from a Table 9.1

\[ P_x = \frac{(x-x_0)(x-x_2)(x-x_3)(x-x_4)}{(x_0-x_2)(x_0-x_3)(x_0-x_4)} f(x_0) + \frac{(x-x_1)(x-x_2)(x-x_3)(x-x_4)}{(x_1-x_2)(x_1-x_3)(x_1-x_4)} f(x_1) + \frac{(x-x_0)(x-x_1)(x-x_3)(x-x_4)}{(x_0-x_1)(x_0-x_3)(x_0-x_4)} f(x_2) + \frac{(x-x_0)(x-x_1)(x-x_2)(x-x_4)}{(x_0-x_1)(x_0-x_2)(x_0-x_4)} f(x_3) + \frac{(x-x_0)(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} f(x_4) \]

Simplifying Equation:

\[ P_x = \frac{(x-25)(x-50)(x-75)(x-100)}{(-25)(-50)(-75)(-100)} f(0) + \frac{(x)(x-50)(x-75)(x-100)}{(25)(-25)(-75)(-100)} f(50) + \frac{(x)(x-25)(x-50)(x-75)}{(50)(-25)(-75)(-75)} f(25) + \frac{(x)(x-25)(x-50)(x-75)}{(75)(-25)(25)(-25)} f(50) + \frac{(x)(x-25)(x-50)(x-75)}{(100)(75)(25)(25)} f(0) \]

Collecting all the terms produces:

\[ P_x = -0.00002667 x^4 + 0.005333333 x^3 - 0.343333333 x^2 + 7.666666667 x \quad \text{Equation 9.5} \]

Equation 10.5 therefore represents the polynomial of the XZ spline used in the construction of the surface demonstrated in Figure 9-35. However, as the spline was generated in the X and Z axes, a more appropriate representation is:

\[ P_z = -0.00002667 x^4 + 0.005333333 x^3 - 0.343333333 x^2 + 7.666666667 x \quad \text{Equation 9.6} \]

Where the height of the spline in the Z axis can be calculated from any given x coordinate specified.
As the YZ spline used in the construction of the surface demonstrated in Figure 9-35, was identical to the XZ spline, Equation 9.6 can be simply modified to include the Y axis as follows:

\[ P_z = -0.00002667y^4 + 0.0053333333y^3 - 0.34333333y^2 + 7.66666667y \]  

Equation 9.7

However, if the two splines used in the creation of a surface are not identical, a suitable polynomial equation must be created for each spline respectively.

Plotting Equation 9.7 for x values 0-100 produces the curve shown in Figure 9-37. The curve generated from the polynomial shows excellent correlation with the spline created in CAD, where any distortion is due to respective scales.

![Graph showing the polynomial curve.](image)

**Figure 9-37:** Polynomial curve plotted for x = 0-100

Combining Equation 9.6 and 9.7 and collecting the similar terms produces the quadratic equation of the surface demonstrated in Figure 9-37, as follows:

\[ Y \text{ values created from fourth order polynomial } y = 0.00002867x^4 + 0.00533333x^3 - 0.34333333x^2 + 7.66666667x \]
\[ P_z = -0.00002667x^4 + 0.005333333x^3 - 0.34333333x^2 + 7.6666666667x \]  
\text{Equation 9.6}

\[ P_z = -0.00002667y^4 + 0.005333333y^3 - 0.34333333y^2 + 7.6666666667y \]  
\text{Equation 9.7}

\[ Z_{\text{surface}} = -0.00002667x^4 - 0.00002667y^4 + 0.005333333x^3 + 0.005333333y^3 \]
\[ -0.34333333x^2 - 0.34333333y^2 + 7.6666666667x + 7.6666666667y \]  
\text{Equation 9.8}

Equation 9.8 therefore fully describes the surface geometry and can be used to calculate the z coordinate of the surface for any given (x,y) coordinate. Plotting Equation 9.8 for x and y values 0 - 100 produces the 3D surface demonstrated in Figure 9-38.

![3D Surface Plot](image)

Figure 9-38: Surface plot of developed surface Equation 9.8

Using the methodology developed, it becomes possible to establish a suitable (xy) domain (dimensional limit) and set control points for both x and y axes. The creation of the quadratic surface can then be defined by the z coordinates corresponding to the control points identified and an equation of the surface generated.
9.8 Sphere to surface intersection

The next principle of the intersecting-sphere algorithm is the capability to calculate the intersection of the packing sphere with the targeted surface geometry. The methodology for achieving this type of operation is dependent on the description used for the target surface and packing spheres. As an analytical quadratic description is being utilised for the description of the experimental test surface geometries, it is therefore appropriate to use an analytical description of the packing sphere and to develop a methodology for calculating the intersection accordingly.

9.8.1 Description of Sphere

The mathematical description of a sphere when using Cartesian coordinates and centred on the origin of the system is given by the equation:

\[ x'^2 + y'^2 + z'^2 = r'^2 \]

Equation 9.9

Where

\( r \) = radius
\( x \) = x coordinate
\( y \) = y coordinate
\( z \) = z coordinate

When calculating a sphere not centred on the origin, the equation becomes:

\[
(x - x_o)^2 + (y - y_o)^2 + (z - z_o)^2 = r^2 \]

Equation 9.10

Where

\( x_0 \) = x axis sphere centre coordinate
\( y_0 \) = y axis sphere centre coordinate
\( z_0 \) = z axis sphere centre coordinate

While Equations 9.9 and 9.10 both provide exact solutions for the surface of a sphere, they do not provide an efficient description of parameterised points on the surface of the sphere. An alternative system describing the surface of a sphere that does efficiently calculate parameterised points is the use of spherical coordinates. This system
calculates points on the surface of the sphere using the longitude and colatitude angles as demonstrated in Figure 9-38.

Figure 9-39: Sphere representation

The equations for Cartesian coordinates on the surface of the sphere are described individually by three equations:

\[
\begin{align*}
    x &= r \cos \theta \sin \alpha \\
    y &= r \sin \theta \sin \alpha \\
    z &= r \cos \alpha
\end{align*}
\]

Where

\[
\begin{align*}
    r &= \text{radius} \\
    x &= \text{x coordinate} \\
    y &= \text{y coordinate} \\
    z &= \text{z coordinate} \\
    \theta &= 0 - 360^\circ \\
    \alpha &= 0 - 180^\circ
\end{align*}
\]

When defining a sphere that is not centred on the origin using spherical coordinates, the equations become:

\[
x = x_0 + r \cos \theta \sin \alpha
\]
\[ y = y_0 + r \sin \theta \sin \alpha \]  
Equation 9.15

\[ z = z_0 + r \cos \alpha \]  
Equation 9.16

Where:

- \( x_0 \) = x axis sphere centre coordinate
- \( y_0 \) = y axis sphere centre coordinate
- \( z_0 \) = z axis sphere centre coordinate

Using the spherical coordinate system it becomes possible to calculate the intersection of a sphere with a target surface geometry when the centre point of the sphere is constrained to the surface.

### 9.8.2 Sphere-surface intersection

Consider a packing sphere of a specified radius constrained by its centre point to a surface geometry with a known equation, as demonstrated in Figure 9-40.

![Figure 9-40: Sphere-surface intersection](image)

Using the spherical coordinates, it becomes possible to calculate a hemispherical curve of surface points for every longitudinal coordinate as demonstrated in Figure 9-41.
This can be achieved by making the longitudinal angle constant while calculating the Cartesian coordinates for the entire range of the colatitude angle.

Where:
\[ \theta = 0^\circ \]
\[ \alpha = 0^\circ - 180^\circ \]

Substituting these values into equations 9.14, 9.15 and 9.16 produces:

\[ x = x_0 + r \sin(0^\circ - 180^\circ) \]  \hspace{1cm} Equation 9.17
\[ y = y_0 \]  \hspace{1cm} Equation 9.18
\[ z = z_0 + r \cos(0^\circ - 180^\circ) \]  \hspace{1cm} Equation 9.19

Also, the equation of the surface will be known and in the form:
\[ Z_{\text{surface}} = f(x, y) \]  \hspace{1cm} Equation 9.3

Calculating the Cartesian coordinates of the sphere for a constant longitude angle and the range of colatitude angles iteratively, for example every 10°, the x and y coordinates
calculated can be substituted into equation 9.3, producing the z coordinate on the target surface at this specified x and y location, as demonstrated in Figure 9-42:

\[
Z_{\text{surface}} = f((x_0 + r \sin(0^\circ - 180^\circ)), y_0) \tag{Equation 9.20}
\]

Figure 9-42: Z coordinate of sphere and surface for \( \theta = 0^\circ \) and \( \alpha = 0^\circ \)

Comparing Equation 9.19 to Equation 9.3, initially (at alpha = 0) the z coordinate of the sphere will be greater than the recorded z coordinate of the surface for the same global x and y coordinates of the Cartesian system.

Where:

\[
z_{\text{sphere}} > Z_{\text{surface}} \tag{Equation 9.21}
\]

Continuing the iterative process, it becomes possible to isolate the colatitude angle that produces:

\[
z_{\text{sphere}} < Z_{\text{surface}} \tag{Equation 9.22}
\]
As an iterative approach is being utilised, the exact intersection between the isolated series of sphere coordinates and the target surface is between a colatitude angle range of $\alpha = 90^\circ - 100^\circ$. Reemploying the iterative technique in reverse, it now becomes possible to calculate the Cartesian coordinates of the sphere from $\alpha = 100^\circ - 90^\circ$ in 1 degree increments until a colatitude angle that satisfies Equation 9.21 is calculated:

$$z_{sphere} > Z_{surface} \quad \text{Equation 9.21}$$

This therefore isolates a colatitude range of one degree. Repeating the iterative process it becomes possible to continually reduce the intersect colatitude angle range until an accurate intersection can be calculated, where:

$$z_{sphere} = Z_{surface} \quad \text{Equation 9.22}$$

When Equation 9.22 is true, the constant longitudinal angle and colatitude angle calculated can be substituted back into equations 9.17, 9.18 and 9.19 to calculate the exact intersection of the sphere with the surface for a particular longitudinal angle.
Repeating this process for the whole series of longitudinal angles $(0-360^\circ)$ in one degree increments therefore produces 360 intersection points that can then be used as control points for the creation of a 3D closed spline that represents the entire sphere-surface intersection for a specified sphere centre point.

### 9.9 Spline to spline intersection

The spline to spline intersection is the final key technique of the intersecting-sphere algorithm, as the intersection sites calculated between splines are used as packing locations for additional spheres. The intersection of 3D splines is a complex problem, often associated with CAD systems, where spline ambiguity can cause poor accuracy. However, for this application, the creation of splines using a combination of the spherical coordinates and the equation of the target surface dictates that the splines created are always constrained to the surface of the target geometry, as demonstrated in Figure 9-44.

![Sphere-surface intersection splines constrained to the target surface geometry, black circles](image1)

**Figure 9-44:** Sphere-surface spline showing intersection sites

Therefore, if the splines appear to overlap, as demonstrated in Figure 9-44, it can be assumed that they intersect at two possible locations, also demonstrated in Figure 9-44.

This assumption allows a simple method to be implemented that accurately calculates the intersection of the two splines. As the splines are assumed to intersect, the 3D splines can be projected on to a suitable mapping plane; therefore reducing the spline to a 2D curve as demonstrated in Figure 9-45.
Figure 9-45: 3D splines projected to mapping plane creating 2D curves

Projecting the 3D spline control points to the mapping plane can easily be achieved by removing the relevant Cartesian axis coordinate. Therefore, when mapping to a 2D (xy) plane, as demonstrated in Figure 9-45, the z coordinates of the spline are removed producing a 2D curve.

Once the complexity of the splines is reduced to 2D curves, the intersection of the 2D curves can be calculated using simultaneous equations or interpolation. Once the intersection sites are calculated in (xy) coordinates, the respective coordinates can be substituted into the target surface equation to calculate the z coordinate point on the surface of the target geometry where the two splines intersect.

Having addressed all the key principles required by the intersecting-sphere technique, the next research task is the actual coding of the methodology investigated to produce the experimental meshing system. The next chapter in this thesis will therefore be the experimental testing of the created meshing system.
Chapter 10: Experimental testing of intersecting-sphere meshing

10.1 Introduction

The aim of this chapter is to present the experimental testing of the intersecting-sphere meshing methodology investigated in Chapter 9. In order to test the capabilities of the new meshing algorithm, it was necessary to code the rules of the algorithm into a working experimental meshing system to allow meaningful results to be efficiently generated. To achieve this, the rules investigated in the previous chapter were coded into a complete meshing system using the Matlab software. The actual coding of the algorithm was a significant task, requiring 6 months of intensive coding and debugging to ensure the algorithm worked as required. The full code of the created algorithm can be seen in Appendix A.

10.2 Experimental meshing

To test the meshing capabilities of the intersecting-sphere technique, a range of quadratic surfaces, as discussed in the previous chapter, were defined and used as sample target surface geometries. The sample surface used gradually increased in surface curvature and therefore complexity to fully establish the capabilities of the new meshing system. The surfaces chosen were designed to reflect the level of curvature that can be expected from various areas of a human form. For each test surface used, the packing sphere radius was continually reduced and mesh structures recorded and analysed. The mesh structures were then statistically examined for equidistance and visually examined for uniformity. As complete equidistance is highly improbable, standard deviation of the recorded nodal spacing was used to show the spread of the nodal spacing and therefore ascertain any apparent trend towards equidistance with decreasing packing sphere radius.
10.3 Sample surfaces

The surface geometries used to test the capabilities of the intersecting-sphere technique were all quadratic surface patches, as described in Chapter 9. The use of quadratic surface patches enabled the creation of a series of test geometries where the curvature and complexity could be modified and a single algebraic equation of the surface determined.

10.3.1 Surface dimensions

In order to generate efficient and meaningful results, the dimensions of the sample surface were restricted to a maximum arbitrary dimension value of 50 units, in both the apparent X and Y axes. This resulted in the creation of a 50 by 50 unit surface patch. The spline control points (X0 -X5) and (Y0 -Y5) were set at 0, 12.5, 25, 37.5 and 50 units, as demonstrated in Figure 10-1. The Z unit dimensions were set at these control points for the creation of the respective splines, XZ and YZ, for the generation of the surface equation using polynomial interpolation as discussed previously. While the surface patches provide relatively small meshing domains, the aim of this experimental chapter was to test the intersecting-sphere meshing technique’s capability to mesh curved surface geometries and not the creation of mapping meshes suitable for generating whole conformal RM textiles.

\[ Z \text{ axis} \]
\[ X_0 & Y_0 = 0 \text{ units} \]
\[ X_1 & Y_1 = 12.5 \text{ units} \]
\[ X_2 & Y_2 = 25 \text{ units} \]
\[ X_3 & Y_3 = 37.5 \text{ units} \]
\[ X_4 & Y_4 = 50 \text{ units} \]

Figure 10-1: Spline creation for quadratic surfaces, showing control point locations
10.3.2 Sample surface geometries

The following surface types were selected for investigation:

Planar surface square = Surface 1, demonstrated in Figure 10-2
Half cylindrical surface = Surface 2, demonstrated in Figure 10-3
Parabolic dome surface = Surface 3, demonstrated in Figure 10-4
Saddle surface = Surface 4, demonstrated in Figure 10-5
Peaked surface = Surface 5, demonstrated in Figure 10-6

Surface 1

X dimension : 50
Y dimension : 50
Z dimension at
X axis control points : [0 0 0 0 0]
Z dimension at
Y axis control points : [0 0 0 0 0]
Control surface 1

Figure 10-2: Surface 1 and construction data

Surface 2

X dimension : 50
Y dimension : 50
Z dimension at
X axis control points : [0 21.65 25 21.65 0]
Z dimension at
Y axis control points : [0 0 0 0 0]
Surface curvature approximating the arm or leg of the human form

Figure 10-3: Surface 2 and construction data
Surface 3

Figure 10-4: Surface 3 and construction data

Surface 4

Figure 10-5: Surface 4 and construction data

Surface 5

Figure 10-6: Surface 5 and construction data

X dimension : 50
Y dimension : 50
Z dimension at
X axis control points : [0 5.35 7 5.35 0]
Z dimension at
Y axis control points : [0 -5.35 -7 -5.35 0]
Surface curvature approximating the breast, shoulder or knee of a human form

X dimension : 50
Y dimension : 50
Z dimension at
X axis control points : [0 5.35 7 5.35 0]
Y axis control points : [0 -5.35 -7 -5.35 0]
Surface curvature approximating the armpit or back of the knee of the human form

X dimension : 50
Y dimension : 50
Z dimension at
X axis control points : [0 7.5 5 7.5 0]
Z dimension at
Y axis control points : [0 7.5 5 7.5 0]
Control surface 2
Surface 1 has been included to act as the control geometry. As the results of the meshing process on this surface are predicted to be completely uniform and equidistant, this result can be used as a benchmark to compare the results of the meshing of curved surface geometries. Surface 5 has been included as it failed during the meshing process to assess the limitations of the new intersecting-sphere technique.

10.4 Experimental procedure

For each of the test surface geometries, mesh structures were generated using a range of decreasing radii. These ranged from a maximum of 10 dimensional units to 1 dimensional unit. A total of six mesh structures will therefore be created for each sample surface as follows:

- Mesh structure 1 = 10 unit packing sphere
- Mesh structure 2 = 5 unit packing sphere
- Mesh structure 3 = 4 unit packing sphere
- Mesh structure 4 = 3 unit packing sphere
- Mesh structure 5 = 2 unit packing sphere
- Mesh structure 6 = 1 unit packing sphere

10.4.1 Analysis of mesh structures

- Equidistance

For each mesh structure created for a particular sample surface geometry, the target element dimension is shown along with the recorded maximum, average and minimum nodal spacing dimension generated. To understand the range of nodal spacing dimensions produced, the standard deviation was also calculated. The standard deviation from the mean provides a simple method to indicate the degree of dispersion of recorded variables in a dataset [179]. Therefore, the standard deviation for this purpose will show the dispersion of the recorded nodal spacing dimensions. The standard deviation can be calculated as follows:

\[ \sigma = \sqrt{\frac{\sum (x - \bar{x})^2}{n}} \]  

Equation 10.1
Where:

\[
\begin{align*}
\sigma & = \text{standard deviation} \\
x & = \text{value recorded} \\
\bar{x} & = \text{mean of all values} \\
n & = \text{number of values}
\end{align*}
\]

To further understand the dispersion of the nodal spacing generated, a histogram for each mesh structure was also generated. A histogram is a frequency bar chart of the recorded values. The histogram provides a visual clarification of the dispersion of the recorded nodal spacing and therefore a more meaningful technique for understanding the dispersion of the nodal spacing values recorded.

• **Uniformity**

The uniformity of the mesh structure is dependent on the node to beam element ratio being 6:1 throughout the surface, except at the boundaries, and can be measured as either true or false. Therefore a visual observation of the mesh structure can be used to determine this variable. Where uniformity is not present, false will be recorded in the mesh data and the area of non-uniform mesh structure will be highlighted in the corresponding mesh figure.

• **Comparison with commercial meshing solutions**

In order to ascertain the benefits of the intersecting-sphere methodology over existing meshing methods, the meshes generated for each of the surfaces (with the exception of surface 5) will also be compared with two existing software solutions (Gambit and MSC Patran), both in terms of the uniformity of the mesh structure generated and the level of equidistance achieved within these structures.

The following sections provide images of the mesh structures generated for each surface at each target nodal spacing, alongside details and 0.05 unit histograms of the actual nodal spacings generated, as well as the comparisons with commercial software solutions. For each surface, the results for the 1 unit mesh are displayed; the results for the remainder of the meshes can be found in Appendix B, where 0.1 unit histograms are presented.
10.5 Results: Surface 1

Figure 10-7: Mesh structure generated from 1 unit sphere

Figure 10-8: Histogram of mesh 6 surface 1

Target nodal spacing = 1
Max nodal spacing = 1
Min nodal spacing = 1
Mean nodal spacing = 1
Standard deviation = 0
Uniformity = True

Recorded nodal spacing

Frequency of recorded nodal spacing

155
10.5.1 Discussion of results for surface 1

- Uniformity and mesh coverage

Figure 10-7 demonstrates the mesh structure generated when using a sphere radius of 1 unit. The newly developed intersecting-sphere meshing technique generated a uniform mesh structure with a constant node to beam element ratio of 6:1 in the centre of the mesh and 2, 3 or 4:1 at the boundaries as expected. The mesh structures obtained for the other sphere radius values (which can be found in Appendix B) were also all found to meet this criterion. The images also demonstrated that the overall coverage of the target surface geometry by the mesh structure was increased with a reduction in sphere radius.

- Equidistance

Figure 10-8 demonstrates a histogram of the nodal spacing, the target nodal spacing and the maximum, minimum, mean and standard deviation of the values recorded. This, and the other histograms for each of the mesh structures generated showed no deviation from the target dimension values, nor did the maximum, minimum and mean of the recorded nodal dimension values. The standard deviation of the nodal spacing for each specified sphere radius was also recorded as zero. As no deviation from the target nodal spacing was recorded, an equidistant mesh structure was generated at each specified sphere radius. Table 10.1 documents the standard deviation recorded for each mesh structure generated using the intersecting-sphere meshing algorithm.

<table>
<thead>
<tr>
<th>Surface 1</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh 1: 10 unit radius sphere</td>
<td>0</td>
</tr>
<tr>
<td>Mesh 2: 5 unit radius sphere</td>
<td>0</td>
</tr>
<tr>
<td>Mesh 3: 4 unit radius sphere</td>
<td>0</td>
</tr>
<tr>
<td>Mesh 4: 3 unit radius sphere</td>
<td>0</td>
</tr>
<tr>
<td>Mesh 5: 2 unit radius sphere</td>
<td>0</td>
</tr>
<tr>
<td>Mesh 6: 1 unit radius sphere</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 10.1: Standard deviations recorded from surface 1

Plotting the recorded standard deviation values from Table 10.1 against the relevant sphere radius produces Figure 10-9.
The trend line generated from the plotted data suggests that over the range of sphere radii used, any sphere radius value will produce a standard deviation of zero and therefore a fully equidistant mesh structure.

- **Comparison with commercial meshing solutions**

As a comparison, surface 1 was then meshed with a target nodal spacing of 5 units using two commercially available FEA meshing software solutions; Fluent's pre-processor Gambit and MSC Patran. The results of the meshing process using both systems are demonstrated individually in Figure 10-10 and Figure 10-11. For each mesh structure generated, the max, min, mean and standard deviation of the nodal spacing is documented in addition to a histogram demonstrating the dispersion of the nodal spacing.
Figure 10-12 demonstrates the capability of the intersecting-sphere technique when meshing the same target geometry using the same target nodal spacing and again shows the same mesh data as the previous two figures.

While both commercial meshing systems were able to create a complete mesh representation of surface 1, they were unable to generate a uniform mesh or equidistant structure as was generated by the intersecting-sphere meshing technique.

As all three histograms demonstrated in Figure 10-10, Figure 10-11 and Figure 10-12 are plotted to the same scale, it can be easily seen that the intersecting-sphere generated mesh is an improvement over the commercially available systems when considering uniformity and the trend towards equidistant mesh structures.
Figure 10-10: Gambit generated 5 unit mesh

Figure 10-11: Patran generated 5 unit mesh

Figure 10-12: Intersecting-sphere generated 5 unit mesh

Max nodal spacing = 6.2000
Min nodal spacing = 3.7054
Mean nodal spacing = 5.0579
Standard deviation = 0.5185

Max nodal spacing = 7.2375
Min nodal spacing = 3.9535
Mean nodal spacing = 5.0324
Standard deviation = 0.5950

Max nodal spacing = 5
Min nodal spacing = 5
Mean nodal spacing = 5
Standard deviation = 0
10.6 Results: Surface 2

Figure 10-13: Mesh structure generated from 1 unit sphere

Target nodal spacing = 1
Max nodal spacing = 1.0141
Min nodal spacing = 0.98479
Mean nodal spacing = 0.99978
Standard deviation = 8.2717e-04
Uniformity = True

Figure 10-14: Histogram of mesh 6 surface 2
10.6.1 Discussion of results for Surface 2

- Uniformity and mesh coverage

Figure 10-7 demonstrates the mesh structure generated for a sphere radius of one unit. As before, the results for other sphere radii can be found in Appendix B. For every specified sphere radius, the intersecting-sphere meshing technique again generated a uniform mesh structure with a constant node to beam element ratio of 6:1 in the centre of the mesh and 2, 3 or 4:1 at the boundaries. These results again demonstrate that mesh coverage of the target surface geometry was increased with a reduction in sphere radius.

- Equidistance

Figure 10-8 demonstrates a histogram of the nodal spacing, the target nodal spacing and the maximum, minimum, mean and standard deviation of the values recorded, with the remainder of the graphs located in Appendix B. The results for mesh 1 (with a target spacing of 10 units) showed a deviation from the target value of +/- 0.1 units. However, the remaining mesh structures, (2-6) showed no recorded deviation greater than +/- 0.1 units. As deviation in nodal spacing has been recorded for all mesh structures generated for surface 2, they all are classified as technically not equidistant. However, with the reduction in sphere radius, the deviation recorded decreased to a point where the meshes generated can be deemed equidistant for practical purposes. Table 10.2 documents the standard deviation recorded for each mesh structure generated using the intersecting-sphere meshing algorithm.

<table>
<thead>
<tr>
<th>Surface 2</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh 1: 10unit radius sphere</td>
<td>0.0187</td>
</tr>
<tr>
<td>Mesh 2: 5unit radius sphere</td>
<td>0.0051</td>
</tr>
<tr>
<td>Mesh 3: 4unit radius sphere</td>
<td>0.0034</td>
</tr>
<tr>
<td>Mesh 4: 3unit radius sphere</td>
<td>0.0019</td>
</tr>
<tr>
<td>Mesh 5: 2unit radius sphere</td>
<td>0.0015</td>
</tr>
<tr>
<td>Mesh 6: 1unit radius sphere</td>
<td>8.2717e-004</td>
</tr>
</tbody>
</table>

Table 10.2: Standard deviation recorded from surface 2
Plotting the recorded standard deviation values from Table 10.2 against the relevant sphere radius produces Figure 10-15.

![Standard Deviation against sphere radius](image)

Figure 10-15: Standard deviation against sphere radius

The trend line generated from the plotted data suggests that over the range of sphere radii used, a reduction of sphere radius produces a reduction in standard deviation. This therefore suggests that mesh generation at decreasing radius values produces a trend towards fully equidistant mesh structures.

- **Comparison with commercial meshing solutions**

As for surface 1, surface 2 was again meshed with a target nodal spacing of 5 units using both FEA meshing software solutions used previously. The results of the meshing process using both systems are demonstrated individually in Figure 10-16 and Figure 10-17. For each mesh structure generated, the maximum, minimum, mean and
standard deviation of the nodal spacing is again documented, in addition to a histogram demonstrating the dispersion of the nodal spacing.

Figure 10-18 demonstrates the capability of the intersecting-sphere technique when meshing the same target geometry using the same target nodal spacing and again shows the same mesh data as the previous two figures.

While both commercial meshing systems were able to create a complete mesh representation of surface 2, they were once again unable to generate a uniform mesh structure as was generated by the intersecting-sphere meshing technique. The histograms demonstrated in Figure 10-16 and Figure 10-17 also show a greater dispersion of nodal spacing values when compared to the histogram generated from results recorded from the intersecting-sphere meshing of the same geometry, demonstrated in Figure 10-17. It can be immediately observed that the intersecting-sphere generated mesh is an improvement over the commercially available systems when considering uniformity and the trend towards equidistant mesh structures.
Max nodal spacing = 5.9110
Min nodal spacing = 3.7417
Mean nodal spacing = 4.9300
Standard deviation = 0.4126

Max nodal spacing = 7.6063
Min nodal spacing = 3.9256
Mean nodal spacing = 4.9749
Standard deviation = 0.5715

Max nodal spacing = 5.0119
Min nodal spacing = 4.9644
Mean nodal spacing = 4.9981
Standard deviation = 0.0051

Figure 10-16: Gambit generated 5 unit mesh
Figure 10-17: Patran generated 5 unit mesh
Figure 10-18: Intersecting-sphere generated 5 unit mesh
10.7 Results: Surface 3

Figure 10-19: Mesh structure generated from 1 unit sphere

Figure 10-20: Histogram of mesh 6 surface 3

Target nodal spacing = 1
Max nodal spacing = 1.2968
Min nodal spacing = 0.91353
Mean nodal spacing = 0.99091
Standard deviation = 0.0249
Uniformity = True
10.7.1 Discussion of results for Surface 3

- Uniformity and mesh coverage
Figure 10-19 demonstrates the mesh structure generated from a sphere with a radius of one unit for surface 3, with the remainder of the structures demonstrated in Appendix B. For every specified sphere radius value utilised, the newly developed intersecting-sphere meshing technique generated a uniform mesh structure with a constant node to beam element ratio of 6:1 in the centre of the mesh and 2, 3 or 4:1 at the boundaries. These figures again demonstrate that mesh structure coverage of the target surface geometry was increased with a reduction in sphere radius.

- Equidistance
Figure 10-20 demonstrates an example histogram of the nodal spacing, the target nodal spacing and the maximum, minimum, mean and standard deviation of the values recorded. The values recorded over the range of decreasing radii (see Appendix B for graphical representations) initially showed a much greater dispersion in nodal spacing than those recorded for surface 2. The variation in nodal spacing can be attributed to the target geometry now exhibiting curvature in two axes when only curvature in one axis was present in surface 2. However, as the sphere radius was reduced, again the dispersion of the nodal spacing was also reduced in a similar trend to that previously seen. The dispersion, standard deviation and range of nodal spacing values once again showed a decreasing trend with the reduction in sphere radius until the generation of mesh 6. Mesh 6 was created with a 1 unit sphere radius and at this particular mesh resolution an increase in the range of nodal spacing is recorded. While the dispersion and standard deviation was reduced inline with the developing trend, a minority of larger nodal spacings are recorded and demonstrated in the histogram of Figure 10-2. After closer inspection of the mesh data, it was ascertained that these larger nodal spacings were found only at the boundary of the mesh structure. As the nodes at the boundary of the mesh are only required to be equidistant from between two and four other possible nodes, their locations could therefore easily be modified to produce a nodal spacing closer to the target value.
Table 10.3 documents the standard deviation recorded for each mesh structure generated using the intersecting-sphere meshing algorithm.

<table>
<thead>
<tr>
<th>Surface 3</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh 1: 10 unit radius sphere</td>
<td>0.1912</td>
</tr>
<tr>
<td>Mesh 2: 5 unit radius sphere</td>
<td>0.0872</td>
</tr>
<tr>
<td>Mesh 3: 4 unit radius sphere</td>
<td>0.0670</td>
</tr>
<tr>
<td>Mesh 4: 3 unit radius sphere</td>
<td>0.0553</td>
</tr>
<tr>
<td>Mesh 5: 2 unit radius sphere</td>
<td>0.0404</td>
</tr>
<tr>
<td>Mesh 6: 1 unit radius sphere</td>
<td>0.0249</td>
</tr>
</tbody>
</table>

Table 10.3: Standard deviation recorded from surface 3

Plotting the recorded standard deviation values from Table 10.3 against the relevant sphere radius produces Figure 10-21.

Figure 10-21: Standard deviation against sphere radius
The trend line generated from the plotted data again suggests that over the range of sphere radii used, a reduction of sphere radius produces a reduction in standard deviation, as was seen in surface 2, suggesting that mesh generation at decreasing radius values produces a trend towards equidistant mesh structures.

- **Comparison with commercial meshing solutions**
  
  As for previous surfaces, surface 3 was again meshed with a target nodal spacing of 5 units using both FEA meshing software solutions used formerly. The results of the meshing process using both systems are demonstrated individually in Figure 10-22 and Figure 10-23. For each mesh structure generated, the maximum, minimum, mean and standard deviation of the nodal spacing is again documented, in addition to a histogram demonstrating the dispersion of the nodal spacing. Figure 10-24 demonstrates the capability of the intersecting-sphere technique for meshing the target geometry at the same target nodal spacing and again shows the same mesh data as the previous two figures.

  While both commercial meshing systems were able to create a complete mesh representation of surface 3, both systems were again unable to generate a uniform mesh structure as was generated by the intersecting-sphere meshing technique. The histograms demonstrated in Figure 10-22 and Figure 10-23 also show a greater dispersion of nodal spacing values when compared to the histogram generated from results recorded from the intersecting-sphere meshing of the same geometry, demonstrated in Figure 10-24. Once again it can be easily seen that the intersecting-sphere generated mesh is an improvement over the commercially available systems when considering uniformity and the trend towards equidistant mesh structures.
Max nodal spacing = 6.2922
Min nodal spacing = 3.5861
Mean nodal spacing = 4.8643
Standard deviation = 0.4582
10.8 Results: Surface 4

Figure 10-25: Mesh structure generated from 1 unit sphere

Figure 10-26: Histogram of mesh 6 surface 4

Target nodal spacing = 1
Max nodal spacing = 1.1941
Min nodal spacing = 0.9308
Mean nodal spacing = 1.0103
Standard deviation = 0.0252
Uniformity = True
10.8.1 Discussion of results for Surface 4

- Uniformity and mesh coverage

Figure 10-25 demonstrate the mesh structure generated for a one unit sphere radius for surface 4, with the structures produced at the other sphere radii shown in Appendix B. For every specified sphere radius value utilised, the newly developed intersecting-sphere meshing technique generated a uniform mesh structure with a constant node to beam element ratio of 6:1 in the centre of the mesh and 2, 3 or 4:1 at the boundaries. These figures also demonstrated that mesh structure coverage of the target surface geometry was increased with a reduction in sphere radius.

- Equidistance

Figure 10-26 demonstrates a histogram of the nodal spacing, the target nodal spacing and the maximum, minimum, mean and standard deviation of the values recorded for the one unit sphere; the histograms produced for the other sphere radii can be found in Appendix B. Once again the dispersion, standard deviation and range of nodal spacing values recorded for all the specified sphere radii showed a decrease with reduced sphere radius. However, the two unit sphere radius for this surface produced a mesh structure almost equally comprised of two nodal spacing values. Throughout all the mesh structures generated until this example, the histograms have shown the most frequent nodal spacing value as the target value. While the reason for this particular dispersion of nodal spacing seen only in this case remains unclear, the recorded nodal spacing values still adhered to the developing trend of decreased standard deviation with decreased sphere radius.

Table 10.4 documents the standard deviation recorded for each mesh structure generated using the intersecting-sphere meshing algorithm.
Table 10.4: Standard deviation recorded from surface 5

Plotting the recorded standard deviation values from Table 10.4 against the relevant sphere radius produces Figure 10-27.

Figure 10-27: Standard deviation against sphere radius
The trend line generated from the plotted data once again suggests that over the range of sphere radii used, a reduction of sphere radius produces a reduction in standard deviation, again suggesting that mesh generation at decreasing radius values produces a trend towards equidistant mesh structures.

- **Comparison with commercial meshing solutions**

As completed for previous surfaces, surface 4 was again meshed with a target nodal spacing of 5 units using both FEA meshing software solutions used previously. The results of the meshing process using both systems are demonstrated individually in Figure 10-28 and Figure 10-29. For each mesh structure generated, the maximum, minimum, mean and standard deviation of the nodal spacing is again documented, in addition to a histogram demonstrating the dispersion of the nodal spacing. Figure 10-30 demonstrates the capability of the intersecting-sphere technique when meshing the same target geometry using the same target nodal spacing and again shows the same mesh data as the previous two figures.

While both commercial meshing systems were able to create a complete mesh representation of surface 4, both systems were again unable to generate a uniform mesh structure as was generated by the intersecting-sphere meshing technique. The histograms demonstrated in Figure 10-28 and Figure 10-29 also show a greater dispersion of nodal spacing values when compared to the histogram generated from results recorded from the intersecting-sphere meshing of the same geometry, as demonstrated in Figure 10-30. As all three histograms are plotted to the same scale, it can be easily seen that the intersecting-sphere generated mesh is an improvement over the commercially available systems when considering uniformity and the trend towards equidistant mesh structures.
Max nodal spacing = 6.3672
Min nodal spacing = 3.1762
Mean nodal spacing = 4.7859
Standard deviation = 0.5477

Max nodal spacing = 6.5057
Min nodal spacing = 3.5720
Mean nodal spacing = 4.8558
Standard deviation = 0.5410

Max nodal spacing = 5.3245
Min nodal spacing = 4.7953
Mean nodal spacing = 5.0296
Standard deviation = 0.0853
10.9 Results: Surface 5

Figure 10-31: Mesh structure generated from 1 unit sphere

10.9.1 Discussion of results for Surface 5

- Uniformity and mesh coverage
Figure 10-31 demonstrates the mesh structure generated for a one unit sphere radius for surface 5, which initially appears to show a uniform mesh. However, on closer examination it was found that there were several areas which were non-uniform at the boundaries of the surface.

The mesh structure generated, without the underlying surface, is demonstrated again in Figure 10-32, which also highlights a magnified view of one area where non-uniformity was observed in the mesh structure (1.). Figure 10-32 also demonstrates that non-uniformity was observed in a total of 8 areas, highlighted as (1 – 8), found at the boundary of the surface geometry. It can be seen from Figure 10-32 that, where failure has occurred, the triangulation process has created additional beam elements between
mesh nodes. If the triangulation process had not created the additional beam elements, quadrilateral elements would have been observed in the mesh structure.

As non-uniformity was recorded in the mesh structure, this immediately deemed the mesh unsuitable as a mapping mesh; however, it is still relevant to discuss the level of equidistance between the nodes.

- **Equidistance**
  
  Figure 10-33 demonstrates a histogram of the target and actual nodal spacings and the maximum, minimum, mean and standard deviation of the values recorded for the one unit sphere.
The histogram shows that, while a non-uniform mesh structure was generated, the majority of recorded elements were at the target value of 1 unit. The range and standard deviation of the spacings were greater than those seen for the previous surfaces, although based on previous results these values would be expected to show a substantial improvement over the meshes which could be produced by commercial systems.

Surface 5 has therefore highlighted the limitation of the newly developed intersecting-sphere meshing technique and shows that for very complex curved surfaces a uniform mesh structure cannot be achieved at the sphere radius values utilised, although the level of equidistance between nodes is still superior to that which would be expected from a commercial system. Further investigation at reduced sphere radii may enable the generation of a uniform mesh structure and has been addressed in the recommendations for further work regarding the intersecting-sphere meshing technique.
10.10 Discussion of results

The experimental testing to ascertain the capabilities of the newly developed intersecting-sphere meshing technique has generated very positive results. For every sample surface examined (with the exception of surface 5), the intersecting sphere technique was able to generate a uniform mesh structure. This is a particularly significant result, as uniformity within the mesh structure immediately renders the mesh potentially suitable as a mapping mesh for the creation of conformal RM textiles.

The meshing of a planar surface (surface 1) has shown that the new technique is capable of creating a fully uniform and equidistant mesh structure, potentially for any desired nodal spacing and dimension of planar surface. The meshing of geometry with curvature only apparent in one axis (surface 2), has shown that with a reduction in sphere radius or desired nodal spacing, the new technique is capable of creating a uniform mesh structure that is equidistant for all practical purposes. This is due to the deviation in nodal spacing observed being so insignificant at reduced sphere radius values, as to be negligible.

The meshing of curved surface geometries where curvature was apparent in two axes (surface 3, and 4), has echoed this developing trend towards equidistance with a reduction in sphere radius. While equidistance was not fully achieved for these surfaces, high quality uniform mesh structures with only very small standard deviation and range of nodal spacing values were generated at reduced sphere radius values. Once again it was felt that these minor deviations rendered the mesh, practically speaking, equidistant.

There was an observed trend towards equidistance with decreasing sphere radius in all cases. This can be explained as follows. The results of the experimental meshing have shown that where no curvature is present in the target geometry, equidistance can be achieved for all the resolutions of sphere radii utilised. When curvature is apparent in two axes of the target geometry, meshing at larger sphere radii produced greater nodal spacing deviation than that of smaller resolution of sphere radii.
If two spheres, one substantially larger than the other, are constrained by their centre point centrally to two identical curve surfaces their intersection splines will be as demonstrated in Figure 10-34.

![Figure 10-34: Example spheres constraint to identical surfaces](image)

The region of the curved surface that each sphere encapsulates can be examined and demonstrated in Figure 10-35.

![Figure 10-35: Region of surface encapsulated by both spheres](image)

If a closer examination of the two regions is undertaken using a frontal view as indicated in Figure 10-35, the two created regions at a manipulated scale can be made to appear identical in size. However, an intersection curve generated parallel to the viewing plane through the centre of the surface regions will show the apparent surface curvature experienced across the diameter of the respective spheres, as demonstrated in Figure 10-36.
Figure 10-36: Apparent surface curvature experienced by both spheres.

Figure 10-36 demonstrates the surface curvature experienced across the diameter of both spheres and clearly indicates a more pronounced level of curvature for the larger sphere than for the smaller one. In fact, as the radius of the sphere is reduced further, the encapsulated region of the target surface eventually approximates a planar surface. When this curvature ratio is achieved between a particular surface and a radius of sphere, the meshing process is approximately meshing a planar surface and therefore has the potential of generating an equidistant mesh structure. This relationship is therefore the likely cause of the seen trend towards equidistant mesh structures for reduced sphere radii and helps to explain the results generated in the experimental testing.

The final topic of discussion relates to the comparison of the intersecting-sphere meshing technique with the two tested commercially available systems. The two system chosen are capable of generate surface mesh structures that are indicative of current FEA pre-processors commercially available. For every test surface meshed, both commercial systems were unable to generate a uniform mesh structure which therefore makes their application for the generation of mapping meshes for RM textiles redundant. In addition, the nodal spacing values generated by both commercial systems showed a
greater range and standard deviation for all tested surfaces compared with those generated by the intersecting-sphere method. While this is not an exhaustive comparison, the newly developed meshing technique represents a significant improvement over the commercial systems tested for the surface configurations utilised in the experimental process.

The two commercial systems did show improved results over the intersecting-sphere technique with the creation of a complete mesh representation of the target geometry. While the intersecting-sphere technique produced better mesh coverage of the target geometry at reduced sphere radii, a complete representation was never achieved. However, it was the removal of this boundary constraint that now enables uniform and mesh structures with a trend towards equidistant to be actually generated.

The results of the experimental testing of the newly developed intersecting-sphere meshing technique demonstrates, to the author's knowledge, the only meshing system capable of generating suitable, uniform and tolerance-based equidistant mesh structures of curved surfaces that can be successfully implemented for the generation of conformal RM textiles. Having investigated and developed the generation of a suitable mesh structure, the next topic for exploration is the capability of mapping complex geometric data to surface meshes.
Chapter 11: Investigation of a new complex geometry mapping tool

11.1 Introduction

The requirement of the mapping tool is the capability to map complex 3D geometries to a surface mesh for the creation of conformal RM textile structures. One methodology for achieving such a task is the system used by TexGen discussed in Chapter 5, where an RVE is established and mapped to the individual enclosed section of a quadrilateral surface mesh. However, the initial experimental work highlighted the limitations of the RVE and the lack of geometric complexity achievable in the TexGen modelling system.

In order to achieve higher levels of geometric complexity, a more appropriate solution would be the direct mapping of CAD data. Using this technique, all the geometric complexity offered by CAD could be utilised for RM textile design. However, the removal of the RVE means an alternative mapping technique must be established.

FE surface meshes include two entities in their construction; nodes and elements, as demonstrated in Figure 11-1.

![Figure 11-1: Structure of FE meshes](image)
Nodes, shown as red spheres, are described as the end or corner points of elements and are represented in the mesh data file as Cartesian coordinates in a global coordinate system. The elements of the mesh are the geometry connecting the nodes, shown as beams; these are represented by which nodes they are connected to in the mesh data.

The nodes and elements of the mesh therefore provide locations on the surface where RM textile link geometries could be positioned; this is demonstrated in Figure 11-1. By mapping one link geometry configuration to the nodes of the mesh and a second link geometry configuration to the midpoints of the elements, an RM textile structure can be readily created. If a more complex RM textile configuration was utilised that only required one link structure, then this could also be achieved by simply mapping to the node or element midpoint only, as demonstrated in Figure 11-3.

Figure 11-2: Link mapping to nodes (left) and nodes and element midpoints (right)

Figure 11-3: Complex link mapped to node only
11.2 Geometry mapping fundamentals

The aim of the following subsections (11.2.1 and 11.2.2) are to provide the reader with a fundamental understanding of the issues surrounding the mapping of geometry to curved structures, such as simple curves and more complex surface meshes, addressing the in-depth mathematical analysis in later sections.

When mapping geometry to any surface mesh or, in a simple 2D case, a curve, the requirement will always be to match accurately the orientation of the geometry to the normal of the curve or the surface normal of the surface mesh at a specified mapping location. Matching the orientation of the geometry to the normals therefore correctly orientates the geometry to the curve or surface mesh.

11.2.1 2D mapping to a curve

If a simple 2D example is considered, as detailed in Figure 11-4, where a simple geometry is to be mapped to a 2D curve, the following example highlights the importance of matching the orientations.

Figure 11-4 shows the intended mapping of an example geometry to nine locations along the curve while maintaining the original orientation of the mapping geometry. While this situation maps the geometry to the locations along the curve as demonstrated in Figure 11-5, the disparity between the out-facing orientation of the geometry and the
normal of the curve at the nine mapping location actually causes inaccurate tessellation of the geometry.

Accurate mapping of the geometry along the curve therefore requires the intended out-facing orientation of the geometry to match the normal of the curve at the nine mapping locations, as detailed in Figure 11-6.

Figure 11-6: Normals of curve at the nine mapping locations
Correctly matching the outfacing orientation of the mapping geometry to the curve normals at the mapping locations therefore accurately maps the geometry to the curve as demonstrated in Figure 11-7.

11.2.2 3D mapping to a surface mesh
Now consider the mapping curve highlighted in the previous example as a curve taken through a surface targeted for geometry mapping, as detailed in Figure 11-8, where a surface is used to approximate a surface mesh. It can be easily seen that while the mapped geometry conforms to the normals of the curve at the nine mapping locations, it does not conform to the surface normal at the same mapping locations.
The surface normal, or the normal of a surface at a specified location, can be described as a vector perpendicular to a newly created tangential plane constrained to the surface at this specified location. This can be more easily described by Figure 11-9.

Tangential plane (translucent blue) to surface centred at specific mapping location (black sphere)

The surface normal is a vector (red arrow) perpendicular to the tangential plane at the specific mapping location

Figure 11-9: Surface normal of mapping surface at a specified location

The surface normal at the mapping location therefore provides the correct orientation of the mapping geometry. However, while the orientation is determined by the surface normal, this does not provide any information regarding the rotation of the mapping geometry at this location. Therefore the geometry is unconstrained and able to rotate using the surface normal as an axis while still maintaining the correct surface normal orientation as demonstrated in Figure 11-10.
The capability of the mapping geometry to rotate about the surface normal can be eliminated by the generation of a local coordinate system at the mapping location. The local coordinate system consists of three vectors; the surface normal acts as the Z axis while two further orthogonal vectors are created representing the X and Y axes, completing the local system, as demonstrated by Figure 11-11.
The new local system at the mapping location therefore provides reference axes for the mapping geometry when orientating from the original global coordinate system and its initial orientation, to the surface normal orientation at the mapping location, fully constraining the orientation of the geometry into the correct position. If a local coordinate system is developed for every mapping location on a surface mesh, this therefore allows the accurate mapping of geometries that match the surface normals at these locations and guarantees the correct orientation of the geometries, as demonstrated by Figure 11-12.

![accurately mapped geometry using local coordinate systems at each mapping location](image)

Figure 11-12: Accurately mapped geometry using local coordinate systems at each mapping location

### 11.3 Mapping methodology

In order to achieve the complex geometry mapping capability discussed in section 11.1, the developed mapping tool must be able to perform several functions. These include the capability to read the data contained in the surface mesh file, calculate the surface normal and local coordinate system at all mesh node locations, import CAD geometry data files and finally the capability to copy and reposition the CAD geometry data at the nodes and element midpoints in an orientation that matches the surface normal at these locations. Once this has been achieved, the final requirement of the mapping tool would be the capability to export the complete RM textile structure as the STL data as required for manufacture by RM systems.

The sequence of events required for the successful mapping of a singular RM textile link geometry can be more easily described in a pictorial step by step progression as follows in Figure 11-13:
Stage one: Import CAD data and surface mesh data into a global coordinate system
Stage two: Calculate surface normal and local coordinate system at required node mapping location
Stage three: Orientate link geometry from global coordinate system to local coordinate system
Stage four: Translate orientated link to mapping location

The next sections in this chapter relate to the four stages of the mapping methodology, detailing and developing the mathematical principles and equations necessary for the successful mapping of one RM textile link structure. Creating a methodology for the successful mapping of one RM textile link structure to the node or element midpoint of a surface mesh can therefore be easily extended to produce a complete RM textile structure.

Where possible 2D and 3D examples have been utilised to aid the reader’s understanding of the development process required in the mapping methodology.
Stage one: Import CAD and surface mesh (mesh shown as surface)

Stage two: Determine local coordinate system at node site

Stage three: Orientate link geometry to local coordinate system while remaining at the origin

Stage four: Translate orientated geometry to mapping location

Figure 11-13: Mapping methodology
11.4 Stage one: Import CAD data and surface mesh data

The initial step of the mapping process requires the import of the RM textile link CAD data. As the ultimate aim of the mapping tool is the creation of a complete RM textile STL data file for manufacture, it seems appropriate to use the STL file format as the direct input. Using the STL format also removes any necessity for any geometric data format conversion later in the process.

The STL file format represents the geometry it is describing by a system of vertices and triangles. Each triangle is described individually by three vertices which are described as Cartesian coordinates. An extract of a single triangle from the ASCII STL format is demonstrated in Figure 11-14.

<table>
<thead>
<tr>
<th>Solid: Name</th>
<th>Facet normal</th>
<th>Outer loop</th>
<th>End loop</th>
<th>End facet</th>
<th>End solid</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>n(i) n(j) n(k)</td>
<td>Vertex 1(x) 1(y) 1(z)</td>
<td>Vertex 2(x) 2(y) 2(z)</td>
<td>Vertex 3(x) 3(y) 3(z)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>File name of geometry</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Start of triangle creation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Vertex 1 x,y and z coordinates</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Vertex 2 x,y and z coordinates</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Vertex 3 x,y and z coordinates</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>End triangle creation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Triangle complete</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Geometry complete</td>
</tr>
</tbody>
</table>

Figure 11-14: Example ASCII STL file extract

Using the STL file format for CAD geometry import means the only requirement of the mapping tool is the capability to read the coordinates of the vertices. Recreating each vertex sequentially, and therefore each triangle, will ultimately lead to the recreation of the entire RM textile STL link geometry. Also, when reorientation of the geometry occurs in the mapping process, each vertex of the triangle can be recalculated individually. Manipulating each vertex sequentially while retaining the vertex connectivity as detailed in Figure 11-14, therefore recreates the triangle in a modified orientation.

Another important consideration is the position of the geometry during creation and therefore its initial position and orientation when imported to the mapping tool. To
simplify the later process of orientating the geometry to match the local system at the mapping location, the imported geometry is required to be situated at the origin of the global coordinate system in which is was created. Once fully created in a CAD package and converted to STL format, the global position it occupies will also be recorded with the STL file.

The second requirement of the mapping tool is the capability to read the data contained within the mesh file so the surface normal and local coordinate system at every node or element midpoint location can be calculated. Generic FE mesh data files, in a similar manner to the STL file, consists of a node list with the corresponding Cartesian coordinates set to a global coordinate system, and a further list detailing which nodes are connected each other. The second list of node connectivity therefore describes the element geometries of the surface mesh. Again, similar to the STL file usage, the mapping tool requires only the capability to read the coordinates of the nodes and document which other nodes are connected to them within the surface mesh.

11.5 Stage two: Calculate surface normal and local coordinate system at mesh nodes and element midpoints

Calculating the surface normal and local coordinate system at the nodes and element midpoints requires the vectors describing the elements connected to a particular node to be known. While this data is not included in the mesh file it can be easily calculated through the node connectivity list and the application of simple vector analysis. Figure 11-15 shows an example surface with a small section of a corresponding example surface mesh. The example surface mesh consists of nodes (A-G) and elements (ab-ag). The coordinates for the nodes (A-G) would be supplied within the mesh data file with a further list detailing which nodes are connected to node (A).
The vectors of the elements (ab-ag) can be calculated using vector analysis and the node coordinates from the mesh file as follows:

\[
\begin{align*}
  ab &= B - A \\
  ac &= C - A \\
  ad &= D - A \\
  ae &= E - A \\
  af &= F - A \\
  ag &= G - A
\end{align*}
\]

Once the element vectors have been calculated, it then becomes possible to calculate the surface normal at node (A) by using a system of calculating the vector product of pairs of connected vector elements.

The vector product (sometimes described as the cross product) can be defined as a third vector perpendicular or normal to two input vectors. The notation for the cross product of two input vectors a and b for example, becomes \(|a \times b|\), as shown in Figure 11-16.
Figure 11-16: Vector cross product

The vector cross product \( \mathbf{a} \times \mathbf{b} \) can be calculated using the determinant of the following matrix:

\[
\begin{vmatrix}
  i & j & k \\
  x_1 & y_1 & z_1 \\
  x_2 & y_2 & z_2 \\
\end{vmatrix}
\]

Equation 11.1

Where the input vectors \( \mathbf{a} \) and \( \mathbf{b} \) are described as:

\[
\mathbf{a} = (x_1, y_1, z_1) \\
\mathbf{b} = (x_2, y_2, z_2)
\]

Therefore:

\[
\begin{align*}
\mathbf{a} \times \mathbf{b} &= i(y_1 z_2 - y_2 z_1) - j(x_1 z_2 - x_2 z_1) + k(x_1 y_2 - x_2 y_1) \\
\end{align*}
\]

Equation 9.1 must therefore be executed for every adjacent pair of non-parallel elements connected to node (A) in the example surface mesh to produce a third vector perpendicular to both input vectors. The following figures show the creation of the cross product vectors for every adjacent pair of non-parallel elements connected to node (A) in the example surface mesh.
Figure 11-17: Vector cross product of ab and ac (left), ac and ad (right)

Figure 11-18: Vector cross product of ad and ae (left), ae and af (right)

Figure 11-19: Vector cross product of af and ag (left), ag and ab (right)
Once completed, a series of six vectors are created from which an average can be taken that corresponds to the surface normal at the example node location, as demonstrated in Figure 11-20.

![Surface normal creation at node location](image)

The surface normal created at the example node location is therefore the Z component of the new local coordinate system at node (A) required to successfully orientate the link geometry from its original orientation. The Y component of the local coordinate system can be calculated using the cross product of the calculated surface normal and an existing mesh element vector that approximates to the global X axis, temporarily representing the X component of the local system as demonstrated in Figure 11-21.

![Y component of local system calculation using mesh element](image)
This process can then be repeated to determine an accurate X component of the local system by calculating the cross product of the Z component (surface normal) and the newly calculated Y component. Figure 11-22 shows the final component and the completion of the local coordinate system at the node location.

Using the equations and methodology developed in this section it becomes possible to calculate a local coordinate system at every surface mesh node from generic mesh data for the purpose of mapping complex geometry. While not included within this example, the development of a local coordinate system for the midpoints of mesh elements, for example (ab) as demonstrated in Figure 11-22, can be achieved by first calculating the midpoint of the element using vector analysis and the local coordinate systems at both nodes (A) and (B), or any connected nodes that describe the element. Once achieved, the local coordinate system at the midpoint of the mesh elements can be determined by averaging the appropriate vectors from both the calculated local coordinate systems at both nodes which define the element.

Having the capability to calculate the required local coordinate system for accurate mapping of complex geometry as required by the mapping tool, the next requirement is
the capability to calculate the orientation of the geometry to match the calculated local systems.

11.6 Stage three: Orientate link geometry from global coordinate system to local coordinate system

The orientation of the link geometry to match the newly calculated local coordinate system can be achieved by transforming the individual coordinates of the vertices that form the triangles of the STL file from their initial global coordinate system to the new local system calculated at the node location. Manipulation of the vertex coordinates, while retaining their connectivity information from the STL data, as shown in Figure 11-14 where each triangle is defined by its own section of code, therefore enables the recreation of the individual triangles of the geometry in a new orientation.

In order to achieve the 3D transformation of the triangle vertices from the global coordinate system to the local system calculated at the node location, an equation must be developed that calculates the new coordinates of the vertices when orientated in the new local system. An equation for achieving this can be initially derived from a simple 2D transformation, as shown in the following example.

11.6.1 2D transformation

Consider a 2D transformation where \( P \) is a point \((x, y)\) in the \( XY \) global coordinate system and also a point \((x', y')\) in the \( X'Y' \) local coordinate system, where \( \theta \) is the rotation angle between the two coordinate systems \((XY)\) and \((X'Y')\).
From Figure 11-23 it can be seen that:

Shown as red arrows
\[ x = x' \cos \theta - y' \sin \theta \]  
Equation 11.2

And

Shown as blue arrows
\[ y = x' \sin \theta + y' \cos \theta \]  
Equation 11.3

Equations 11.2 and 11.3 calculate the coordinates of \((x, y)\) with the angle of rotation and the coordinates \((x', y')\) in the local system as inputs. However, for the orientation of the geometry in the mapping process for the creation of RM textile structures, the requirement is the calculation of the coordinates in the newly calculated local coordinate systems at the mesh nodes or element midpoints. Therefore, each equation must be solved for the new local axes \((x', y')\).

Solving equations 11.2 and 11.3 for \(x'\) and \(y'\), first solving for \(y'\):

Multiply equation 11.2 by \(\sin \theta\)
\[ x \sin \theta = x' \sin \theta \cos \theta - y' \sin^2 \theta \]  
\text{Equation 11.4}

Multiply equation 11.3 by \( \cos \theta \):
\[ y \cos \theta = x' \sin \theta \cos \theta + y' \cos^2 \theta \]  
\text{Equation 11.5}

Subtracting equation 11.4 from equation 11.5:
\[ y \cos \theta - x \sin \theta = y' (\cos^2 \theta + \sin^2 \theta) \]
Where: \( \cos^2 \theta + \sin^2 \theta = 1 \)
Therefore:
\[ y' = -x \sin \theta + y \cos \theta \]  
\text{Equation 11.6}

Solving for \( x' \), multiply equation 11.2 by \( \cos \theta \):
\[ x \cos \theta = x' \cos^2 \theta - y' \sin \theta \cos \theta \]  
\text{Equation 11.7}

Multiply equation 11.3 by \( \sin \theta \):
\[ y \sin \theta = x' \sin^2 \theta + y' \sin \theta \cos \theta \]  
\text{Equation 11.8}

Adding equation 11.7 to equation 11.8:
\[ x \cos \theta + y \sin \theta = x' (\cos^2 \theta + \sin^2 \theta) \]
Where: \( \cos^2 \theta + \sin^2 \theta = 1 \)
Therefore:
\[ x' = x \cos \theta + y \sin \theta \]  
\text{Equation 11.9}

Using equations 11.9 and 11.6, any specified point in a 2D system can now be transformed to its corresponding point in a new 2D local coordinate system assuming the angle of rotation between the two systems, \( \theta \), is known.
\[ x' = x \cos \theta + y \sin \theta \]  
\text{Equation 11.9}
\[ y' = -x \sin \theta + y \cos \theta \]  
\text{Equation 11.6}

Equations 11.9 and 11.6 can be further simplified and combined into matrix notation as:
\[
\begin{pmatrix}
  x' \\
  y'
\end{pmatrix} =
\begin{pmatrix}
  \cos \theta & \sin \theta \\
  -\sin \theta & \cos \theta
\end{pmatrix}
\times
\begin{pmatrix}
  x \\
  y
\end{pmatrix}
\]  

Equation 11.10

The individual matrices of equation 11.10 can be further simplified and represented as:

Local coordinate axes matrix (L) = \[
\begin{pmatrix}
  x' \\
  y'
\end{pmatrix}
\]

Rotation matrix (R) = \[
\begin{pmatrix}
  \cos \theta & \sin \theta \\
  -\sin \theta & \cos \theta
\end{pmatrix}
\]

Global coordinate axes matrix (G) = \[
\begin{pmatrix}
  x \\
  y
\end{pmatrix}
\]

Therefore \[ L = R \times G \]  

Equations 11.11

Equation 11.10 and 11.11 can be further developed to utilise the unit vectors of the axes of the local coordinate system as an alternative to the angle of rotation. However, first it is appropriate to define unit vectors and their equations for clarity.

Considering a point (A) in 2D space with the coordinates (x,y), the vector position of that point can be written as vector a.

Where: \[ a = xi + yj \]  

Equation 11.12

The length of the vector becomes the distance from the global origin to point (A), which can be calculated using Pythagoras's theorem. The notation for the length of the vector a becomes |a|.

Where: \[ |a| = \sqrt{i^2 + j^2} \]  

Equation 11.13

The unit vector is described as a vector with the same direction as the input vector but with a magnitude or length of one unit. The notation for the unit vector of vector a becomes â and can be calculated by multiplying the known vector a by the reciprocal of its length.
Therefore, \( \hat{a} = |a|^{-1} \times a \)  

Equation 11.14

Returning to the development of a 2D transformation using unit vector components, Figure 11-24 shows the unit vectors (shown in green) of the coordinate system \((X',Y')\) detailed in Figure 11-23.

![Figure 11-24: Unit vectors of local system axes](image)

The new local coordinate system \((X'Y')\) will have a new unit vector shown in green with two components \((i, j)\) shown as red and blue arrows respectively for each of the new local system axes \(X'\) and \(Y'\):

Where:

\[
\hat{X}' = (i + j) \\
\hat{Y}' = (i + j)
\]

As the unit vector acts in the same direction as the system axis it is representing but with a vector length of one unit, the components \((i, j)\) of the unit vectors therefore become:

\[
\hat{X}' = (i + j) = (\cos \theta + \sin \theta) \\
\hat{Y}' = (i + j) = (\sin \theta + \cos \theta)
\]

Equation 11.15  
Equation 11.16

As demonstrated in Figure 11-25, where \(1 \times \sin \theta = \sin \theta\)
Assigning a value to the angle of rotation (θ) between the global system and the local system of the example detailed in Figure 11-23, Figure 11-24 and Figure 11-25 to (-60) degrees, the rotation matrix (R) of equation 12.11 becomes:

\[
\text{Rotation matrix (R)} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{bmatrix}
\]  

Equation 11.17

Also, the unit normal of the local coordinate system \( \hat{X}' \) and \( \hat{Y}' \) as demonstrated in Figure 11-26, become:

\[
\hat{X}'(i) = 1/2 \\
\hat{X}'(j) = \sqrt{3}/2 \\
\hat{Y}'(i) = -\sqrt{3}/2 \\
\hat{Y}'(j) = 1/2
\]

Therefore, substituting the unit vector components into \( (R) \) from equation 11.17
Rotation matrix (R) =
\[
\begin{bmatrix}
\cos \theta & \sin \theta \\
-\sin \theta & \cos \theta 
\end{bmatrix}
\begin{bmatrix}
1/2 \\
-\sqrt{3}/2 
\end{bmatrix}
\]
Equation 11.17

Substituting components of unit normal:

New rotation matrix (R) =
\[
\begin{bmatrix}
\hat{X}'(i) & \hat{Y}'(i) \\
\hat{X}'(j) & \hat{Y}'(j) 
\end{bmatrix}
\]

Substituting (R) back into equation 12.10 produces:

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} =
\begin{bmatrix}
\hat{X}'(i) & \hat{Y}'(i) \\
\hat{X}'(j) & \hat{Y}'(j) 
\end{bmatrix}
\begin{bmatrix}
x \\
y
\end{bmatrix}
\]
Equation 11.18

Also, reverting back to algebraic notation generates two new equations, one for each of the local coordinate system axes as follows:

\[
x' = x \times \hat{X}'(i) + y \times \hat{Y}'(i)
\]
Equation 11.19

\[
y' = x \times \hat{X}'(j) + y \times \hat{Y}'(j)
\]
Equation 11.20

This substitution can be achieved because the unit vector components (i,j) of the local coordinate system always conform to the correct sign convention negating the requirement of a $-\sin \theta$ value in equation 11.6.

Having established an equation for the 2D transformation of a point, this equation can now be expanded further to calculate a 3D transformation as follows.

11.6.2 3D Transformation

Consider the transformation of a single vertex of an STL file from its initial global coordinate system (XYZ) to a calculated local system (X'Y'Z'), as demonstrated in Figure 11-27.
Any vertex of the STL file in will be described as a global coordinate \((x, y, z)\) and in the local system as \((x', y', z')\). Also, as a 3D system is now being utilised, each unit vector of a particular local axis will now have three components \((i, j, k)\), corresponding to their Cartesian coordinates.

If equation 11.18 is reconsidered:

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} =
\begin{bmatrix}
  \hat{X}'(i) & \hat{Y}'(i) \\
  \hat{X}'(j) & \hat{Y}'(j)
\end{bmatrix}
\times
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
\]

Equation 11.18

Modifying equation 11.18 to include the new third axis of the global and local systems, \((Z)\) and \((Z')\) respectively will also require the addition of five more variables, shown as \(\text{?}\); these are required in order to validate the matrix equation.

\[
\begin{bmatrix}
  x' \\
  y' \\
  z'
\end{bmatrix} =
\begin{bmatrix}
  \hat{X}'(i) & \hat{Y}'(i) & \text{?} \\
  \hat{X}'(j) & \hat{Y}'(j) & \text{?}
\end{bmatrix}
\times
\begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix}
\]

Equation 11.21

As each axis of the 3D local coordinate system will have a unit vector consisting of three components \((i,j,k)\), five new variables are created as follows:
k or z component of $X'$ axis unit normal $\hat{X}'(k)$

k or z component of $Y'$ axis unit normal $\hat{Y}'(k)$

i or x component of $Z'$ axis unit normal $\hat{Z}'(i)$

j or y component of $Z'$ axis unit normal $\hat{Z}'(j)$

k or z component of $Z'$ axis unit normal $\hat{Z}'(k)$

Therefore, including the new unit vector components in equation 11.19 generates equation 11.20:

$$
\begin{bmatrix}
  x' \\
  y' \\
  z'
\end{bmatrix} =
\begin{bmatrix}
  \hat{X}'(i) & \hat{Y}'(i) & \hat{Z}'(i) \\
  \hat{X}'(j) & \hat{Y}'(j) & \hat{Z}'(j) \\
  \hat{X}'(k) & \hat{Y}'(k) & \hat{Z}'(k)
\end{bmatrix} \times
\begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix}
$$

Equation 11.22

Also, reverting back to algebraic notation generates three new equations, one for each of the local coordinate system axes as follows:

$$
x' = x \times \hat{X}'(i) + y \times \hat{Y}'(i) + z \times \hat{Z}'(i)
$$

Equation 11.23

$$
y' = x \times \hat{X}'(j) + y \times \hat{Y}'(j) + z \times \hat{Z}'(j)
$$

Equation 11.24

$$
z' = x \times \hat{X}'(k) + y \times \hat{Y}'(k) + z \times \hat{Z}'(k)
$$

Equation 11.25

Equations 11.22 or 11.23-11.25 therefore provide a method for calculating the transformation of a single vertex in an STL file from its initial global coordinate system to any potential local coordinate system calculated from the surface mesh as detailed in the previous section. Repeating this process for every vertex within an STL file while retaining its triangle connectivity information ultimately leads to the recreation of the original STL geometry in a new orientation, while still remaining at the global origin.
Having the capability to calculate the required local coordinate system at the surface mesh nodes and element midpoints, coupled with the equations to orientate the geometry to match such local systems, the next requirement of the mapping tool is the translation of the orientated geometry to the surface mesh node or element midpoints.

11.7 Stage four: Translation of orientated link geometry to mapping location

The final stage of the methodology requires the translation of the newly orientated link geometry to the actual mapping location as demonstrated in Figure 11-28. The translation of the mapping geometry is achieved by translating the individual vertices of the STL file from its new orientation position at the global origin to the mapping location. Once again this is achieved by translating the individual vertices of the STL while retaining the individual triangle connectivity. Repeating this process for every vertex within the STL file will ultimately lead to the recreation of the entire geometry at the mapping location.

Figure 11-28: The mapping process up to translation of geometry to mapping location

The translation of the geometry can be achieved by using the vector from the global origin to the mapping location, as demonstrated in Figure 11-29.
As a 3D coordinate system is being utilised, the vector describing the mapping location will obviously include three components (xyz) as demonstrate in Figure 11-30.

Therefore, by adding the relevant component of the vector to the relevant coordinate of the orientated vertex, the vertex will be translated to the correct position at the mapping location. This is demonstrated in the following simple example:

Mapping location vector = \( (5+5+5) = (x,y,z \text{ components}) \)

Vertex coordinate of STL file = \( (1,1,0) = (x,y,z \text{ coordinates}) \)

New x coordinate of vertex = original (x) coordinate + (x) component of vector

\[ = (1+5) = 6 \]

New y coordinate of vertex = original (y) coordinate + (y) component of vector

\[ = (1+5) = 6 \]

New z coordinate of vertex = original (z) coordinate + (z) component of vector

\[ = (1+5) = 6 \]
\[ = (0+5) = 5 \]

Therefore the new coordinate of the specific vertex becomes (6,6,5).

This operation therefore concludes the process required by the mapping tool for the successful mapping of a single RM textile structures to a surface mesh. Repeating the methodology of at every mesh node and element midpoint contained within a targeted surface mesh therefore enables the creation of a conformal RM textile structure.

11.8 Final STL creation

The final requirement of the mapping tool is the creation of the STL file required for manufacture of the complete RM textile structure. As the connectivity of the original STL file describing a single RM textile link structure has been maintained throughout the developed mapping process, the mapping tool need only document the individually mapped copies as shell entities within a new STL file assembly while omitting the original structure. A shell entity of an STL file can be defined as a non connected body. Therefore, each structure becomes a separate entity which is fully described within the final STL file that can then be used for manufacture.

11.9 Coding the developed mapping methodology

To fully validate the capabilities of the methodology when mapping geometric STL data to surface meshes for the creation of RM textile structures, the methodology was coded into a standalone software mapping tool known as ChainLink. To expedite the coding process, the methodology developed was coded in collaboration with researchers at Nottingham University who had the expertise and experience necessary to create a standalone software tool.

The next chapter in this thesis is therefore the experimental testing of the developed methodology by using the ChainLink software to map complex geometry to surface meshes.
Chapter 12: Validation of mapping of complex geometries to surface meshes

12.1 Introduction

The aim of this chapter is the validation of the mapping methodology explained in the previous chapter. The validation of the mapping methodology has been executed using the C++ coded version of the mapping methodology, ChainLink.

ChainLink utilises the exact methodology investigated in previous chapter to map geometric STL data to the nodes and element midpoints of specified surface mesh structures. At the time of submission of this doctoral thesis an appropriate interface between the newly developed intersecting-sphere meshing system and ChainLink had not been fully established, negating the mapping of geometric STL data to the uniform mesh structures that can now be generated. However, for development and testing purposes, ChainLink was initially coded to import Gambit-generated surface mesh files. As demonstrated in chapter 11, the Gambit-created surface meshes do not have a uniform structure or exhibit the same high quality as surface meshes created from the intersecting-sphere system when considering a trend towards equidistance. However, the process of mapping geometric STL data to nodes and element midpoints is an identical process no matter what perceived level of quality the surface mesh may exhibit. Therefore, mapping geometric STL data accurately to any quality of surface mesh will ultimately validate the capabilities of the mapping methodology.

An interface between the intersecting-sphere meshing algorithm investigated in chapter 10 and the ChainLink software is part of an ongoing IMCRC/EPSRC project, and forms one of the recommendations for further work in Chapter 15.
12.2 Geometry Orientation

As discussed in the previous chapter, the accurate mapping of geometric STL data to a surface mesh requires the location in Cartesian space and the surface normal to be calculated. However, the process of determining the orientation or surface normal at a mapping location from a mesh data only, as investigated in Chapter 11, is the only mathematical technique that can be successfully implemented. As no other system can be utilised without actual target surface data, it was considered that a visual inspection of the geometry mapped by ChainLink would therefore be a suitable evaluation technique.

To evaluate the orientation calculated by ChainLink when mapping geometric STL data, a visual inspection of the final STL, using a suitable CAD system, was utilised. To aid the visual inspection process, the test geometry for this experiment, test geometry 1, is demonstrated in Figure 12-1.

![Intended orientation and Global origin](image)

Figure 12-1: Test geometry 1 showing intended orientation

Figure 12-1 demonstrates the test geometry and shows its initial orientation. When correctly mapped to a mesh structure, the cylindrical arrow will be presented in the same orientation as the surface normal at the specified mapping location. A visual inspection of the generated structure therefore provides a good indication of the accuracy of the process. Testing of the orientation produced by ChainLink is presented here.
In order to validate the orientation capability of ChainLink, test geometry 1 was mapped to the surface mesh structures demonstrated in Figure 12-2. Once mapping was complete, the STL file created by ChainLink was then imported into Studio Max for a visual inspection of the created geometry.

- Surface mesh 1: triangular planar surface mesh with a target 1 unit nodal spacing
- Surface mesh 2: triangular hemispherical surface mesh with a target 1 unit nodal spacing

Figure 12-2: Gambit-created surface mesh structures, (left) surface mesh 1, (right) surface mesh 2

12.2.1 Surface mesh 1
Figure 12-3 shows a right sided rendered view of the STL data created by ChainLink, by mapping test geometry 2 to surface mesh 1.

Figure 12-3: Side view taken from studio max of ChainLink created planar structure
A perspective representation is also shown in Figure 12-4.
As surface mesh 1 is a planar mesh, the surface normal at each point is perpendicular to the surface at any given point, and therefore has the same orientation as the z axis. Figures 18 and 19 clearly demonstrates that all of the mapped geometries are orientated in-line with the z axis, meaning that all geometries are perpendicular to the planar mesh and are therefore correctly orientated.

### 12.2.2 Surface mesh 2

Figures 19 and 20 demonstrates a right sided and a perspective view respectively, of the structure created by ChainLink from the mapping of test geometry 2 to surface mesh 2.
Once again it can be seen that the geometry appears to be orientated perpendicular to the surface at each mapping location.

The results of this set of experiments indicate that the surface normal calculation method utilised by ChainLink to accurately orientate the mapping geometry at node and element midpoint locations is indeed adequate. The orientation of the mapped geometry at each location can be seen to correlate to what would be expected when mapping the geometry to locations on the surface of a hemisphere whilst matching the surface normal at such locations.

As the system is dependent solely on the mesh structure to calculate the surface normal, any increase in mesh quality by uniformity or equidistance will generate a more precise surface normal calculation, which would be expected to further improve the final model produced.

12.3 Geometric complexity

ChainLink has been developed to allow any level of geometric complexity that can be described as STL data to be mapped to a surface mesh. This capability therefore
reflects the level of complexity achievable by additive manufacturing and provides the possibility for RM textiles of high geometric complexity to be readily created. To evaluate this capability, a selection of three complex test geometries have been mapped to a the hemispherical surface shown previously, surface mesh 2, and the final STL data file visually inspected using a suitable CAD system. This visual inspection was utilised to ascertain whether any distortion was apparent in the complex geometry as it was mapped over the curved surface mesh structure. The structures used for this validation experiment are not examples of structures that can be utilised for the creation of RM textiles as they will not tessellate in the required manner. However, the complexity of these structures provides a good indication of the level of complexity that can be successfully mapped. Figure 12-7 and Figure 12-8 represent the three selected test geometries (2, 3 and 4) respectively.

Sections 12.3.1, 12.3.2 and 12.3.3 show the results of mapping geometries 2, 3 and 4 to surface mesh 2 using ChainLink.

Figure 12-7: (left) Test geometry 2, Teapot, (right) Test geometry 3, Mobius knot
12.3.1 Test geometry 2: Teapot

Figure 12-9 demonstrates test geometry 2, mapped to surface mesh 2 and a magnified view of several of the mapped geometries.
12.3.2 Test geometry 3: Mobius knot

Figure 12-10 demonstrates test geometry 3, a Mobius knot, mapped to surface mesh 2 and a magnified view of several mapped geometries.

Figure 12-10: Test geometry 3 mapped to surface mesh 2

Again, the STL data created by ChainLink and demonstrated in Figure 12-10 was visually inspected using Studio Max, and this inspection of the individual mapped geometries revealed absolutely no deformation of the mapped geometries when compared to the original configuration. Again, the same seemingly random rotation of the mapped geometries was observed about the surface normal at the mapping location.
The visual inspection of the STL data demonstrated in Figure 12-9 using Studio max indicated that no deformation was present in any of the individual mapped geometries. Careful inspection of each mapped geometry revealed that the original configuration of the mapping geometry was consistently maintained throughout the entire mapping process. An important observation from Figure 12-9 is the seemingly random rotation of the mapped geometry. The cause of this inconsistent rotation is a product of the local axis calculation at the mapping location and the apparent quality of the mesh. When comparing the STL data demonstrated in Figure 12-9, the spout and handle of the teapot mapping geometry are actually perfectly aligned to a beam element of the mesh. Section 11.5 from the previous chapter explained the methodology of this system, where an appropriate mesh element is temporarily selected to approximate the global x axis for the calculation of the local coordinate system. Using this system in combination with a geometry that is not intended to tessellate in a triangular, hexagonal pattern, ultimately leads to the rotation as demonstrated in Figure 12-9. However, when considering RM textile structures that are required to tessellate in such a controlled manner and mapped to a uniform mesh structure, the observed rotation of mapped geometries no longer becomes an issue.
12.3.3 Test geometry 4: Complex lattice structure

Figure 12-11 demonstrates test geometry 4, a complex lattice structure, mapped to surface mesh 2 and a magnified view of several mapped geometries.

![Figure 12-11: Test geometry 4 mapped to surface mesh 2](image)

Test geometry 4 is the most complex mapping structure yet attempted. Again, the data demonstrated in Figure 12-11 was visually inspected through Studio Max and again revealed absolutely no deformation of the individual mapped structures.

The results of the work presented in this section indicate that ChainLink has the capability to map the example complexity of geometries described in STL format with absolutely no deformation of the structure. As the methodology employed by ChainLink is identical for all complexities of STL geometry, the ability to map these particularly complex geometries provides a strong indication that no matter what complexity the selected geometry may possess, ChainLink will be capable of successfully mapping the STL data to any surface mesh.
12.4 Discussion of results

The two validation experiments undertaken in this experimental chapter were designed to test the main capabilities of the geometric mapping methodology investigated by the author and realised through the coding of ChainLink.

The first experiments were designed to validate the calculation of the surface normal at mapping locations from the mesh data only. This experiment was undertaken with a lower quality of mesh than can now be achieved. However, the visual inspection of both the planar and hemispherical generated structures indicates that the process developed in the mapping methodology is entirely adequate and capable of correctly orientating the mapped geometry.

Finally, experiments were carried out to evaluate the level of geometric complexity that can be successfully mapped with no deformation. The system developed for achieving this capability recalculates the position of each individual vertex of a triangle within the original STL file using a single equation while maintaining the triangle connectivity. Therefore, having the capability to accurate orientate a single triangle, if this were the intended mapping geometry, means that the methodology naturally also possesses the capability to orientate any number of triangles describing the intended mapping geometry. As an increased complexity of mapping geometry will obviously require a larger number of triangles to describe it, the mapping methodology is therefore capable of mapping any complexity of geometry described as STL data.

Combining the results of the validation experiments indicates that the methodology developed is indeed capable of mapping any level of geometric complexity described as STL data accurately to surface mesh structure with a correct orientation. The methodology has been shown to be robust when mapping to both nodes and element mid-points, and on both 2 and 3D surface meshes.

These capabilities therefore provide all the functionality required by the mapping methodology for the efficient generation of complex and conformal RM textiles.
Therefore, when utilised with a surface mesh generated by the intersecting-sphere techniques, the mapping methodology will allow the generation of uniform and tolerance-based equidistant conformal RM textile data suitable for manufacture.

The validation of the mapping tool therefore concludes the research and experimental work undertaken in this doctoral thesis. The next chapter will therefore be a final discussion of the research undertaken and the main conclusions for this work achieved.
Chapter 13: Final conclusions and Recommendations for Further work

13.1 Attained research objectives

The research contained within this doctoral thesis has been undertaken with the intention of addressing the specified research objectives documented in chapter 6 and presented again here:

Research Issue 1:
1. Extended literature review of current surface meshing techniques and related issues
2. Investigation of uniform and equidistant meshing
3. Potential testing and validation of any existing or newly developed meshing technique

Research Issue 2:
1. Investigation of complex geometry mapping
2. Potential testing and validation of any existing or newly developed mapping tool

Addressing each section individually, the research undertaken for the objectives in research issue 1 has produced a review of current surface meshing capabilities and ascertained that no current system was capable of producing the quality of mapping mesh required for the generation of uniform and conformal RM textile data. The requirement of the mapping mesh therefore resulted in the investigation and development of a new meshing algorithm designed especially for this task. The subsequent creation of an experimental meshing system from the developed algorithm enabled experimental testing and validation of the new meshing algorithms capabilities to be undertaken. Therefore, it can be concluded that all of the research objectives for section one have been successfully attained.
The research objectives within research issue 2 relate to the research problem of 3D complex geometry mapping. The research for these objectives has produced a detailed mathematical investigation of mapping 3D complex geometry to a surface mesh, resulting in a complete mapping methodology. The coding of the newly developed mapping methodology into the stand-alone mapping tool, ChainLink, has enabled the mapping methodology to be tested and validated. Therefore, it can again be concluded that all of the research objectives for section two have been successfully attained.

The research has suggested that combining the newly developed meshing algorithm with the newly developed mapping tool, an improved methodology for the efficient generation of conformal RM textiles of a high geometric complexity can be produced. While not actually achieved in this thesis, the research undertaken, and the systems created, provide the tools necessary to achieve this aim and therefore provide a substantial contribution to the research area of RM textiles.

13.2 Conclusions

The following text highlights the main conclusions that can be taken from this work.

- Intelligent and high performance textiles are increasingly seen as relevant research areas which potentially provide the platform for the integration of technologies and functionalities desired by modern society into our local and daily environment, and provide the first step for the creation of truly wearable computers. However, the current manufacture of intelligent and high performance textiles requires a very long and arduous manufacturing process that are ultimately limiting their complexity and actual creation, requiring a more suitable manufacturing method to be identified.

- The application of additive manufacturing to intelligent and high performance textiles provides an elegant manufacturing solution to the limitations being currently observed, where the issues of DFMA and limited geometric complexity currently experienced can be potentially resolved through the generation of RM textiles. Current RM textiles rely on the individual linking of separate geometric structures and
therefore exhibit an entirely different structure to that of conventional fibre based textiles. This shift away from fibres enables a much higher level of geometric complexity to be achieved as the base structure in a textile hierarchy.

- The major issue preventing the production of RM textiles is that current CAD systems do not have the inherent functionality required for the efficient generation of conformal RM textile data. Also, current CAD systems are being pushed to their functional limit when considering the manufacturing complexity achievable by additive manufacturing. Whilst experimental Textile CAD systems such as TexGen do provide some functionality for the generation of conformal RM textile data, they are extremely limited in the geometric complexity achievable for individual links and in the uniformity of the link distribution.

- A methodology for the generation of RM textile data was proposed that would allow the mapping of 3D geometry to a surface mesh in order to create the RM textile structure. In order to do this, two main requirements were identified:
  1. The ability to create a uniform and equidistant mesh to ensure the consistency of the link positions
  2. The ability to map complex 3-D geometries to a mesh structure.

- The intersecting-sphere methodology developed in this research has been shown to be capable of producing a uniform and near-equidistant mesh over a variety of curvatures of surfaces. The generation of such a uniform and equidistant surface mesh structures has also been shown to be unattainable using current commercial and experimental meshing techniques. In this respect the research described here therefore provides a major advance, and is, to the author’s knowledge, the only meshing system capable of generating a uniform and tolerance-based equidistant mesh structure suitable as a mapping mesh for the generation of conformal RM textile data.
The development of ChainLink has enabled the accurate and efficient mapping of complex geometric data described in STL format to a surface mesh structure. Combining the intersecting-sphere meshing system with the capabilities of ChainLink will offer the only technique at present that potentially allows the generation of uniform and almost equidistant conformal RM textile data to be accurately and efficiently created for subsequent manufacture.

As with any research undertaken, there remain some areas of this particular topic which could benefit from further work. These issues will be discussed in the following section.

13.3 Recommendations for further work

Based on the research and findings of this thesis, the areas of RM textiles that could benefit from further research and investigation are listed below:

- Integration of the intersecting-sphere meshing technique with ChainLink
  At the time of submission for this thesis, a fully working interface between the new meshing technique and the mapping tool had not been established. Therefore the actual creation of uniform and near equidistant conformal RM textile data could not be generated. Combining the mapping and meshing tools will provide the full capability for modelling RM textiles.

- Intersecting-sphere meshing technique
  While the experimental meshing system was proved to generate the required mesh structures for the generation of conformal RM textile data for the sample surfaces tested, the system is currently limited to the meshing of quadratic surface patches that can be analytically described in Matlab. To fully explore the capabilities of the new meshing system, a more robust meshing system needs to be developed where surface geometry created in CAD or captured by body scanning techniques can be imported. This will ultimately require a standalone version of the meshing technique to be created.
The new meshing system developed also has a potential range of different applications, where the creation of a uniform and almost equidistant mesh structures is desirable. Therefore, further applications of this new meshing technique should be investigated.

- **ChainLink**
The mapping methodology employed by ChainLink was proved to be very robust and capable of generating complex RM textile data. However, other applications of mapping complex geometric data to mesh structures may exist and should be investigated. Such applications include the generation of RM macro and micro textures.

- **Dataset**
The future creation of conformal RM textiles at a reduced resolution will obviously create larger datasets. If it is theorised that the resolution of additive manufacturing capabilities will improve to the stage where it can compete with the current resolution of conventional textiles; this will result in extremely large RM textile datasets. Therefore, before this manufacturing resolution is achieved, innovative techniques need to be established to handle such large datasets efficiently.

- **Efficient manufacture**
The capability to generate uniform and conformal textiles does not mean that they have to be manufactured in their final intended state. An efficient manufacturing method would be to utilise the intended properties of the RM textile and fold or collapse the textile structure into its lowest possible potential energy state, thus maximising the build volume of additive manufacturing systems. Initial investigations have been undertaken in this area by the author and a preliminary methodology established. However, further work is required to extend this methodology to the complexity and size of RM textile structures that can now be potentially generated.

- **Design of RM textile link structures**
As the systems investigated in this thesis provides the potential for generating conformal RM textile data it now becomes possible to investigate the actual design of RM textile
link structures to deliver desirable functionality. Further work is required to understand the relationship between RM textile link resolution and design and the macro level functionality it can therefore generate.

- **Applications of RM textiles**
  While the application of high performance and intelligent textiles is an immediate application of RM textiles, further research is required to ascertain other potential applications of this new textile manufacturing technique.

- **RM processes, build materials and resolution**
  The work undertaken in this thesis has revolved around the current RP processes that are available and the potential capabilities of future RM system. However, it may be required that a dedicated textile additive manufacturing system be developed. Any new system would require suitable build materials and resolution capabilities that would allow the true scope of RM textile to be realised. These requirements and the design of such new additive manufacturing systems could also be investigated.
References


229


[38] Zhang, X. and Tao, X., Smart textiles: passive smart, Textile Asia, pp 45-49, 2001
[40] Zhang, X. and Tao, X., Smart textiles: very smart, Textile Asia, pp 35-37, 2001


[64] Limaye, A., Design and analysis of a mask projection microstereolithography system, MSc Thesis, Georgia Institute of Technology, 2004


[69] UK Manufacturing: We can make it better, Foresight Manufacturing 2020 Panel, Final Report, Findlay Publications


[75] EOSINT M hardware evaluation, Prototype magazine, v8, n3, pp 26-28, 2005


232
Gady, B., *Upgrading Readiness: Successes and Improvements of the Mobile Parts Hospital*, Defense Technical Information Center, Paper no. 05MV-4, 2005


Brown, D., Morgan, M. and Mcllhagger, R., A system for the automatic generation of solid models of woven structures, Composites Part A, v34, pp 511-515, 2003


Designed and created by Guy Bingham, Rapid Manufacturing Research Group, Loughborough University, 2006


Owens, S., A survey of unstructured mesh generation technology, Proceedings of the 7th International Meshing Roundtable, Dearborn, pp 239-267, 1998


Lohner, R., Progress in grid generation via the advancing front technique, Engineering with Computers, v12, pp 186-210, 1996


235


237


[175] Thomson, J., On the structure of the atom: an investigation of the stability and periods of oscillation of a number of corpuscles arranged at equal intervals around the circumference of a circle; with application of the results to the theory of atomic structure, Philosophical Magazine s6, v7, pp 237-265, 1904


238
Appendix A
% WORKING 3D Sphere Packing Algorithm by G Bingham
%

% Function list:  
% Firstspherecreation3D  
% Secondspherecreation3D  
% Nthspherecreation3D  
% radiusboundary  
% calculateunique  
% clustersphereboundary  

% DATA STRUCTURE:  
% D.poly = Polynomial coefficients  
% D.radius = Sphere radius  
% D.centres = xyz sphere centre  
% D.sph2surfinter = Sphere to surface intersection spline  
% D.sph2surfinter2D = 2D sphere to surface intersection curve  
% D.inter = Cross over sites of 2D intersections (new sphere sites)
%

% Stage one. Surface creation
%

clear all;  
clf;  

SURFACE 1: Planar
%  
% X axes control points  
% Z points at x axis control points  
% Polynomial creation from points: Power^4
%  
% Y axes control points
%  
% Z points at y axes control points  
% Polynomial creation from points: Power^4
%  
% SURFACE 2: Cylinder
%  
% X axes control points  
% Z points at x axis control points  
% Polynomial creation from points: Power^4  
%  
% Y axes control points  
% Z points at y axes control points  
% Polynomial creation from points: Power^4
%  
% SURFACE 3: Parabolic surface
%  
% X axes control points  
% Z points at x axis control points  
% Polynomial creation from points: Power^4
%  
% Y axes control points  
% Z points at y axes control points  
% Polynomial creation from points: Power^4
%  
% SURFACE 4: inverse Parabolic surface
%  
% X axes control points  
% Z points at x axis control points  
% Polynomial creation from points: Power^4
%  
% Y axes control points  
% Z points at y axes control points
% py=polyfit(yd,zyd,4); % Polynomial creation from points; Power^4

% SURFACE 5: Saddle
% x=[0 12.5 25 37.5 50];
% z=[0 5.35 7.5 3.5 0];
% px=polyfit(x,zd,4); % Polynomial creation from points: Power^4
% yd=[0 12.5 25 37.5 50];
% zyd=[0 -5.35 -7 -5.35 0];
% py=polyfit(yd,zyd,4);

% % SURFACE 6: Peaked
% x=[0 12.5 25 37.5 50];
% z=[0 7.5 5 7.5 0];
% px=polyfit(x,zd,4); % Polynomial creation from points: Power^4
% yd=[0 12.5 25 37.5 50];
% zyd=[0 7.5 5 7.5 0];
% py=polyfit(yd,zyd,4);

xaxis=50; yaxis=50;

%%%% Surface creation %%%
% The surface equation is therefore:
% z=px(1)'x^4 + py(1)'y^4 + px(2)'x^3 + py(2)'y^3 + px(3)'x^2 + py(3)'y^2 + px(4)'x + py(4)'y;
i=1; % Counter set to 1
a=1; % Increments of the x and y axes for surface
for x=0:a:xaxis; % Count through x axis
    for y=0:a:yaxis; % Count through y axis
        z(i,:)=px(1)'x^4 + py(1)'y^4 + px(2)'x^3 + py(2)'y^3 + px(3)'x^2 + py(3)'y^2 + px(4)'x + py(4)'y;
        i=i+1; % Increase counter by 1
    end
end

%%%% Plot the surface %%%
x=0:a:xaxis; % X axis matrix creation
y=0:a:yaxis; % Y axis matrix creation
lx=length(x); % Lenght of x matrix
ly=length(y); % Length of y matrix
znew=reshape(z,ly, lx); % Reshape the z matrix
MSD=surf(x,y,znew); % Plot the surface
shading interp; % Removes surface lines
set(MSD,'facecolor','yellow'); % Set surface parameters
camlight; % Lighting
lighting phong; % Rendering
view(45,30); % Viewing angle
hold on; % Hold the data
grid on;

%%%% Stage two. Data structure creation %%%

% c=1; % Set counter to 1
D(c).poly(1,:)=[px(1),px(2),px(3),px(4)];
% Set D.poly = xz polynomial coefficients
D(c).poly(2,:)= [py(1), py(2), py(3), py(4)];

% Set D.poly = yz polynomial coefficients
%!

D(c).radius=5; % Data structure radius creation

x0=25; % X axis centre point of first sphere
y0=25; % y axis centre point of first sphere
z0=px(1)*x0^4 + py(1)*y0^4 + px(2)*x0^3 + py(2)*y0^3 + px(3)*x0^2 + py(3)*y0^2 + px(4)*x0 + py(4)*y0;

D(c).centres=[x0, y0, z0];
D(c).dataflag=1;

clear MSD a i ly x x0 xd y y0 yd z z0 znew zxd zyd

% Stage three. First sphere creation

Firstspherecreation3D(D); % Call first sphere creation function
D=ans; % Returned answer = data structure

% Second sphere creation

Secondspherecreation3D(D); % Call the Second create sphere function
D=ans; % Returned answer = data structure

% Third and Nth Sphere creation

tic % TIMER - START
D(2).newinter=[D.inter];
for n=1:1 % Predetermined number of cycles
n % Show cycle number
Nthspherecreation3D(D); % Call the Nth sphere creation
D=ans; % Returned answer = data structure
clear ans
c=length(D); % Length of data structure
% D(c).cylecentres=c; % Mark cycle number
if D(c).dataflag==0; % If the dataflag is empty
    break % Break the cycle, mesh complete.
end

% if (n==2)(n==10) |(n==20) |(n==30) |(n==40) |(n==50) |(n==60) |(n==70) |(n==80) |(n==90) |(n==100);
% Mcheck=input('#### Continue meshing? 1 = YES, 0 = NO ====> ');
% if Mcheck==1;
%  continue
% elseif Mcheck==0; % THIS SECTION ALLOWS THE ALGORITHM
% break % TO STOP AFTER CERTAIN CYCLES TO
% else % CHECK THE MESH
% end % end

% THIS SECTION ALLOWS THE ALGORITHM TO STOP AFTER CERTAIN CYCLES TO CHECK THE MESH

cwd = pwd;
cd(tempdir);
pack
cd(cwd)
end
%
% Create the mesh, visual
%
% clear D.poly D.radius D.dataflag D.sph2surfinter D.sph2surfinter2D D.inter D.newinter
r=D(1).radius; % Data structure radius creation
figure(2) % Call another figure window
clf; % Clear an potential data
temp=[D.centres]; % Collect sphere centres
dtemp,ltemp]=size(temp); % Size of centres matrix
temp1=reshape(temp,3,(ltemp/3));% Reshape centres matrix
compcentres=temp1'; % Transpose the matrix
compcentres=unique(compcentres,'rows');
clear temp dtemp temp1 D

plot3(compcentres(:, 1), compcentres(:,2), compcentres(:,3), 'black'); % Plot the centres
grid on % Grid on
hold on; % Hold the data
[dcomp,lcomp]=size(compcentres); % Size of new centre matrix
%
% Determine if one centre point is less then the radius away from another % centre point by using the sphere equation
c=1;
for i=1:dcomp;
    for j=1:dcomp;
        diff=(compcentres(i, 1)-compcentres(j,1))^2+(compcentres(i,2)-compcentres(j,2))^2+(compcentres(i,3)-compcentres(j,3))^2;
        if diff==0;
            break;
        else diff1=sqrtm(diff);
            if (diff1<=(r+(r*.4)))&(diff1>=(r*.75)); % original criteria
                elements(c,:)=diff1;
                c=c+1;
                x=[compcentres(i,1);compcentres(j,1)];
                y=[compcentres(i,2);compcentres(j,2)];
                z=[compcentres(i,3);compcentres(j,3)];
                plot3(x,y,z);
            end
        end
    end
end
clear Mcheck c dcomp diff1
All centres connected to relevant centres completing the mesh

### Second surface creation ###
i=1; % Counter set to 1
a=1; % Increments of the x and y axes for surface
for x=0:a:xaxis; % Count through x axis
for y=0:a:yaxis; % Count through y axis
z(i,:)=px(1)*x^4 + py(1)*y^4 + px(2)*x^3 + py(2)*y^3 + px(3)*x^2 + py(3)*y^2 + px(4)*x + py(4)*y;
% For each xy point create a z point
i=i+1; % Increase counter by 1
end
end

### Plot the surface ###
x=0:a:xaxis; % X axis matrix creation
y=0:a:yaxis; % Y axis matrix creation
lx=length(x); % Length of x matrix
ly=length(y); % Length of y matrix
znew=reshape(z,lx,ly); % Reshape the z matrix
znew=znew-.25; % Removes surface lines
MSD1=surf(x,y,znew); % Plot the surface
shading interp; % Set surface parameters
camlight; % Lighting
lighting phong; % Rendering
view(25,60); % Viewing angle
hold on; % Hold the data
grid on;

% Meshing complete

MAXELEMENT=max(elements); % Size of smallest element
MINELEMENT=min(elements); % Size of largest element
MEANELEMENT=mean(elements); % Mean of element
toc % TIMER - END
disp('========== Meshing Complete =========='
MAXELEMENT
MEANELEMENT
figure;
grid on;
hold on;
xvalues=MINELEMENT-1:0.1:MAXELEMENT+1;
hist(elements,xvalues);
h = findobj(gca,'Type','patch');
set(h,'FaceColor','yellow','EdgeColor','black')

function[D]=Firstspherecreation3D(D)
Function: Create the first sphere intersection of the surface

Stage One. Sphere to Surface intersection

The intersection works using the spherical coordinates of a sphere and the algebraic equation of the surface. Counting through all 180 degrees of the colatitude for every latitude degree (1 to 180). The system employs an iterative search to narrow down to a tolerance of degree the x,y,z coordinate that correlate with the algebraic equation.

c=1; % Set counter to 1
xO=[O(c).centres(1,1)]; % X axis sphere centre
yO=[O(c).centres(1,2)]; % Y axis sphere centre
zO=[O(c).centres(1,3)]; % Z axis sphere centre
r=O(c).radius; % Radius of sphere
px=D(c).poly(1,:); % XZ coefficients
py=D(c).poly(2,:); % YZ coefficients

%## Stage 1. ##% 360degree window
s=1; % Set counter to 1
for a=0:10:360; % Start latitude angle
for b=O:10:180; % Start colatitude angle
tol=0.0001; % Tolerance of the intersection
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp); % Z coordinate at spherical x and y equation point
if zsp<=zsurfcal % If the spherical z is less than
    cb=b; % the z surface point, record angle
    break % New start angle
end
%## Stage 2. ##% 10degree window
for b=cb:-1:(cb-10); % New start angle
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp); % Z coordinate at spherical x and y equation point
if zsp>=zsurfcal % If the spherical z is less than
    cb=b; % the z surface point, record angle
    break; % New start angle
end
%## Stage 3. ##% 1degree window
for b=cb:0.1:(cb+1);
xsp=x0+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=y0+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=z0+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
% Z coordinate at spherical x and % y equation point
if zsp<=zsurfcal % If the spherical z is less than % the z surface point, record angle
cb=b; % New start angle
break;
end
end

%## Stage 4. ##% 0.1degree window
for b=cb:-0.01:(cb-0.1);
xsp=x0+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=y0+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=z0+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
% Z coordinate at spherical x and % y equation point
if zsp>=zsurfcal % If the spherical z is less than % the z surface point, record angle
cb=b; % New start angle
break;
end
end

%## Stage 5. ##% 0.01degree window
for b=cb:0.001:(cb+0.01);
xsp=x0+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=y0+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=z0+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
% Z coordinate at spherical x and % y equation point
if zsp<=zsurfcal % If the spherical z is less than % the z surface point, record angle
cb=b; % New start angle
break;
end
end

%## Stage 6. ##% 0.001degree window
for b=cb:-0.001:(cb-0.001);
xsp=x0+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=y0+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=z0+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
% Z coordinate at spherical x and % y equation point
if abs(zsp-zsurfcal)<=tol % If spherical z minus zsurface is less % than the tolerance, intersection
inter(s,:)=[xsp,ysp,zsurfcal]; % Inter is equal to xy spherical an z surface
s=s+1; % increase counter by 1
break;
end
end

% Stage two. Creation of spline from intersection calculation

% Option to plot the calculated intersection points

% plot3(inter(:,1),inter(:,2),inter(:,3),'k*')
% Plot the points in black

% Check the intersection matrix

if isempty(inter)
    disp([' ===== No intersection found, ERROR ====='])
    return
end
% Create complete intersection matrix

xint=inter(:,1)'; % Transpose x coordinates of intersection matrix
yint=inter(:,2)'; % Transpose y coordinates of intersection matrix
zint=inter(:,3)'; % Transpose z coordinates of intersection matrix
xyzint=[xint;yint;zint]; % Complete matrix for spline creation
fnplt(cscvn(xyzint(:,[1:end 1]))); % plot spline from complete matrix
A=Aspline'; % Transpose the spline points into xyz format
D(c).sph2surfinter=A; % Create sphere to surface intersection points

% Stage three. Creation of 2D curve

% Create a 2D curve of the intersection

% plot2(A(:,1),A(:,2)) % Plot the 2D sphere to surface intersection
D(c).sph2surfinter2D=[A(:,1),A(:,2)];% Create 2D sphere to surf intersection curve
function[D]=Secondspherecreation3D(D)

%---------------------------------------------------------------
% Function: Create the second sphere intersection of the surface
%---------------------------------------------------------------

% Stage One. Sphere to Surface intersection
%---------------------------------------------------------------

% The intersection works using the spherical coordinates of a sphere and
% the algebraic equation of the surface. Counting through all 180 degrees of
% the colatitude for every latitude degree (1 to 180). The system employs a
% iterative search to narrow down to a tolerance of degree the x,y,z coordinate
% that correlate with the algebraic equation.

% c=2; % Set counter to 2
D(c).centres(1,1)=D(c-1).sph2surfinter(1,1); % New x axis centre point
D(c).centres(1,2)=D(c-1).sph2surfinter(1,2); % New y axis centre point
D(c).centres(1,3)=D(c-1).sph2surfinter(1,3); % New z axis centre point
D(c).radius=D(c-1).radius; % Radius of sphere
r=D(c).radius; % Radius of sphere
xO=[D(c).centres(1,1)]; % XO
yO=[D(c).centres(1,2)]; % YO
zO=[D(c).centres(1,3)]; % ZO
D(c).poly(1,:)=D(c-1).poly(1,:); % XZ coefficients
D(c).poly(2,:)=D(c-1).poly(2,:); % YZ coefficients
px=D(c).poly(1,:); % XZ coefficients
py=D(c).poly(2,:); % YZ coefficients
D(c).dataflag=D(1).dataflag;

%## Stage 1. ##% 360degree window
s=1; % Set counter to 1
for a=0:10:360; % latitude angle
for b=0:10:180; % Start colatitude angle
tol=0.0001; % Tolerance of the intersection
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
znurfal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
% Z coordinate at spherical x and % y equation point
if zsp<=znurfal % If the spherical z is less than
% the z surface point, record angle
cb=b; % New start angle
break
end
end
%## Stage 2. ##% 10degree window
for b=cb:-1:(cb-10); % New start angle
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
\[ zsp = z_0 + r \cos d(b); \] % Spherical coordinate z equation
\[ zsurfcal = (p_1 xsp^4 + p_2 ysp^4 + p_3 xsp^3 + p_4 ysp^3 + p_5 xsp^2 + p_6 ysp^2 + p_7 xsp + p_8 ysp); \] % Z coordinate at spherical x and y equation point
if \( zsp \geq zsurfcal \) % If the spherical z is less than
\( \text{the z surface point, record angle} \)
\( cb = b; \) % New start angle
break;
end

\[ \text{%## Stage 3. ##}\]
for \( b = cb:0.1:(cb+1) \);
\( xsp = x_0 + r \cos d(a) \sin d(b); \) % Spherical coordinate x equation
\( ysp = y_0 + r \sin d(a) \sin d(b); \) % Spherical coordinate y equation
\( zsp = z_0 + r \cos d(b); \) % Spherical coordinate z equation
\( zsurfcal = (p_1 xsp^4 + p_2 ysp^4 + p_3 xsp^3 + p_4 ysp^3 + p_5 xsp^2 + p_6 ysp^2 + p_7 xsp + p_8 ysp); \) % Z coordinate at spherical x and y equation point
if \( zsp \leq zsurfcal \) % If the spherical z is less than
\( \text{the z surface point, record angle} \)
\( cb = b; \) % New start angle
break;
end

\[ \text{%## Stage 4. ##}\]
for \( b = cb:-0.01:(cb-0.1) \);
\( xsp = x_0 + r \cos d(a) \sin d(b); \) % Spherical coordinate x equation
\( ysp = y_0 + r \sin d(a) \sin d(b); \) % Spherical coordinate y equation
\( zsp = z_0 + r \cos d(b); \) % Spherical coordinate z equation
\( zsurfcal = (p_1 xsp^4 + p_2 ysp^4 + p_3 xsp^3 + p_4 ysp^3 + p_5 xsp^2 + p_6 ysp^2 + p_7 xsp + p_8 ysp); \) % Z coordinate at spherical x and y equation point
if \( zsp \geq zsurfcal \) % If the spherical z is less than
\( \text{the z surface point, record angle} \)
\( cb = b; \) % New start angle
break;
end

\[ \text{%## Stage 5. ##}\]
for \( b = cb:0.001:(cb+0.01) \);
\( xsp = x_0 + r \cos d(a) \sin d(b); \) % Spherical coordinate x equation
\( ysp = y_0 + r \sin d(a) \sin d(b); \) % Spherical coordinate y equation
\( zsp = z_0 + r \cos d(b); \) % Spherical coordinate z equation
\( zsurfcal = (p_1 xsp^4 + p_2 ysp^4 + p_3 xsp^3 + p_4 ysp^3 + p_5 xsp^2 + p_6 ysp^2 + p_7 xsp + p_8 ysp); \) % Z coordinate at spherical x and y equation point
if \( zsp \leq zsurfcal \) % If the spherical z is less than
\( \text{the z surface point, record angle} \)
\( cb = b; \) % New start angle
break;
end
for b=cb:-0.0001:(cb-0.001);
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
% Z coordinate at spherical x and y equation point
if abs(zsp-zsurfcal)<=tol % If spherical z minus zsurface is less than the tolerance, intersection % found
    inter(s,:)=[xsP,ysP,zsurfcal]; % Inter is equal to xy spherical and z surface
    s=s+1; % increase counter by 1
    break;
end
end

% Stage two. Creation of spline from intersection calculation
%
% ## Option to plot the calculated intersection points ## %
% plot3(inter(:,1),inter(:,2),inter(:,3),'k*') % Plot the points in black
% ## Check the intersection matrix ## %
% if isempty(inter)
%    disp([' ===== No intersection found, ERROR ====='])
%    return
% end
% ## Create complete intersection matrix ## %
% ## Create data for spline creation ## %
% xint=inter(:,1)'; % Transpose x coordinates of intersection matrix
% yint=inter(:,2)'; % Transpose y coordinates of intersection matrix
% zint=inter(:,3)'; % Transpose z coordinates of intersection matrix
% xyzint=[xint;yint;zint]; % Complete matrix for spline creation
% fnplt(cscvn(xyzint(:,[1:end 1]))); % plot spline from complete matrix
% Aspline=fnplt(cscvn(xyzint(:,[1:end 1]))); % Create spline points
% A=Aspline'; % Transpose the spline points into xyz format
% D(c).sph2surfinter=A; % Create sphere to surface intersection points
%
% Stage three. Creation of 2D curve
%
% ## Create a 2D curve of the intersection ## %
% plot(A(:,1),A(:,2)) % Plot the 2D sphere to surface intersection
% D(c).sph2surfinter2D=[A(:,1),A(:,2)]; % Create 2D sphere to surf intersection curve
% [xinter,yinter]=polyxpoly([D(1).sph2surfinter2D(:,1)], [D(1).sph2surfinter2D(:,2)],
% [D(c).sph2surfinter2D(:,1)], [D(c).sph2surfinter2D(:,2)],'unique');
% Lxinter=length(xinter);
% s=1; % Set counter to 1
% for i=1:Lxinter
zinter(s,:)=(px(1)*xinter(i,:)^4 + py(1)*yinter(i,:)^4 + px(2)*xinter(i,:)^3 + py(2)*yinter(i,:)^3 +
px(3)*xinter(i,:)^2 + py(3)*yinter(i,:)^2 + px(4)*xinter(i,:) + py(4)*yinter(i,:));
s=s+1;
end

% Calculate the z points on the surface
% at xinter and yinter
D(c).inter=[xinter,yinter,zinter];  % Create intersection sites
plot3(xinter,yinter,zinter,'r*');  % Plot the intersection sites
function [D] = Nthspherecreation3D(D);
% % Nth Sphere Creation function
% Stage 1. determine if intersections are valid.
% %
% [dD,ID]=size(D); % Size of the data structure
compinter=[]; % Define new matrix
% calculateinter(D); % FUNCTION, collects all intersection sites
% compinter=ans; % complete intersection matrix
compinter=[D(ID).newinter]; % Modified complete intersection matrix
if isempty(compinter) % If there are no intersections
    D(ID).dataflag=0; % Data flag is 0, return to main code, end meshing
    return
else
    D(ID).dataflag=1;
end
calculateunique(compinter); % FUNCTION, calculates unique intersections
compinter=ans; % complete unique intersection matrix
clear ans
% Collect all the data centres
temp=[D.centres]; % Centres matrix
[dtemp,ltemp]=size(temp); % Size of centres matrix
temp1=reshape(temp,3,(ltemp/3)); % Reshape the centres matrix
compcentres=temp1'; % Transpose the matrix
clear dtemp temp temp1
% This is an edit, may 18th 2007
% temp=[D.sph2surfinter2D]; % 2D intersection points matrix
% [dtemp,ltemp]=size(temp); % Size of 2D intersection matrix
% temp1=reshape(temp,2,(ltemp/2)); % Reshape the 2D intersection matrix
% comp2Dpoints=temp1'; % Transpose the matrix
px=D(1).poly(1,:); % XZ coefficients
py=D(1).poly(2,:); % YZ coefficients
sphereradii=D(1).radius; % Radius of packing spheres
% FUNCTION, defines points close to sphere centre and removes them
% Version 1: circleboundary
% Version 2: sphereboundary
% Version 3: radiusboundary
radiusboundary(compinter,compcentres,sphereradii);
compinter1=ans; % valid intersection matrix
clear compinter
% sphereboundary(compinter,compcentres,px,py,sphereradii);
% circleboundary(compinter,compcentres);
% Original function, replaced with above
% calculatesetdiff(compinter,compcentres);
% FUNCTION, determines the intersections
% that are not already sphere centres
clustersphereboundary(compinter1,px,py,sphereradii);
compinter1=ans; % valid intersection matrix
clear ans
% clusterboundary(compinter1,px,py);
% Version 1: clusterboundary
% Version 2: clustersphereboundary
FUNCTION, defines points that are
% within a cluster and calculates a
% triangulated average new point

% #-----------------------------------------------------------------------------------------------
% % Determine if the intersection sites are within the surface boundary
% [dcompi,lcompi]=size(compinter1);
c=1;
vsites=[];
for i=1:dcompi
    if
        (compinter1(i,1)>=[0.00])&(compinter1(i,1)<=[50])&(compinter1(i,2)>=[0.00])&(compinter1(i,2)<=[50]);
        vsites(c,:)=compinter1(i,:);
c=c+1;
    end
end
if isempty(vsites);
    D(ID).dataflag=0;
    return
else D(ID).dataflag=1;
end
% % All valid sites calculates, plot valid sites
plot3(vsites(:,1),vsites(:,2),vsites(:,3),"r")
clear compinter1
%-----------------------------------------------------------------------------------------------
% % At this point all the validsites have been defined. I'm going to leave
% % this here for the moment so all the validsites will be updates in the
% % command window while meshing. I can then see if its failing or remeshing
% % validsites
%-----------------------------------------------------------------------------------------------
% % Stage 2. Plot spheres at valid sites
%-----------------------------------------------------------------------------------------------
% [dvsites,lvsites]=size(vsites);    % Size of valid site matrix
% c=ID+1;               % Set counter to length of data structure +1
% % Plot sphere for all valid sites
% % for i=1:dvsites;
% D(c).centres(1,1)=vsites(i,1);    % Data centre x coordinate
% D(c).centres(1,2)=vsites(i,2);    % Data centre y coordinate
% D(c).centres(1,3)=vsites(i,3);    % Data centre z coordinate
% D(c).radius=D(1).radius;         % Data radius radius
% x0=[D(c).centres(1,1)];          % Sphere x coordinate
% y0=[D(c).centres(1,2)];          % Sphere y coordinate
% z0=[D(c).centres(1,3)];          % Sphere z coordinate
% r=D(c).radius;                    % Sphere radius
% D(c).poly(1,:)=D(1).poly(1,:);   % XZ coefficients
% D(c).poly(2,:)=D(1).poly(2,:);   % YZ coefficients
% px=D(c).poly(1,:);               % XZ coefficients
% py=D(c).poly(2,:);               % YZ coefficients
%## Stage 1. ##% 360deg window
s=1; % Set counter to 1
for a=0:10:360; % Latitude angle
for b=0:10:180; % Start colatitude angle
tol=0.0001; % Tolerance of the intersection
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp); % Z coordinate at spherical x and y equation point
if zsp<=zsurfcal % If the spherical z is less than
    cb=b; % the z surface point, record angle
    break % New start angle
end
end
%## Stage 2. ##% 10deg window
for b=cb:-1:(cb-10); % New start angle
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp); % Z coordinate at spherical x and y equation point
if zsp>=zsurfcal % If the spherical z is less than
    cb=b; % the z surface point, record angle
    break; % New start angle
end
end
%## Stage 3. ##% 1deg window
for b=cb:0.1:(cb+1); % New start angle
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp); % Z coordinate at spherical x and y equation point
if zsp<=zsurfcal % If the spherical z is less than
    cb=b; % the z surface point, record angle
    break; % New start angle
end
end
%## Stage 4. ##% 0.1deg window
for b=cb:-0.01:(cb-0.1); % New start angle
xsp=xO+r*cosd(a)*sind(b); % Spherical coordinate x equation
ysp=yO+r*sind(a)*sind(b); % Spherical coordinate y equation
zsp=zO+r*cosd(b); % Spherical coordinate z equation
zsurfcal=(px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp); % Z coordinate at spherical x and y equation point
if zsp >= zsurfcal
    cb = b;
    break
end

% y equation point
if the spherical z is less than the z surface point, record angle

% New start angle

%% Stage 5. 0.01 degree window
for b = cb:0.001:(cb+0.01);
    xsp = xO + r*cosd(a)*sind(b);
    % Spherical coordinate x equation
    ysp = yO + r*sind(a)*sind(b);
    % Spherical coordinate y equation
    zsp = zO + r*cosd(b);
    % Spherical coordinate z equation
    zsurfcal = (px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
    % Z coordinate at spherical x and y equation point
    if zsp <= zsurfcal
        cb = b;
        break
    end
end

%% Stage 6. 0.001 degree window
for b = cb:-0.0001:(cb-0.001);
    xsp = xO + r*cosd(a)*sind(b);
    % Spherical coordinate x equation
    ysp = yO + r*sind(a)*sind(b);
    % Spherical coordinate y equation
    zsp = zO + r*cosd(b);
    % Spherical coordinate z equation
    zsurfcal = (px(1)*xsp^4 + py(1)*ysp^4 + px(2)*xsp^3 + py(2)*ysp^3 + px(3)*xsp^2 + py(3)*ysp^2 + px(4)*xsp + py(4)*ysp);
    % Z coordinate at spherical x and y equation point
    if abs(zsp - zsurfcal) <= tol
        % If spherical z minus zsurface is less than the tolerance, intersection found
        inter(s,:) = [xsp, ysp, zsurfcal]; % Inter is equal to xy spherical an z surface
        s = s + 1; % Increase counter by 1
        break;
    end
end
cwd = pwd;
cd(tempdir);
pack
cd(cwd)

% Stage two. Creation of spline from intersection calculation

% Option to plot the calculated intersection points
% plot3(inter(:,1),inter(:,2),inter(:,3),k*)
% Plot the points in black

% Check the intersection matrix
if isempty(inter)
    disp([' ===== No intersection found, ERROR ====='])
    return
end
% # Create complete intersection matrix %
% Create data for spline creation %
xint = inter(:,1)';  % Transpose x coordinates of intersection matrix
yint = inter(:,2)';  % Transpose y coordinates of intersection matrix
zint = inter(:,3)';  % Transpose z coordinates of intersection matrix
xyzint = [xint;yint;zint];  % Complete matrix for spline creation
fnptt(cscvn(xyzint(:,1:end 1)));  % plot spline from complete matrix
Aspline = fnplt(cscvn(xyzint(:,1:end 1)));  % Create spline points
A = Aspline';  % Transpose the spline points into xyz format
D(c).sph2surfinter = A;  % Create sphere to surface intersection points
D(c).sph2surfinter2D = [A(:,1), A(:,2)];  % Create 2D sphere to surf intersection curve
clear A
%
% Sphere created and added to data structure
%
% Determine intersections with new sphere
%
s=1;  % Counter
temp = D.centres;  % Recollect centre sites
dtemp, ltemp = size(temp);  % Size of centres matrix
temp1 = reshape(temp, 3, (ltemp/3));  % Reshape centres matrix
compcentres = temp1';  % Transpose the matrix
dc, lc = size(compcentres);  % Size of new matrix
clear temp1 temp dtemp
for i = 1:dc;
    % Calculate the distance between sphere centres
    diff = (compcentres(i,1) - D(c).centres(1,1))^2 + (compcentres(i,2) - D(c).centres(1,2))^2;
    if diff == 0;
        break;
    end
    diff1 = sqrt(diff);  % Square root of answer = distance
    if (diff1 <= (r+(r*.75)) && (diff1 >= r-(r*.25)));
    end
    % if (diff1 <= (r+r)) && (diff1 >= r-(r/4));
    % Intersection
    [xinter, yinter] = polyxpoly([D(i).sph2surfinter2D(:,1)], [D(i).sph2surfinter2D(:,2)], [D(c).sph2surfinter2D(:,1)], [D(c).sph2surfinter2D(:,2)], 'unique');
    Lxinter = length(xinter);
    T = 1;  % Set counter to 1
    zinter = [];
    % Create new matrix z intersection points
    for i = 1:Lxinter
        zinter(T,1) = (px(1)*xinter(i,:)+py(1)*yinter(i,:)+px(2)*xinter(i,:)+py(2)*yinter(i,:)+px(3)*xinter(i,:)+py(3)*yinter(i,:)+px(4)*xinter(i,:)+py(4)*yinter(i,:));
        T = T + 1;
    end
    % Calculate the z points on the surface
    % at xinter and yinter
    if isempty(zinter)  % If there are no zinter points, go to next
        continue
    end
    Tempxinter(:,s) = xinter;  % Create x intersection matrix
    Tempyinter(:,s) = yinter;  % Create y intersection matrix
    Tempzinter(:,s) = zinter;  % create z intersection matrix
    plot3(xinter, yinter, zinter, 'r*');
s=s+1;
end
end
end
end
%
% All intersection calculate with sphere
%
% Record the intersections and add that to the data structure intersections
%
[dTempx,lTempx]=size(Tempxinter);
% Size of temp intersections (x)
xinter=reshape(Tempxinter,dTempx*lTempx,1);
yinter=reshape(Tempyinter,dTempx*lTempx,1);
zinter=reshape(Tempzinter,dTempx*lTempx,1);
% Reshape the temp intersections
D(c).inter=[xinter,yinter,zinter];
clear xinter yinter zinter Tempxinter Tempyinter Tempzinter
% Add intersection to data structure

c=c+1;
% Increase counter by 1
% Return to valid sites to create a new sphere
end

% This is a new section of code to try and increase the speed of the
% meshing algorithm. Instead of looking through every intersection
% recorded, the algorithm now only looks at the intersection recorded in
% the last cycle.
%
counter=1;
% Set counter to 1
for gg=1:c-1
% from the last cycle number to the present
[dinter,linter]=size(D(gg).inter);
% size of the intersection matrix
for gg=1:dinter;
% from 1 to end of intersection matrix
newinters(counter,:)=D(gg).inter(gg,:);
% Create new intersection matrix
counter=counter+1;
% Increase counter by 1
end
end
D(c-1).newinter=[newinters];
% Record new intersection matrix in Data structure
function [c] = radiusboundary(compinter, compcentres, sphereradii);
% Calculate the intersection sites that are within a defined radius
% around existing sphere centres
%
% a = input data, all of the intersections
% b = input data, all of the sphere centres
% r = tolerance
% r = sphereradii .* 0.8;
% [db, lb] = size(b);
%
% This section of code calculates if a recorder intersection sites is
% within a defined radius distance from an existing sphere centre. If the
% sites is with this range, it is highlighted as invalid and removed from
% the intersection matrix
% c = 1; % Counter
% invalidsites = []; % Invalid sites matrix
% for i = 1: db;
%   [da, la] = size(a);
%   for j = 1: da;
%     diff = (b(i, 1) - a(j, 1))^2 + (b(i, 2) - a(j, 2))^2 + (b(i, 3) - a(j, 3))^2;
%       % Distance between each point
%     diff1 = sqrt(diff);
%       % Square root of distance
%     if diff1 < r;
%       % IF this is less than the tolerance
%       invalidsites(c, :) = a(j, :); % record as invalid
%       c = c + 1;
%     end
%   end
% end
% if isempty(invalidsites);
%   % If the invalid matrix is empty
%   c = a;  % C = output = a = input
% else c = setdiff(a, invalidsites, 'rows');
%   % if invalidsite matrix is not empty
%   % return the sites in that are not
%   % invalid
% end
function[c]=clusterboundary(compinter1,px,py);
%
% Calculate an single valid location from a cluster of points
%
% a=compinter1; % a=input data
c=1; % Counter
r=2; % Radius of circle boundary
[da,la]=size(a); % Size of intersection matrix
%
% This section of code creates a circular 2D polygon around existing sphere
% centres given a radius value (r). For every sphere centre, the code
% generates the boundary points. Once complete, the code then checks if any
% of the intersection points are within the boundary. It then only records
% those that are not.

for i=1:da;
c=1;
if i>da
    break
end

% circlepoints=[];
for j=0:36:360;
circlepoints(c,:)=[(a(i,1)+r*cosd(j),a(i,2)+r*sind(j));
c=c+1;
end

[da,la]=size(a);
pointin=[];
nnewpoints=[];
pointin=inpolygon(a(:,1),a(:,2),circlepoints(:,1),circlepoints(:,2));
nnewpoints=[a(:,1),a(:,2),a(:,3),pointin]
[d,l]=size(newpoints);
c=1;
cluster=[];
for k=1:d;
    if newpoints(k,4)==1 % find the points that are in the polygon
        cluster(c,:)=newpoints(k,1),newpoints(k,2),newpoints(k,3)
c=c+1;
pplot3(cluster(:,1),cluster(:,2),cluster(:,3),'black*')
end
end

[Dclus,Lclus]=size(cluster);
if Dclus==3;
    EI1=[];
    EI2=[];
    EI1=([cluster(2,:)-cluster(1,:))/2]+ cluster(1,:))
    EI2=([cluster(3,:)-cluster(2,:))/2]+ cluster(2,:))
    Line1=[EI1;cluster(3,:)]
    Line2=[EI2;cluster(1,:)]
averagepoint=[];
x0=[];
y0=[];
z=[];
}
\[ x_0, y_0 = \text{polyxpoly(Line1(:,1), Line1(:,2), Line2(:,1), Line2(:,2), 'unique')} \]
\[ z = px(1)*x^4 + py(1)*y^4 + px(2)*x^3 + py(2)*y^3 + px(3)*x^2 + py(3)*y^2 + px(4)*x + py(4)*y; \]
\[ \text{averagepoint} = [x_0, y_0, z] \]
\[ c = 2; \]
\[ \text{for } k1 = 1:d; \]
\[ \text{if newpoints}(k1,4) == 0 \text{ find the points that are in the polygon} \]
\[ \text{averagepoint}(c,:) = [\text{newpoints}(k1,1), \text{newpoints}(k1,2), \text{newpoints}(k1,3)] \]
\[ c = c+1; \]
\[ \text{end} \]
\[ \text{end} \]
\[ a = \text{averagepoint} \]
\[ \text{elseif Dclus == 2;} \]
\[ E11 = []; \]
\[ E11 = [(\text{cluster}(2,:)-\text{cluster}(1,:)/2)+\text{cluster}(1,:)] \]
\[ \text{averagepoint} = []; \]
\[ x0 = []; \]
\[ y0 = []; \]
\[ z = []; \]
\[ x0 = E11(1); \]
\[ y0 = E11(2); \]
\[ z = px(1)*x^4 + py(1)*y^4 + px(2)*x^3 + py(2)*y^3 + px(3)*x^2 + py(3)*y^2 + px(4)*x + py(4)*y; \]
\[ \text{averagepoint} = [x_0, y_0, z] \]
\[ c = 2; \]
\[ \text{for } k1 = 1:d; \]
\[ \text{if newpoints}(k1,4) == 0 \text{ find the points that are in the polygon} \]
\[ \text{averagepoint}(c,:) = [\text{newpoints}(k1,1), \text{newpoints}(k1,2), \text{newpoints}(k1,3)] \]
\[ c = c+1; \]
\[ \text{end} \]
\[ \text{end} \]
\[ a = \text{averagepoint} \]
\[ \text{else Dclus == 1} \]
\[ a = a; \]
\[ [da, la] = \text{size}(a); \]
\[ \text{end} \]
\[ [da, la] = \text{size}(a); \]
\[ a = a; \]
\[ \text{end} \]
\[ c = a \]
Appendix B
Results: surface 1

Figure 1: Mesh structure generated from 10 unit sphere

Figure 2: Histogram of mesh 1 surface 1

Target nodal spacing = 10
Max nodal spacing = 10
Min nodal spacing = 10
Mean nodal spacing = 10
Standard deviation = 0
Uniformity = True
Figure 3: Mesh structure generated from 5unit sphere

Figure 4: Histogram of mesh 2 surface 1

- Target nodal spacing = 5
- Max nodal spacing = 5
- Min nodal spacing = 5
- Mean nodal spacing = 5
- Standard deviation = 0
- Uniformity = True
Figure 5: Mesh structure generated from 4 unit sphere

Target nodal spacing = 4
Max nodal spacing = 4
Min nodal spacing = 4
Mean nodal spacing = 4
Standard deviation = 0
Uniformity = True

Figure 6: Histogram of mesh 3 surface 1
Figure 7: Mesh structure generated from 3 unit sphere

Figure 8: Histogram of mesh 4 surface 1

Target nodal spacing = 3
Max nodal spacing = 3
Min nodal spacing = 3
Mean nodal spacing = 3
Standard deviation = 0
Uniformity = True
Figure 9: Mesh structure generated from 2 unit sphere

Target nodal spacing = 2
Max nodal spacing = 2
Min nodal spacing = 2
Mean nodal spacing = 2
Standard deviation = 0
Uniformity = True

Figure 10: Histogram of mesh 5 surface 1
Results: surface 2

Figure 11: Mesh structure generated from 10 unit sphere

Target nodal spacing = 10
Max nodal spacing = 10.071
Min nodal spacing = 9.9053
Mean nodal spacing = 9.9971
Standard deviation = 0.0187
Uniformity = True

Figure 12: Histogram of mesh 1 surface 2
Figure 13: Mesh structure generated from 5unit sphere

Figure 14: Histogram of mesh 2 surface 2

Target nodal spacing = 5
Max nodal spacing = 5.0119
Min nodal spacing = 4.9644
Mean nodal spacing = 4.9981
Standard deviation = 0.0051
Uniformity = True
Figure 15: Mesh structure generated from 4 unit sphere

Figure 16: Histogram of mesh 3 surface 2

Target nodal spacing = 4
Max nodal spacing = 4.009
Min nodal spacing = 3.9701
Mean nodal spacing = 3.9988
Standard deviation = 0.0034
Uniformity = True
Figure 17: Mesh structure generated from 3 unit sphere

![3D mesh structure](image)

Target nodal spacing = 3
Max nodal spacing = 3.0052
Min nodal spacing = 2.9852
Mean nodal spacing = 2.9992
Standard deviation = 0.0019
Uniformity = True

Figure 18: Histogram of mesh 4 surface 2

![Histogram of nodal spacing](image)
Figure 19: Mesh structure generated from 2 unit sphere

Target nodal spacing = 2
Max nodal spacing = 2.0049
Min nodal spacing = 1.9827
Mean nodal spacing = 1.9995
Standard deviation = 0.0015
Uniformity = True

Figure 20: Histogram of mesh 5 surface 2
Results: Surface 3

Figure 21: Mesh structure generated from 10 unit sphere

Figure 22: Histogram of mesh 1 surface 3

Target nodal spacing = 10
Max nodal spacing = 10.371
Min nodal spacing = 9.4636
Mean nodal spacing = 9.935
Standard deviation = 0.1912
Uniformity = True
Figure 23: Mesh structure generated from 5 unit sphere

Figure 24: Histogram of mesh 2 surface 3

- Target nodal spacing = 5
- Max nodal spacing = 5.1887
- Min nodal spacing = 4.7377
- Mean nodal spacing = 4.97
- Standard deviation = 0.0872
- Uniformity = True
Figure 25: Mesh structure generated from 4 unit sphere

Target nodal spacing = 4
Max nodal spacing = 4.1405
Min nodal spacing = 3.8085
Mean nodal spacing = 3.9767
Standard deviation = 0.0670
Uniformity = True

Figure 26: Histogram of mesh 3 surface 3
Figure 27: Mesh structure generated from 4 unit sphere

Figure 28: Histogram of mesh 4 surface 3

Target nodal spacing = 3
Max nodal spacing = 3.1178
Min nodal spacing = 2.8502
Mean nodal spacing = 2.9808
Standard deviation = 0.0553
Uniformity = True
Figure 29: Mesh structure generated from 2 unit sphere

Figure 30: Histogram of mesh 5 surface 3

- Target nodal spacing = 2
- Max nodal spacing = 2.0929
- Min nodal spacing = 1.8759
- Mean nodal spacing = 1.9854
- Standard deviation = 0.0404
- Uniformity = True
Results: surface 4

Figure 31: Mesh structure generated from 10 unit sphere

Target nodal spacing = 10
Max nodal spacing = 10.604
Min nodal spacing = 9.3329
Mean nodal spacing = 10.068
Standard deviation = 0.2040
Uniformity = True

Figure 32: Histogram of mesh 1 surface 4
Figure 33: Mesh structure generated from 5 unit sphere

Figure 34: Histogram of mesh 2 surface 4

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target nodal spacing</td>
<td>5</td>
</tr>
<tr>
<td>Max nodal spacing</td>
<td>5.3245</td>
</tr>
<tr>
<td>Min nodal spacing</td>
<td>4.7953</td>
</tr>
<tr>
<td>Mean nodal spacing</td>
<td>5.0296</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>0.0853</td>
</tr>
<tr>
<td>Uniformity</td>
<td>True</td>
</tr>
</tbody>
</table>
Figure 35: Mesh structure generated from 4 unit sphere

Figure 36: Histogram of mesh 3 surface 4

- Target nodal spacing = 4
- Max nodal spacing = 4.2758
- Min nodal spacing = 3.8287
- Mean nodal spacing = 4.0241
- Standard deviation = 0.0696
- Uniformity = True
Figure 37: Mesh structure generated from 3 unit sphere

Figure 38: Histogram of mesh 4 surface 4

Target nodal spacing = 3
Max nodal spacing = 3.18
Min nodal spacing = 2.8369
Mean nodal spacing = 3.0184
Standard deviation = 0.0572
Uniformity = True
Figure 39: Mesh structure generated from 2 unit sphere

Target nodal spacing = 2
Max nodal spacing = 2.1627
Min nodal spacing = 1.8600
Mean nodal spacing = 2.0138
Standard deviation = 0.0413
Uniformity = True

Figure 40: Histogram of mesh 5 surface 4