University of Twente

Faculty of Engineering Technology
Department of Mechanical Engineering
Chair of Production technology

CONFIDENTIAL

A Rapid Prototyping system for the Hot-Wire Cutting process

Master Thesis

J.H. VAN RAVENHORST

January 2008
CTW PT 0852

Enschede, January 15, 2008

A Rapid Prototyping system for the Hot-Wire Cutting process

Master's Thesis - Mechanical Engineering

by

J.H. VAN RAVENHORST
Student number: s0010979

Submitted in partial fulfillment of the requirements for the degree of Master of Science

Examination committee:

Prof. Dr. Ir. R. Akkerman	Supervisor, University of Twente
Ing. C. van Driel	Supervisor, 3EL-Company bv
Ir. G. van Ouwerkerk	University of Twente
Ing. A. de Smit	Delft University of Technology
Ir. H. Tragter	University of Twente

Institute of education:
Chair of Production Technology
Department of Mechanical Engineering
Faculty of Engineering Technology
University of Twente
Enschede, The Netherlands

Master Project location:
3EL-Company bv
Enschede, The Netherlands

CTW PT 0852

Abstract

This Master Project was performed at 3EL-Company bv, a company specialized in cutting EPS using a CNC hot-wire cutting (HWC) machine. The problem is the lack of preprocessing automation compared to the competing milling process. The goal for this Master Project is to develop an automated method which slices a CAD model into layers and generates the tool paths. First, a literature study is performed. Next, the requirements are assessed. Subsequently, suitable algorithms are chosen and implemented. Finally, a layered object is manufactured and measured. The layer thickness is iteratively optimized with respect to constraints involving accuracy, the fabrication process and material surface quality. Therefore the thickness of the layer depends on local model surface properties such as the curvature and orientation. In order to preserve features, direct slicing of the CAD geometry is used instead of a tessellated approximation. It is concluded that the automation yields satisfactory results for non-branching input models with relatively simple topology, although the calculation time should be reduced. Furthermore, in general, a curvature-based layer thickness estimation is worth the effort in the first iteration only. Also, intersection through edges and vertices should be avoided and the resulting curves must be checked for validity before use. The manufacturing accuracy of a layered object has a measured RMS accuracy in the order of 1 mm in the layer plane and 0.1 mm per adhesive interface. The measured tolerance is in the order of 6 mm.

Samenvatting

Dit afstudeerproject is uitgevoerd bij 3EL-Company bv, een bedrijf dat is gespecialiseerd in het snijden van EPS met een CNC hete draad snijmachine. Het probleem is het gebrek aan automatisering van werkvoorbereiding vergeleken met het concurrerende freesproces. De doelstelling van dit afstudeerproject is het ontwikkelen van een geautomatiseerde methode die een CAD model in lagen opdeelt en de snijbanen genereert. Allereerst wordt een literatuurstudie gedaan. Vervolgens wordt het pakket van eisen vastgesteld. Daarna worden geschikte algorithmen gekozen en geïmplementeerd. Uiteindelijk wordt een gelaagd object gefabriceerd en opgemeten. De laagdikte wordt iteratief geoptimaliseerd met betrekking tot randvoorwaarden betreffende de nauwkeurigheid, het fabricageproces en de oppervlaktekwaliteit. De dikte van een laag hangt daarmee af van lokale eigenschappen van model oppervlakken zoals de kromming en de oriëntatie. Teneinde vormkenmerken te behouden, wordt het CAD model *direct* in lagen opgedeeld, zonder de tussenkomst van meshing. Er kan worden geconcludeerd dat de automatisering bevredigende resultaten oplevert voor niet-vertakkende input modellen met relatief eenvoudige topologie, hoewel de rekentijd verlaagd zou moeten worden. Verder kan gesteld worden dat in het algemeen een op kromming-gebaseerde laagdikte schatting slechts in de eerste iteratie de moeite waard is. Ook moet intersectie door model-randen en -punten worden voorkomen en er moet worden gecheckt of de resulterende curves geldig zijn voor verder gebruik. Het fabricageproces van een gelaagd object heeft een RMS nauwkeurigheid in de ordegrootte van 1 mm in het vlak van de lagen en 0.1 mm per lijmlaag. De gemeten tolerantie ligt in de ordegrootte van 6 mm.

Acknowledgements

First of all, I would like show my gratitude to my parents for their unconditional support of their eldest son who embarked at this rather technical study at the University of Twente, bearing the name of 'Mechanical Engineering'.

The master project activities were located at 3EL-Company bv in Enschede, The Netherlands. I would like to thank my supervisor Cock van Driel from 3EL-Company for his advice and shared knowledge about both hot-wire cutting and its applications. I also want to thank Annelies van Driel who compensated for the technical and theoretical neuroses of the men around her in the company.

From the University of Twente, also located in Enschede, I would like to thank my supervisor Professor Remko Akkerman who, in spite of his enormous amount of activities, was always willing to find time and share ideas. Appreciation is also there for my fellow-students and staff at the chair of Production Technology who attended the meetings during the master project, leading to helpful discussions about various subjects.

Quite some time was spent to make crucial decisions and it was Hans Tragter who helped me convince in choosing direct slicing over tessellated slicing. This was not until later on in the process, where its advantages became more clear, that I learned to appreciated it. I also would like to thank his students for their time and thought of several slicing-related problems.

The conceptual design of the Skymobil body as used throughout this work was made available by Götz Steudel, a retired engineer from Germany. It is part of the Skymobil project involving the development of a flying car, primarily involving an EPS-carbon sandwich construction. His inspiration was most welcome.

I would like to thank Erik Schoonbergen for sharing his well-documented experience in the field of photogrammetry as well as Robert Kroon from the company Geodelta for his advice about photogrammetry in general. Finally, I would like to thank Jilles Eindhoven for providing the photograph for figure 2.1.

List of Symbols and Abbreviations

The following lists include the most important nomenclature used throughout this work. The page numbers denote the first location of appearance. Unlisted nomenclature receives its descriptions from the context and can have multiple meanings, depending on the location of appearance.

Symbols

ÆR	arc length ratio, page 41	
ÆR*	minimum arc length ratio for a single layer, page 42	
ÆR$_{min}$	minimum allowed arc length ratio, page 42	
δ	cusp height or maximum normal deviation, page 27	m
ϵ	error in layer plane, page 32	m
$\hat{I}$	'unit change'-based thickness list index, page 43	
κ	curvature of a curve, page 30	m^{-1}
κ_n	normal curvature of a surface, page 30	m^{-1}
$\kappa_{n,1}$	maximum principal curvature, page 31	m^{-1}
$\kappa_{n,2}$	minimum principal curvature, page 31	m^{-1}
$\mathbf{B}$	binormal vector, page 77	
$\mathbf{D}$	second fundamental matrix, page 80	
$\mathbf{G}$	first fundamental matrix, page 79	
ρ	radius of curvature, page 29	m
σ	standard deviation, page 110	
θ	wire angle, page 39	
θ^*	maximum wire angle for a single layer, page 39	
θ_{max}	maximum allowed wire angle, page 39	

$\tilde{I}$	κ_n-based thickness list index, page 43	
$\vec{c}$	unit correspondence vector, page 40	
$\vec{d}$	unit direction vector which, together with $\vec{n}_S$, defines plane P, page 29	
$\vec{e}_1$	maximum principal curvature direction, page 31	
$\vec{e}_2$	minimum principal curvature direction, page 31	
$\vec{e}_z$	build direction, page 13	
$\vec{n}$	nominal unit surface normal vector, page 28	
$_b$ (subscript)	denotes the base of a layer, page 20	
$_t$ (subscript)	denotes the top of a layer, page 20	
C	parametric curve, page 20	
e	actual normal deviation, page 30	m
e^*	maximum absolute actual normal deviation for a single layer, page 40	m
I	final thickness list index, page 43	
R	ruled parametric surface, page 20	
S	nominal parametric surface, page 27	
s	arc length, page 77	m
T	layer thickness, page 43	m
t	layer thickness estimation, page 29	m
t^*	critical (smallest) layer thickness estimation, page 30	m
u	parameter of a parametric curve or first parameter of a parametric surface, page 79	
v	second parameter of a parametric surface, page 79	

Abbreviations

API	application programming interface, page 46
CAD	computer-aided design, page 12
CAM	computer-aided manufacturing, page 14
CNC	computer numerical controlled, page 10

EDM	electrical discharge machining, page 17
FEM	finite element method, page 19
HWC	hot-wire cutting, page 10
KWC	kerf width correction, page 26
LM	layered manufacturing, page 12
LMT	layered manufacturing technologies, page 12
NURBS	non-uniform rational B-spline, page 19
NVP	normal vertical plane - a plane spanned by $\vec{n}_S$ and $\vec{e}_z$, page 29
PG	photogrammetry, page 108
RHR	right hand rule, page 49
RMS	root mean square, page 110
RP	rapid prototyping, page 12
SE	Solid Edge, page 46
SFF	solid freeform fabrication, page 12
STL	stereo lithography (file format), page 18
SW	Solidworks, page 46
TLOM	thick layered object manufacturing, page 14

Contents

Chapter 1

Introduction

In this chapter, hot-wire cutting and the rapid prototyping process are introduced. Both techniques are tightly connected. Rapid prototyping techniques are essential for advanced use of hot-wire cutting, as will become clear in this chapter. Definitions used in the rapid prototyping area are presented as far as applicable to this work. Next, the purpose of the overall programme is described as well as the problem being investigated in the master project. Finally, the outline for the rest of this work is given.

1.1 Introduction to hot-wire cutting

The 'hot-wire cutting' (HWC) process is used to cut foams made of polystyrene or other thermoplastic materials. HWC was partly developed in the area of model airplane building. Currently, it is used for various applications. Examples are the production of models and molds for the wind turbine blade industry and ship hull production. Other applications are the production of display signage, free form architecture, skatepark shaping and many others. It is also used as a preprocessing step for milling.

HWC makes use of a current that is fed through a wire which heats up as a consequence, reaching a temperature that is sufficiently high to vaporize and melt the foam. Ideally, the foam is vaporized just ahead of the advancing wire instead of being touched by it. The most common application is using the wire in a straight line while subjected to an actively controlled tension, allowing it to make so called 'ruled surfaces'. See also section 2.5.1 for its definition. The principle of HWC is shown in figure 1.1 with a four-axis 'computer numerical controlled' (CNC) HWC machine.

The wire is held between two carriages which are independently driven in both horizontal and vertical direction using four spindles, each driven by stepper motors. Assume distances A, T and P known, as well as the product contours on planes a and b. Assume that a relation, indicating which points on the two contours correspond with each other, is known. The machine then has enough information to determine the carriage trajectories in both portal planes and start cutting by controlling the stepper engines motors. When the two drawings are equal copies with no scale, shift or deformation differences and each contour point on plane a corresponds with its copy on plane b, the resulting shape is often called 2D. In

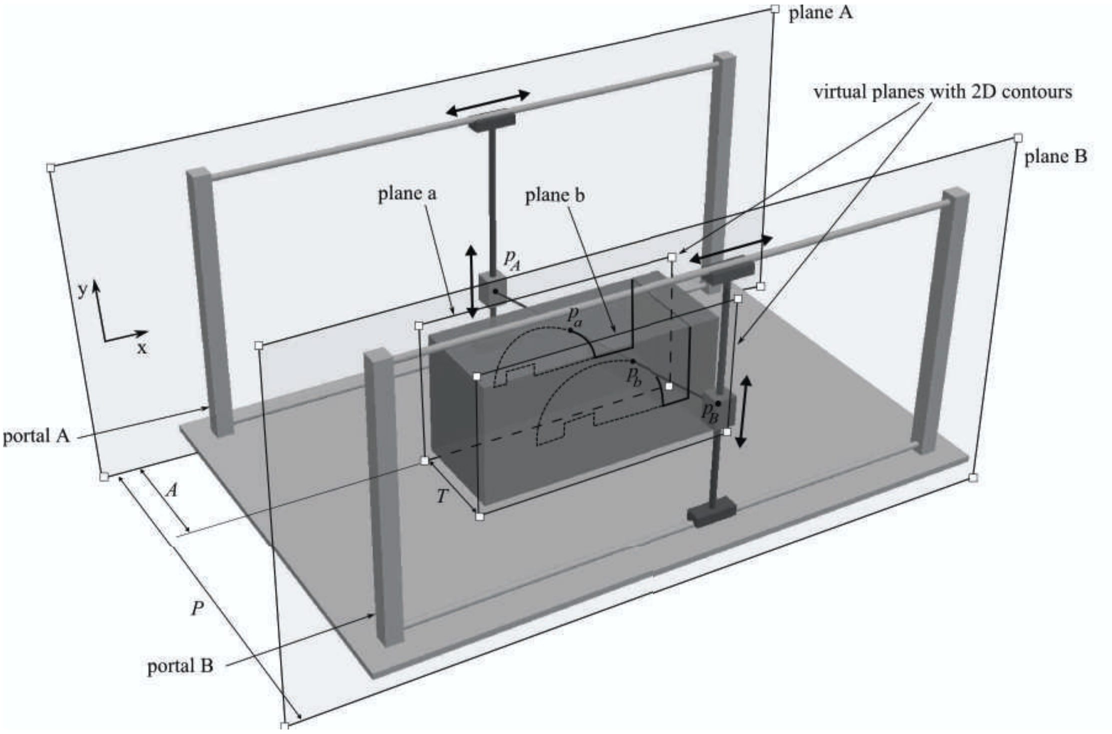


Figure 1.1: Schematic representation of a four-axis CNC HWC Machine

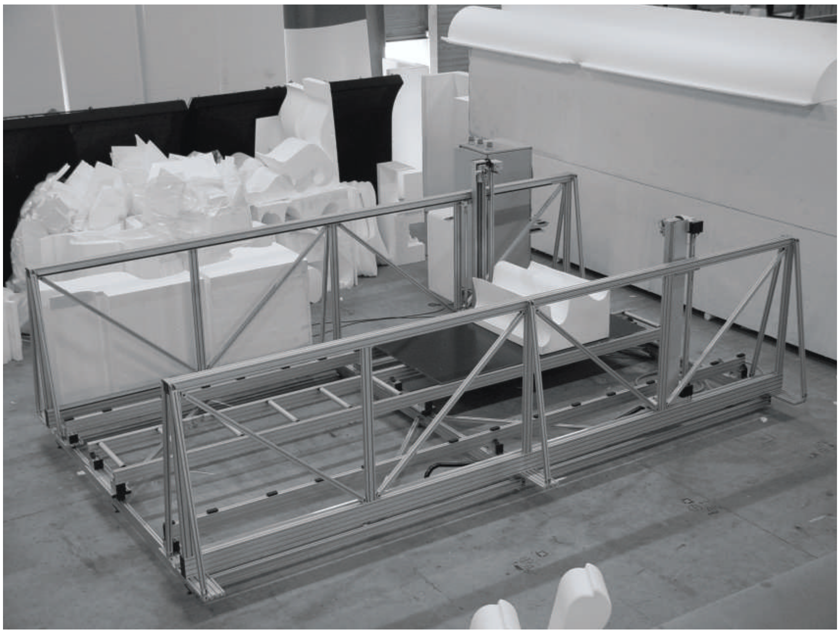


Figure 1.2: Step Four®HWC machine from at 3EL-Company

that case, the carriages at portals A and B describe the same trajectory in time. Otherwise, the shape is called 2.5D. In all cases, the resulting part surface is a ruled surface.

Advantages of HWC are, amongst others, its intrinsic simplicity and low energy use. Also, the speed for production of large shapes can be high compared to other material removing processes. It is in some cases considered a potential competitor in the field of three- to five-axis CNC milling, especially for products larger than the order of one meter and without much detail. Reasons are the lower cost due to reduced production time and reduced material usage. Disadvantages are some limitations in freedom of shape and level of detail. A photograph of a HWC machine is depicted in figure 1.2.

1.2 Introduction to rapid prototyping

'Rapid prototyping' (RP) is a terminology used to refer to techniques for creating parts directly from 'computer-aided design' (CAD) models within a relatively short time. It is also referred to as 'solid freeform fabrication' (SFF). RP can be incremental or decremental [5]. Incremental RP uses material addition primarily and decremental RP starts with a raw block of material and shapes the final part by material removal such as milling. The incremental variant will be of main interest for this thesis. In the following, RP is used to denote the incremental form. The most well-known example of RP is stereo lithography, where parts are 'printed out' in three dimensions. In many cases, thin layers are created and subsequently or simultaneously fused together. The family of techniques using this principle is also called 'layered manufacturing technologies' (LMT). Materials used vary widely, but most often consist of (foamed) plastics or paper, although materials such as metals are used as well. The RP process is often characterized as an optimization between part accuracy and building time or cost. A significant production time reduction for prototypes as well as final products can be achieved, leading to an important role for RP in the design process [29].

Incremental RP processes can be generally subdivided in the steps shown in figure 1.3. In the following paragraphs, each step is explored. The second step is treated in more detail due to the relevance for the rest of this work as well as the fact that most systems make use of the same principles mentioned in that step.

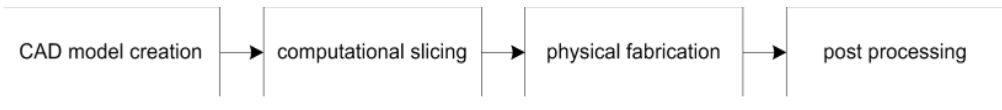


Figure 1.3: Incremental RP process chain

The first step in the RP chain is to obtain a CAD model of the intended product. This can either be a solid or a surface model. The model can also be approximated by a set of triangles that cover the surface, which is called 'tessellation' or 'meshing'. An example is shown in figure 1.4.

The next step is computational slicing, which requires the CAD model as input. The original CAD model can be computationally subdivided or 'sliced' into layers

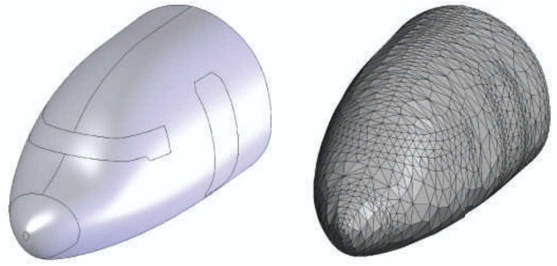


Figure 1.4: (left) Exact CAD model of an aircraft body and (right) the tessellated version, using triangular facets

without presence of tessellation, which is then called ‘direct slicing’. When a tessellated approximation is used, the process is denoted as ‘tessellated slicing’. Whether or not direct slicing is used, the part is subdivided in a number of layers with finite thickness, bounded by two section planes. Use of a constant layer thickness is called ‘uniform slicing’. Varying the layer thickness as a function of local curvature, topology and other parameters leads to the term ‘adaptive slicing’, which can save the amount of layers considerably and hence cut cost as well. The ‘vertical’ or ‘build direction’, denoted by $\vec{e}_z$, is defined perpendicularly to the section planes. The most basic techniques produce 2D-shaped layers, resulting in a so-called ‘staircase effect’ Take, for example, the two leftmost sliced shapes in figure 1.5. In literature, this has been defined as ‘zero-order slicing’. In general,

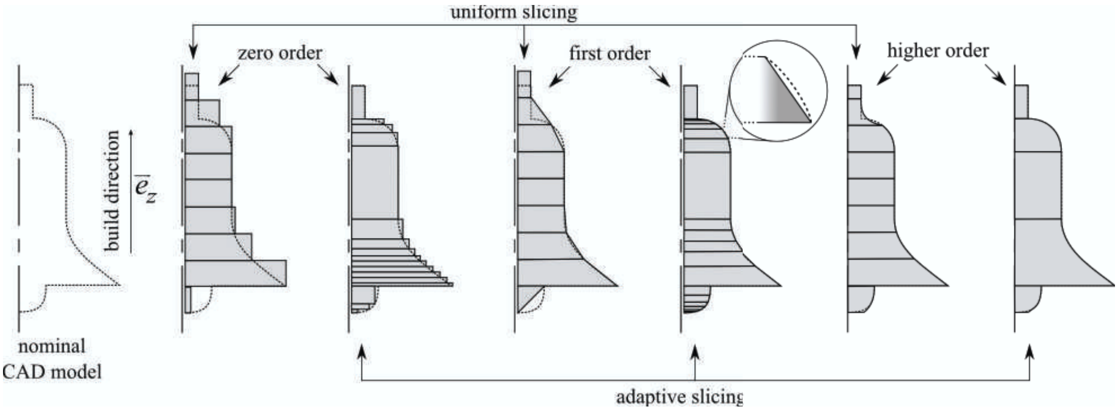


Figure 1.5: Overview of different slicing method combinations for an axisymmetric bell shape, depicted as one half of the cross-sectional shape for simplicity. Note that both efficiency and RP complexity increase from left to right.

zero-order slicing part surfaces have a higher roughness than the original model and hence additional post processing is required in many cases. Using the same building time period, a far more accurate result can be obtained by the use of ‘first order slicing’, where the side surfaces of the layers consist of arbitrary ruled -and hence single curved- surfaces. The amount of layers can be reduced drastically in this way, given a predefined tolerance. However, more expensive four- or five-axis machines are required to achieve this. See also figures 1.1 and 1.6 for examples.

Even more sophisticated is the use of 'higher order' slicing, yielding double curved layer side surfaces, but use of this technique is hardly encountered in literature and considered rare, as can be inferred from RP field overviews as in [29].

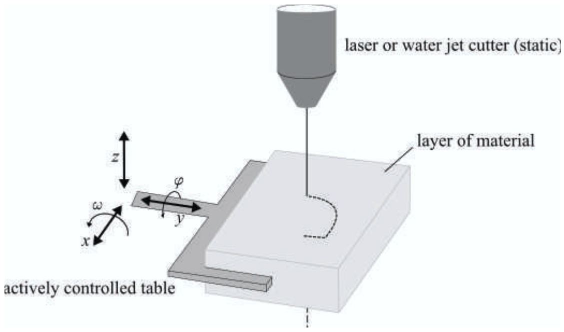


Figure 1.6: Schematic representation of a five-axis machine used for RP

RP technologies can also be classified by the layer thickness used. 'Thin-layered' is the de facto term used when referring to layers with a thickness (<= 1 mm) and thick-layered when a layer thickness of (> 1 mm) is used.

The third step is transferring the sliced geometry to a machine and produce the layers from a specific process-related material. 'Computer-aided manufacturing' (CAM) is often tightly integrated with CAD in the RP process.

Fusing the layers is sometimes not necessary because the layers fuse during the layer creation process. However, some systems require bonding. This often requires additional automation and other means of help such as a guide pin and hole and connector systems, as shown in [1].

'Thick layered object manufacturing' (TLOM) is the term used to denote the fabrication of objects in the order of one cubic meter and larger, whereas traditional RP techniques are usually limited to a cross-section smaller than one square meter. HWC is suited for TLOM and also used for this purpose in practice. This work aims at combining HWC and TLOM as will be explained in the following sections.

1.3 Problem

The project as described in the next section initiated at 3EL-Company bv, a young company that is specialized in cutting of Expanded Polystyrene (EPS) using a four-axis CNC HWC machine. Although the technique is considered a competitor to milling, the software for milling is far more mature and integrated with CAD. Finding the reason for this is beyond the scope of this work, but could be strongly related to smaller demand for HWC. At present, a lot of time is spent on preprocessing CAD models for cutting, using the tools provided by commercial CAD systems. CAD models are sliced by manually defining cross-sections at suitable locations, based on common sense and educated guessing. Sometimes this is accompanied by manual subdivision of more complex models into sliceable segments beforehand. This time-consuming preprocessing is one of a few significant disadvantages of HWC. Time can be saved by automation which has the ability to offer control of dimensional accuracy. Obviously, the surfaces generated with HWC deviate from double curved surfaces, requiring a good approximation method. Manual slicing often does not guarantee a part to be within the required tolerances. Most milling software inherently contain this kind of control by default. Finally, it is important to realize that no matter how much automation is applied, the preprocessing stage still contains many aspects that require creative human thought which cannot be automated in a cost-effective sense due to the vast variety in nature of the customer projects involved. The issues mentioned above have lead to the project as described next, performed in cooperation with the University of Twente at the Department of Mechanical Engineering, Chair of Production Technology.

1.4 Scope of the overall programme

The foregoing lead to the start of the Protostyrene project which focuses at the issues mentioned in the previous section. This project encapsulates the Master project documented in this thesis. The top-level purpose of the project is to create an automated software tool that assists the pre-process engineer in defining an optimal slicing of a CAD model, saving time in the preprocessing phase considerably. Possibly, integration with CAM will be a long-term objective of the project as well. As preprocessing time generally constitutes the largest piece of the total product lead time and effort, this approach is considered justified. This objective is defined in a general way. The Master Project as described in the following section intends to take the first significant steps and sets a more specific objective.

1.5 Master Project objective

The Master Project objective is: To formulate and implement an automated method which subdivides an arbitrary two-manifold CAD model into parallel layers in such a way that it maximally benefits from the four-axis hot-wire cutting process with a predefined accuracy, followed by the determination of the resulting accuracy with a physical demonstrator.

The master project is the first sub-project started in the overall programme and is therefore also used to perform a literature study about existing RP techniques. Using this knowledge, feasible requirements have been defined. Inspired by the simplicity of the HWC process and the possibilities in the RP field as denoted in the previous sections, this work describes the investigation into the suitability of both existing and new RP methods as a tool for HWC preprocessing. Based on related work and practical constraints, a basic prototype program is developed and implemented. Because of the rather extended nature of the project, this thesis focuses on a few important aspects of the entire slicing process. A set of project requirements, simplifications and constraints have been determined in order to make the problem more manageable. The problem under investigation restricts itself to finding and implementing a suitable set of algorithms for adequately slicing a CAD model in such a way that the intent of the designer is retained as well as satisfying the demand that the slicing is performed optimal with respect to a set of rules and boundary conditions. The main focus involves finding a solution to the correspondence problem as well as finding a suitable error-approximation, combined with the creation of a physical demonstrator, which is used to get an indication of practical obtainable accuracy.

1.6 Thesis outline

The contents of this paper are organized as follows. In chapter 2, an overview of existing work is presented and commonly used methods are explained. Chapter 3 discusses the requirements and methods used for implementation of the tool. Next, chapter 4 presents how the methods chosen are implemented in a software tool, connected to a third-party CAD program. The general structure of the program and the most important algorithms are described there. Chapter 5 presents the results using a demonstrator CAD model that is sliced by the program. The model is then physically cut, measured and compared to the sliced CAD geometry. Chapter 6 discusses to what degree the objectives are met, discussing limitations and obtainable accuracy. Finally, chapter 7 summarizes the master project and indicates the most valuable points of interest for further research.

Chapter 2

Background

In this chapter, the research groups and techniques currently employed are described, as far as applicable to the current project. The mathematical background of various subjects is treated in order to comprehend concepts and enjoy confidence in them.

2.1 Related projects

From literature, the advantages of first order adaptive slicing over zero order are well-known [14, 29]. For TLOM, only first- or higher order slicing are used and just a few groups have dedicated themselves to the development of systems capable of doing it. References or citations regarding more specific issues can be found throughout the rest of this work. The most well-known projects are presented in table 2.1. The projects stem from the past two decades, except the last one. Apart from these examples, a few other undocumented or otherwise non-publicly available projects may exist, as well as a range of non-automatized variants.

name	**ref.**	**country**	**machine**	**method**	**order**
CAM-LEM	[40]	USA	5-axis laser	tessell.	first
FF-TLOM	[5]	The Netherlands	hot-knife cutting	direct	higher
Shapemaker II	[35]	USA	5-axis water jet	tessell.	first
Stratoconception	[4]	France	6-axis laser	tessell.	first
TruSurf	[13]	Australia	5-axis water jet	direct	first
VLM-ST	[1]	South Korea	table HWC	tessell.	first
Wirepath	[4]	USA	EDM[1]	direct	first
-	[6]	The Netherlands[2]	hot-knife cutting	-	higher

- unknown
[1] electrical discharge machining
[2] see figure 2.1

Table 2.1: Overview of TLOM projects

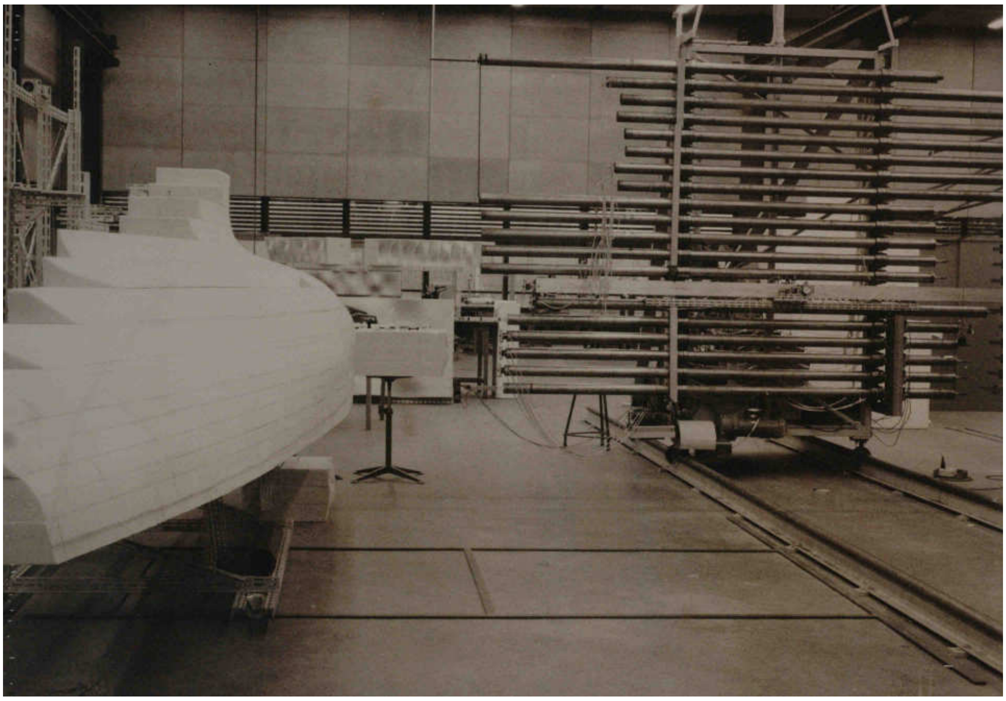


Figure 2.1: Foam cutting machine at the University of Twente in the late seventies

2.2 CAD model import

Tessellated slicing has become very popular in RP. The file format used most often for import of a CAD model to a tessellated slicing system is the 'stereo lithography' (STL) file format. A STL file contains nothing more than a set of triangles without topological information related to the original CAD model. Various arguments in favor of and against STL exist. The most important are listed in table 2.2.

Advantages	**Disadvantages**
de facto standard in RP	topology loss
simple algorithms for slicing required	chordal error
basic and neutral file format	limited functionality for complex parts
	large data set size

Table 2.2: Arguments in favor of and against STL

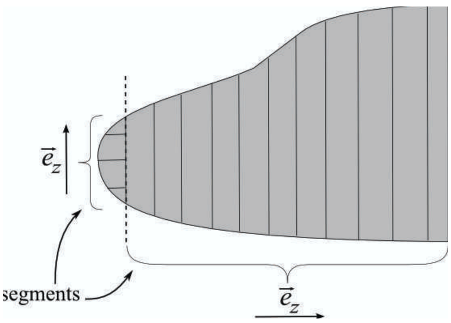
With the use of STL, the absence of topology roughly means that the connectivity relations between the CAD model entities are lost. This makes validity checking computationally expensive, as inter-triangular topological relationships must be reconstructed in order to perform efficient validity or integrity checking [31, 39]. Validity checking is, unfortunately, necessary in the first place because many commercial CAD tessellators are not robust [31]. The topology reconstruction is necessary anyway for drastically speeding up computationally slicing algorithms. Another drawback of the tessellated representation is the introduction of a surface approximation error, also denoted as 'chordal error'. This is the deviation between the tessellation surface and the original CAD surface. Usually, the maximum chordal error can be imposed by the user when tessellating the CAD model. Setting the chordal error one order smaller than the maximum allowed slicing error will eliminate significant contribution of tessellation errors, yet may increase the amount of triangles drastically. Due to the limitations of STL, alternative and more extended formats have been proposed [30, 38, 16], but have not yet lead to new widely accepted standards.

For research purposes, 'finite element method' (FEM) preprocessors can be used instead of STL. FEM tessellators are generally more robust, so the bulk of error checking is not necessary anymore. Besides, inter-triangular topology is preserved, the file size is reduced and import is faster.

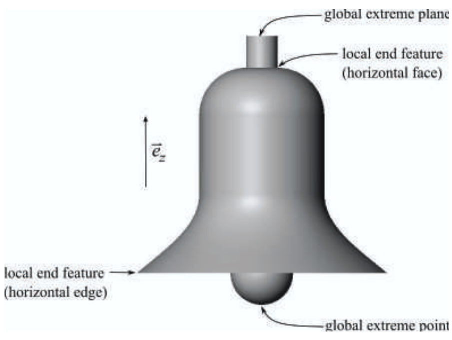
On the other hand, 'exact' CAD file formats like ACIS, IGES and STEP provide a representation for solid and surface geometry as well using the 'non-uniform rational B-spline' (NURBS). When using the non-tessellated CAD geometry, this representation avoids the disadvantages of STL [13].

2.3 Decomposition, segmentation and build direction

Before computational slicing can begin, decomposition and segmentation might be needed. Decomposition or 'splitting up' of a CAD model (assembly) into parts is usually very hard to automate [6]. The intended result depends on the purpose of the resulting physical model, which could, for example, imply that certain part of an assembly need not to be decomposed at all. The structural strength of the individual components is also an aspect when decomposing, just like accuracy, economical aspects and specific process-related constraints.



(a) Two segments with different build directions



(b) End features of a nominal CAD model

Figure 2.2: Segmentation and end features

After decomposition, segmentation can be applied to divide models into sections with optional individual build direction as shown in figure 2.2(a). Subdivision of parts into segments suffers from some of the same issues as in decomposition, although certain methods exist to facilitate this step. Curvature mapping and checking algorithms for convex and concave features, singularities and end feature detection, as in figure 2.2(b), can assist the segmentation process [33, 26, 23, 19, 4]. These subjects will not be discussed in further detail. Here, the build direction is assumed constant for each single segment. This could, of course, be generalized to a varying build direction throughout a segment, giving more freedom, but this would increase the slicing process complexity both theoretically and practically.

2.4 Slicing

Although every type of (non)parametric surface has its own mathematical description, most CAD package geometry kernels are able to offer common interfaces for arbitrary surface types, along with integrated intersection algorithms, greatly facilitating the slicing procedures. The need to program, for example, intersection functions for each type of surface is therefore not necessary.

Computational slicing exists in many varieties. In the current project, only adaptive first-order thick-layered slicing is considered due to the corresponding properties of the HWC process. The slicing step is of particular interest and therefore treated in more detail. The most important part of slicing is the problem of surface reconstruction by ruled surfaces from a set of intersection contours and optionally the original surface. The next section treats this problem, followed by sections describing tool path generation, layer thickness estimation and actual error estimation.

2.5 Correspondence problem

This section will presents a mathematical description of the correspondence problem. Next, implementations of solving the correspondence problem are given.

2.5.1 Ruled surfaces and correspondence

As mentioned before, ruled surfaces play a role in both HWC and RP. Mathematically [9], a ruled surface R can be expressed by straight lines, 'rulings' or 'generators' joining corresponding points on two space curves $\vec{r} = \vec{r}_0(u)$ and $\vec{r} = \vec{r}_1(u)$ in $\mathbf{R}^3$, also known as directrices. The expression can be given in the following form:

$$\vec{r} = \vec{r}(u, v) = (1 - v)\vec{r}_0(u) + v\vec{r}_1(u) \tag{2.1}$$

A ruled surface is depicted in figure 2.3. When the rulings are initially unknown, it is convenient to express the contours as regular parametric curves:

$$C_b(u_b) \text{ and } C_t(u_t),$$

Where subscript b stands for base and t for top, as used throughout the rest of this work. Next, allow u_b and u_t to be changed by $U_b : x \mapsto u_b$ and $U_t : x \mapsto u_t$ with the constraint that $U_b'(x) > 0$ and $U_t'(x) > 0$, yielding regular parameterizations. The 'regular' property of the parameterization ensures that no unnatural self-intersections occur. Now, C_t can be reparameterized by $u_t(u_b) = U_t(U_b^{-1}(u_b))$ and hence the contours can be expressed as

$$C_b(u_b) \text{ and } C_t\left(u_t(u_b)\right). \tag{2.2}$$

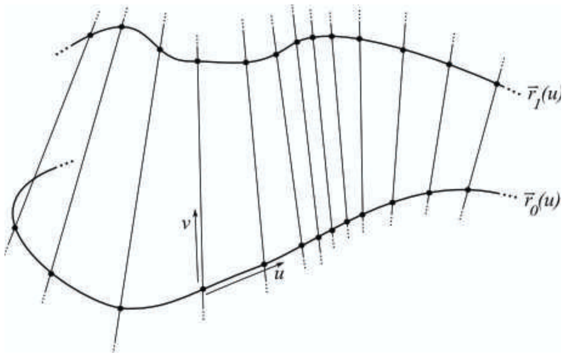


Figure 2.3: Ruled surface

The rulings have now become a result of this functional relationship $u_t(u_b)$. Using the arc length as parameter, finding an acceptable or optimal solution for this relationship simply involves solving the 'correspondence problem'. Stated otherwise, it deals with the topological adjacency relationships between the contours. That is, determining which point on the upper section contour corresponds to another point at the lower contour. Next, assume each curve to lie on a plane and assume that these planes are parallel. Then, the mechanical equivalents of these mathematical concepts can be identified as in table 2.3.

Mathematical		**Mechanical**
top directrix	=	top contour
base directrix	=	base contour
ruling	=	(approximate) wire location at specific time
functional relationship	=	correspondence or (approximate) tool path

Table 2.3: Mechanical equivalents or mathematical concepts

Solving the correspondence problem is severely under constrained and therefore challenging. Stated more loosely, an infinite number of (ruled) surfaces can form the surface connecting the contours. As an example, different correspondence solutions for two circular intersection curves are shown in 2.4. The problem also exists in many other fields such as CT-scan imaging. In those fields, only the contours are available. Most literature focuses at reconstructing a tessellated surface from a set of contours only. However, for RP, the original surface is available as well by a (tessellated) CAD model, giving additional information for correspondence solving.

The correspondence problem can also involve branching as in figure 2.5. Branching occurs in cases where multiple top and/or base contours exist. Branching is well-documented in literature [11, 12, 25, 32] and it deserves a field of its own. Branching is excluded from this project and is therefore not treated in detail.

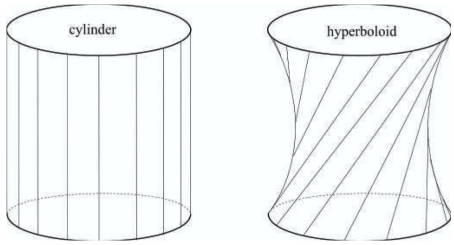


Figure 2.4: Two correspondence solutions yielding a cylinder (left) and a hyperboloid (right)

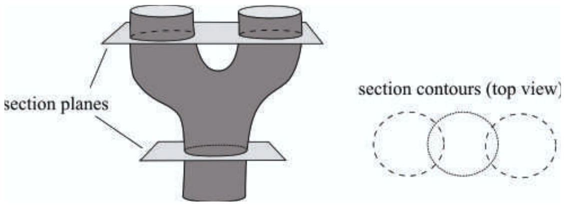


Figure 2.5: A simple example of branching

2.5.2 Implementations

The following implementations make no use of the original surface and base their solutions on the presence of contours only. Subsequently, two implementations that do use the original surface and topology are presented.

In one of the earliest publications, Keppel [18] defined the correspondence problem as a graph theory problem (figure 2.6). The optimal path in a graph was defined by using a cost function maximizing the resulting volume of the contained polyhedron. Later on, similar methods were used minimizing surface area [10] or summed span length. A span is defined as a line segment which represents a correspondence between two points on C_b and C_t. All methods are rather arbitrary due to the under constrained nature of the problem. The minimum surface area is used often, although is has been proved that this generally leads to hyperboloid shaped solutions and thus, in general, not properly representing the original surface [3]. Most of these methods are of order n^3 or sometimes $n^2 \log(n)$, where n is the number of sample points at one of the contours, assuming an approximate equal number of points on both C_b and C_t. Note that this approach and others using triangular facets result in a so-called one-to-many correspondence. That is, one point on C_t can correspond with multiple points on C_b and vice verse.

Some heuristic methods exist as well: The VLM-ST project group decided to use an advancing front technique as used in general purpose meshing. The principle is depicted by an example in figure 2.7. Given a good initial condition or first span, the next span to be built can bridge between points i and $j+1$ OR between j and $i+1$, depending on the shortest distance between the midpoint of line ij to both $i+1$ and $j+1$. In the example, the latter has the shortest distance and thus determines the next span as indicated by the dotted line. Variations to this method exist as well but are omitted here. Order n can be achieved here.

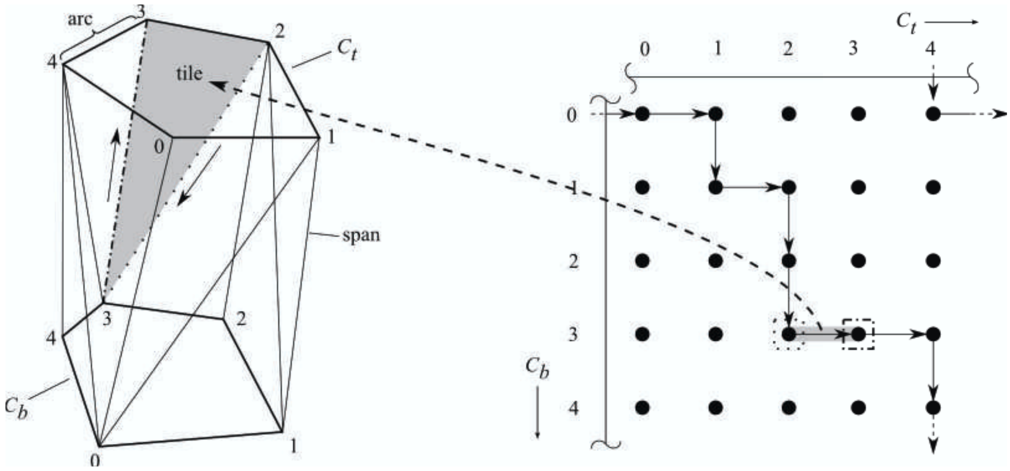


Figure 2.6: Graph theory approach to correspondence solving. The graph path at the right represents a possible solution as depicted left.

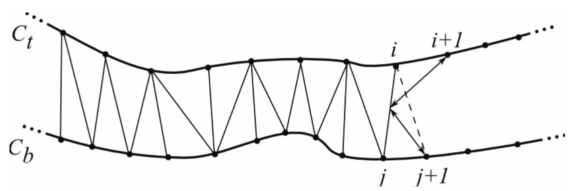


Figure 2.7: Advancing front meshing technique used for surface reconstruction.

Cohen et al. [7] proposes an order n^3 contour matching scheme based on the differential properties of the curves. The technique tends to be feature preserving, twist- and self-intersection free. It matches the directions of the unit tangent vectors. The optimal solution is one with maximum summed dot products (projections) of each pair of corresponding tangent vectors, see also figure 2.8. This method yields good results when using contours that do not differ too much from each other. However, in many cases it proposes a solution that does not represent the original surface with sufficient accuracy.

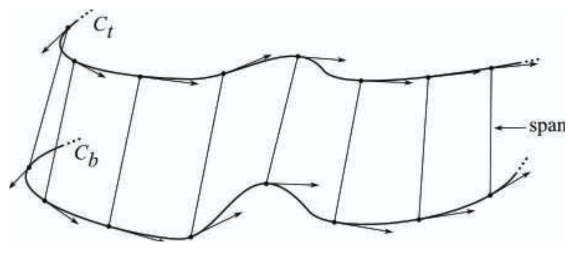


Figure 2.8: Tangent matching result

The following solutions use data from the original CAD model as well to solve for correspondence. The techniques stem from the RP field.

In the CAM-LEM project, Zheng [40] solves the correspondence problem by creating a one-to-one relationship between points at two consecutive sections, usually at high resolution with 'almost' order n complexity. The principle is depicted in figure 2.9. Half way between two sections, an additional middle section is used

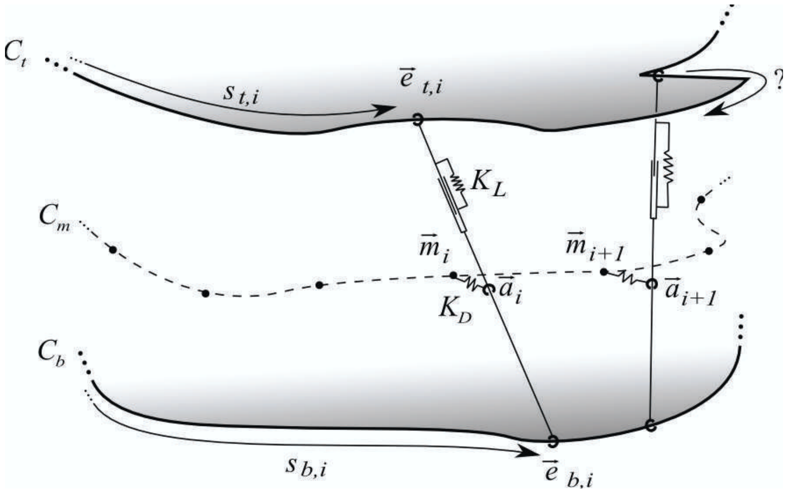


Figure 2.9: Example of the spring system used by Zheng to find a minimum energy solution.

to generate a set of sample points. At each of those points, a guided spring rod, connected with a ball-joint as shown. The rods represent the correspondence spans here. The springs are linear springs with no torsion stiffness. Both rod ends are connected to C_b and C_t with a sliding ring connection. The strain energy function of a single span is given by:

$$E_i(s_{b,i}, s_{t,i}) = \frac{1}{2}\left(K_D|\vec{D}_i|^2 + K_L|\vec{L}_i|^2\right), \text{ for } i = 1, \cdots, n \tag{2.3}$$

where the distance $\vec{D}$ is defined as

$$\vec{D}_i(s_{b,i}, s_{t,i}) = \vec{m}_i - \vec{a}_i(s_{b,i}, s_{t,i}) \tag{2.4}$$

with $\vec{a}_i$ as the acting point on the span and $\vec{m}_i$ as its attractor point on the middle section. $\vec{a}_i$ can be chosen to be either the middle of the span or the closest point to its attractor. Distance $\vec{L}_i$ can be expressed as

$$\vec{L}_i = \vec{e}_{b,i} - \vec{e}_{t,i} \tag{2.5}$$

The summed energy function then serves as the objective penalty function. The algorithm used makes sure the spans never cross each other, making the system non-linear and requiring creative solutions. Using a wavefront method, the system uses relaxation of the objective function. However, spans may touch each other somewhere on C_b and C_t, yielding a one-to-many correspondence relationship between C_b and C_t. When this is not desired, digital filtering techniques can be employed to alleviate this problem. One of the problems of Zhengs approach is the risk of getting stuck at a local minimum. This is shown with the right span in figure 2.9, which is 'trapped' behind a sharp corner at the top section. Solutions for this issue are mentioned by Zheng as well. The ratio of stiffnesses chosen should be such that the system converges fast enough yet stays stable. The ratio is also a compromise between approximation accuracy and ruled surface smoothness. The initial positions of the critical spans as used by Zheng were determined partially by existing techniques as well as following common logic. As good choice also speeds up the convergence of his solution. Zheng also offers various extensions and variations to the method proposed, each with its specific problems and opportunities.

The last correspondence solving method addressed is called 'topology traversal'. It uses the nominal CAD geometry topology to create spans at such locations that most features, such as sharp edges, are preserved [34]. It is depicted in figure 2.10. Simply stated, the algorithm determines which intersection contour segments at the i^{th} section matches a segment at section $i + 1$. When topology gets more complex, this method fails easily and needs user interaction or other methods such as those mentioned earlier to find a solution.

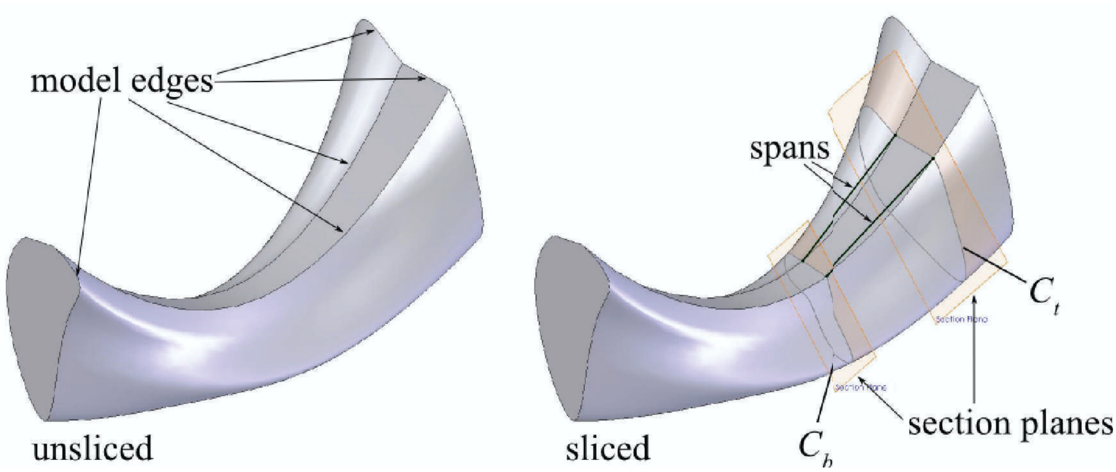


Figure 2.10: Usage of original CAD model topology for correspondence generation

After defining the spans using either of the previous methods, a set of two consecutive spans now bounds a ruled surface patch which is parameterized to define the correspondence between the spans. Each patch is linearly interpolated in terms of arc lengths. Applying it as in figure 2.11 by expressing the top parameter as a function of the base parameter, we get:

$$s_t(s_b) = s_{t,i} + \frac{s_{t,i+1} - s_{t,i}}{s_{b,i+1} - s_{b,i}}(s_b - s_{b,i}), \tag{2.6}$$

with $s_{b,i} \leq s_b \leq s_{b,i+1}$.
For every point on the contours, the correspondence is now known with enough information to generate the tool paths.

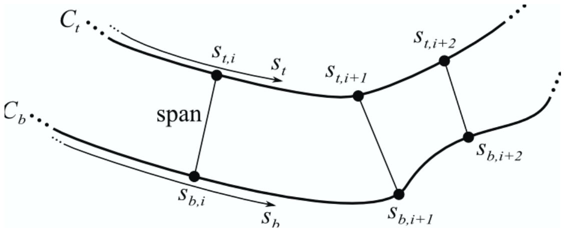


Figure 2.11: Base and top curve with spans indicating the correspondence solution at sample locations

2.6 Tool path generation

Every correspondence solving method mentioned before can be used to generate a one-to-one correspondence between C_b and C_t. This is often highly desirable for tool path generation as excessive burning is avoided this way. When a one-to-one correspondence exists, this can be used directly to generate the tool path. However, in order to obtain high accuracy, compensation for kerf width (figure 2.12) should be taken into account by the kerf width correction (KWC), which lies in the order of 1 mm for HWC. Further treatment of optimal KWC calculation goes beyond the scope of this work.

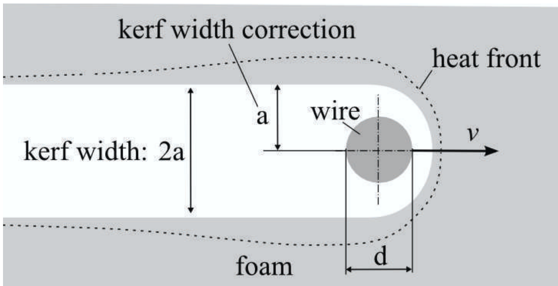


Figure 2.12: Kerf width geometry

Assuming one-to-one correspondence, the result after assembling the layers is depicted in figure 2.14 on the far left. However, when triangulation was used for correspondence solving, the correspondence has a one-to-many relationship and the denominator in equation 2.6 evaluates to zero in many cases. Direct use of this correspondence can result in excessive burning of the material due to low local wire speeds. In this case, another approach is desirable, as used by the VLM-ST group [24]. Given a layer with one-to-many correspondence, the set of facets as obtained from correspondence solving (not a STL mesh!) is intersected half way C_b and C_t, resulting in C_m. The tool path orientations are then calculated as $\vec{n}_i \times \vec{t}_i$, as shown in figure 2.13. This method can also be applied to the exact CAD surface without solving for correspondence as a separate step. In that case, think of the facets as in figure 2.13 to be replaced by the original CAD surface.

A disadvantage of the cross-product solution for tool path generation is the sawtoothed profile that results after fusing the individual layers together, as depicted in the right of figure 2.14. The reason is that the tool path at the level of a

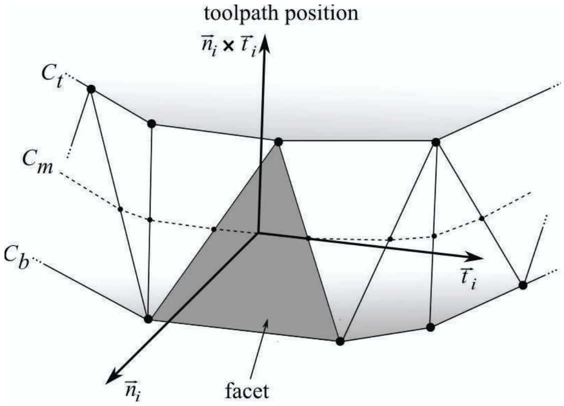


Figure 2.13: Toolpath generation using a one-to-many correspondence

section contour height does not coincide with the original contour itself. Another kind of defect, this time for a single layer (figure 2.15) results in an undercut at the corner of a faceted part. The Truesurf [13] project offers both the arclength parameterization using C_b and C_t and the cross-product method using C_m.

Tightly connected to the cross-product method in literature [13] is the offset, required to meet zero in- or outside tolerances, depending on the post-processing available. Figure 2.14 briefly shows different possibilities. The methods for this subject are beyond the scope of this work.

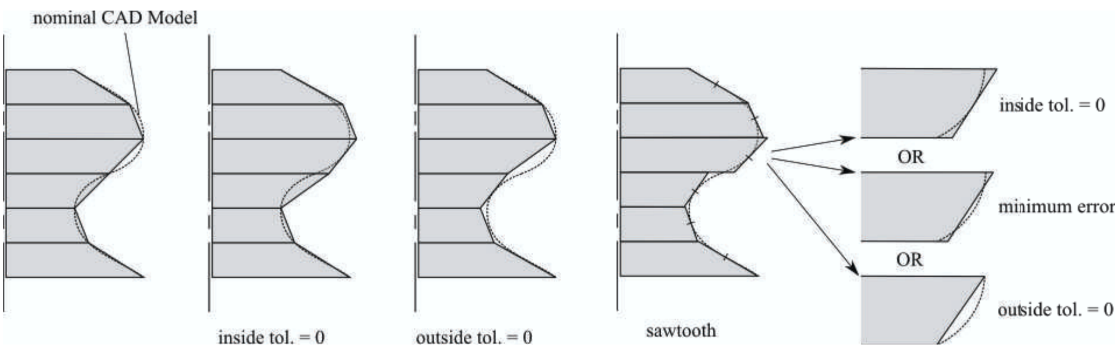


Figure 2.14: Results after generating tool paths and fusing layers together

When the maximum tool angle is exceeded, various solutions can be offered. Brink et al. [4] approximate a ruled surfaces which is too steep by one having the maximum tool angle. This would require additional post-processing. Stated differently for the case of HWC, the layer geometry is adjusted to conform the maximum wire angle.

2.7 Layer thickness estimation

The ruled surface approximation of an arbitrary nominal parametric surface S is often implemented with the requirement of a single user-defined maximum allowable error, also known as 'cusp height', denoted by δ [21]. It represents the

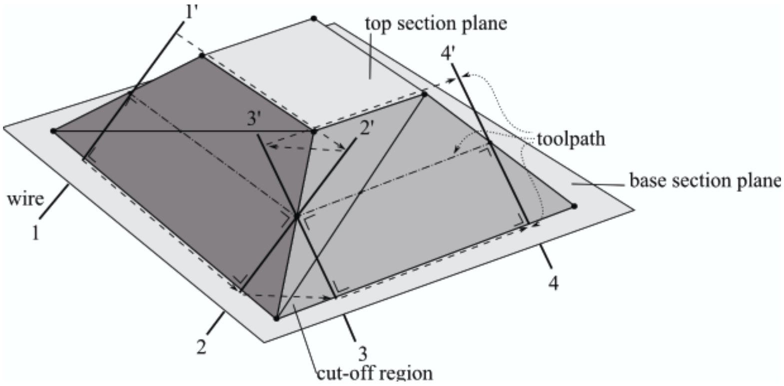


Figure 2.15: Triangular tessellation with corner defect, caused by generated tool path

maximum distance between S and the approximating ruled surface R. Although δ is often not defined accurately, it is usually measured along the normal $\vec{n}_S$ of S, which is also done here. Variations exist [28] where the user can define multiple values for δ for different model regions, but this is omitted here. Here, δ represents the *maximum allowed* 'normal deviation', which may also be known as the 'chordal error'. With the use of adaptive slicing, layer thickness is often determined using the curvature of S in build direction together with δ. In regions with high curvature, this generally results in smaller layer thicknesses than in regions with low curvature. In literature, a description analogous to the following can be found in [15]. The procedure is described in the following paragraphs, introducing the parameters as summarized in figure 2.16 top-down.

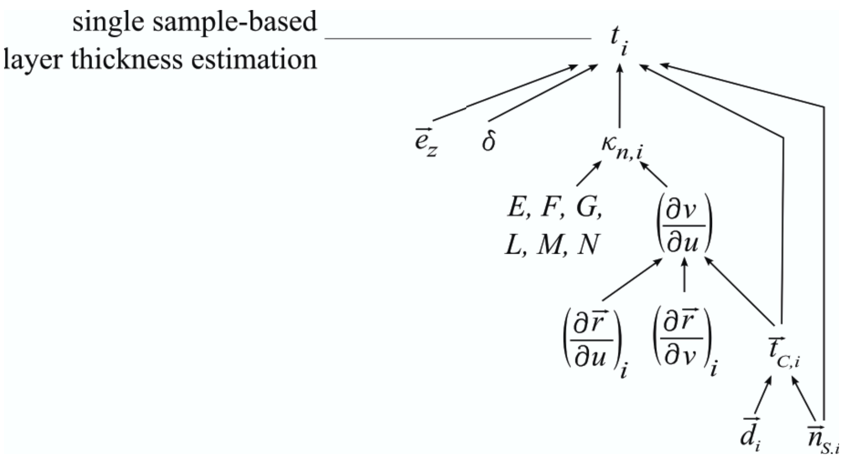


Figure 2.16: Parameter dependencies

Suppose that at a given moment in the slicing procedure, the next layer thickness needs to be established. The top section of last finished layer now serves as the base section for the next layer. Figure 2.17 shows a single sample point X_i on the new layer base contour C_b with known $\vec{n}_{S,i}$. Build direction $\vec{e}_z$ is known as well and is assumed coincident with the base section plane normal. For explanatory purpose, a 'normal plane' P_i is defined, spanned by $\vec{n}_S$ and a unit 'direction vector' $\vec{d}_i$, so the normal of P_i at X_i can be expressed by

$$\vec{n}_{P,i} = \vec{n}_{S,i} \times \vec{d}_i \tag{2.7}$$

In literature [15, 21], substitution $\vec{d}_i = \vec{e}_z$ is often used, defining P as the 'normal vertical plane' (NVP). The following applies to the more general case where $\vec{d}_i$ is

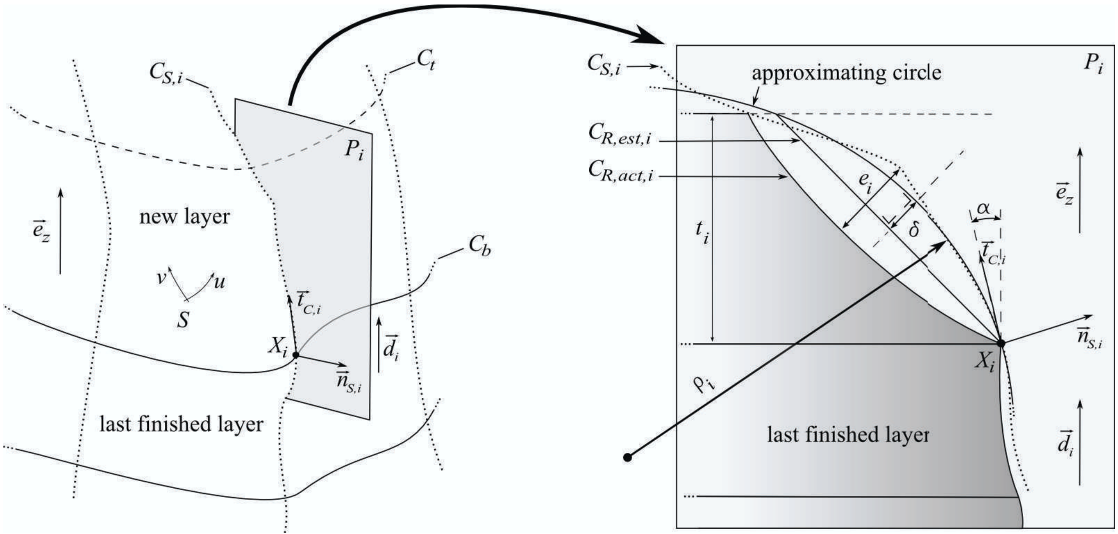


Figure 2.17: Curvature determination in plane P, here spanned by $\vec{n}_S$ and $\vec{e}_z$

not necessarily coincident with $\vec{e}_z$. Although this generalization was not explicitly found in literature, it is still mentioned here because it is strongly related to existing thickness estimations using substitution $\vec{d}_i = \vec{e}_z$. It involves geometry in P_i, as shown at the right of figure 2.17. Denote the curve that results from the intersection of S by P_i as $C_{S,i}$, the actual intersection of R by P_i as $C_{R,act,i}$. Now approximate $C_{S,i}$ at X_i by a circle with radius ρ_i whose value is determined shortly, and approximate $C_{R,act,i}$ by the linear 'estimation' $C_{R,est,i}$. Using these approximations, geometric considerations in appendix A then help to show that the i^{th} sample based layer thickness estimation t_i can be expressed by

$$t_i = \left(\sqrt{\rho_i^2 - a_i^2} - \rho_i\right) \vec{n}_{S,i} \cdot \vec{e}_z + a_i \vec{t}_{C,i} \cdot \vec{e}_z, \tag{2.8}$$

with

$$a_i = \left[8\delta\rho_i - 20\delta^2 + 16\frac{\delta^3}{\rho_i} - 4\frac{\delta^4}{\rho_i^2}\right]^{\frac{1}{2}} \tag{2.9}$$

and $\vec{t}_{C,i}$ as the unit tangent vector of $C_{S,i}$ at X_i. The radius of curvature ρ_i has to be determined in order to evaluate expression 2.8. In general, the radius of curvature of a space curve is the reciprocal of its curvature:

$$\rho = \frac{1}{\kappa} \tag{2.10}$$

The curvature of $C_{S,i}$ at X_i is defined as the 'normal curvature' $\kappa_{n,i}$. Its value depends on the direction of $\vec{t_i}$ along $C_{S,i}$, which is determined by the rotational orientation of P about $\vec{n}_{S,i}$. This orientation is defined by $\vec{d_i}$. Theory for determination of κ_n for parametric surfaces is given in appendix B. The result can be represented by

$$\kappa_n = \frac{L + 2M\dfrac{dv}{du} + N\left(\dfrac{dv}{du}\right)^2}{E + 2F\dfrac{dv}{du} + G\left(\dfrac{dv}{du}\right)^2}, \tag{2.11}$$

with E, F, G and L, M, N representing the first and second fundamental matrix elements of the parametric surface, respectively, which are constants when evaluating curvature in an arbitrary direction at a specific point X_i. Using $\vec{r}(u,v)$ to describe S, du and dv can be solved from

$$\vec{t_i} = \left(\frac{\partial\vec{r}}{\partial u}\right)_i du + \left(\frac{\partial\vec{r}}{\partial v}\right)_i dv, \tag{2.12}$$

where the subscripts denote evaluation at point X_i. Note that the actual intersection curve does NOT have to be calculated for the steps above. Vector $\vec{t_i}$ can easily be obtained after projection of the direction vector on the tangent plane (not shown) along S at X_i:

$$\vec{t_i} = \frac{\vec{d_i} - (\vec{d_i}\cdot\vec{n}_i)\vec{n}_i}{|\vec{d_i} - (\vec{d_i}\cdot\vec{n}_i)\vec{n}_i|} \tag{2.13}$$

In the relations above, the estimated section curve of R is assumed to be a straight line as in figure 2.17. However, this will generally not be the case. Remember that this procedure is used only for layer thickness estimation. The correspondence solution is most probably determined *after* thickness estimation, possibly resulting in a different -nonlinear- intersection curve. Take, for example, a hyperboloid as shown at the right of figure 2.4. Assume the correspondence solution to be represented by the same figure. use the hyperboloid as the nominal shape with substitution $\vec{d_i} = \vec{e}_z$ along the axis of symmetry. Now $C_{R,act}$ deviates from $C_{R,est}$ in the NVP. This implies that the actual error e, measured along $\vec{n}_S$, might exceed δ. Apart from the deviating tool path, the difference in actual error might also be a consequence of the circular approximation. Both contributions are visible in figure 2.17.

The procedure can be repeated for a set of sample points on C_b. Each sample yields a layer thickness estimation, based on the local geometry. The new layer thickness estimation can now be conservatively chosen from n samples as

$$t^* = \min\{t_i \mid i = 1 \ldots n\}, \tag{2.14}$$

Where t^* denotes the critical thickness. The amount of samples and their distribution is a cause for concern. Too little samples could lead to a bad estimation while too many samples would increase computation time. Typically, a proper compromise between the two should be chosen.

The maximum and minimum normal curvatures $\kappa_{n,1}$ and $\kappa_{n,2}$ at a surface point are called the principal curvatures. It can be shown [9] that the principal directions, corresponding to these curvatures, are orthogonal. Now define the orthonormal 'principal frame' $(\vec{e}_1, \vec{e}_2, \vec{n}_S)$ with $\vec{e}_1$ and $\vec{e}_2$ the normalized principal directions. When the principal curvatures and -directions are known, the normal curvature in an arbitrary direction can be obtained by application of the *Euler formula*:

$$\kappa_n(\phi) = \kappa_{n,1} \cos^2(\phi) + \kappa_{n,2} \sin^2(\phi) \tag{2.15}$$

where ϕ is the angle in the tangent plane from $\vec{e}_1$ to $\vec{t}$ in the principal frame. An example is depicted in figure 2.18. In this case, $\kappa_{n,1} = \frac{1}{R}$ and $\kappa_{n,2} = 0$. This relation has not been encountered by the author in RP literature, although it makes use of the principal curvatures which play a key role in the field of differential surface geometry. In spite of its simple appearance, derivation of this relation requires a lot of effort and goes beyond the scope of this work, but was done by Euler in 1760. In general, the principal curvatures can be evaluated easily once the fundamental matrices are known. Once the principal curvatures are known, checking for singularities is not necessary anymore, in contrast to equation 2.11.

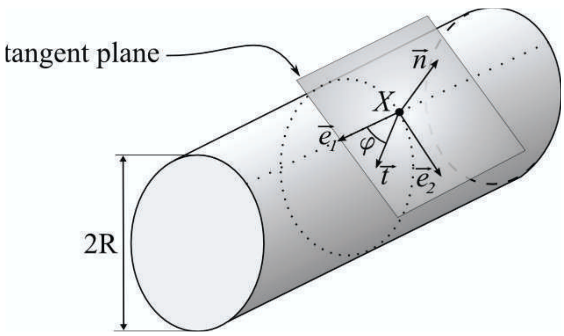


Figure 2.18: Principal curvatures

Banerjee et al. [2] substitute $\vec{d_i}$ not only by $\vec{e}_z$, but also by $\vec{e}_1$ and $\vec{e}_2$, assuming the tool path to lie in one of these three directions during the entire cut. However, range limitations of the cutting machine would often render such an approach less suitable. It was concluded [2] that taking the 'maximum absolute curvature direction' (the direction corresponding to the principal curvature $\kappa_{n,pr}$ for which $|\kappa_{n,pr}| = \max\{|\kappa_{n,1}|, |\kappa_{n,2}|\}$ holds) yields a maximum number of slices but the smallest volume difference error. The volume difference error is described in section 2.8. The 'minimum absolute curvature direction' yields a minimum number of slices but the largest volume difference error. The NVP-approach comes out as a compromise of the two.

2.8 Error analysis

After approximating the nominal surface by a ruled surface for a single layer, the approximation error can be estimated. Its magnitude generally depends on layer thickness, model shape, correspondence solution and error sampling rate.

Hope et al. [15] identify a representation for the actual error other than δ. The maximum distance in the layer plane, ϵ, is taken as a measure for the volume difference between the nominal and approximated model, while the δ usually presents a better roughness representation. Figure 2.19 illustrates both δ and ϵ. When the nominal surface normal approaches the build direction, ϵ is less representative for volume difference. In that case, cusp height gives a better representation for both surface roughness and volume difference.

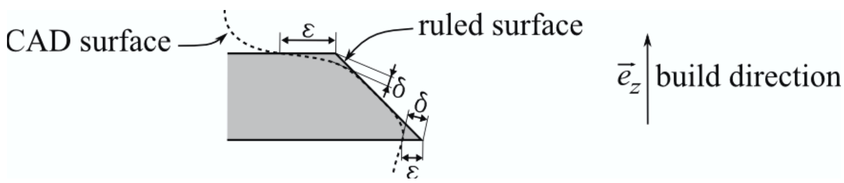


Figure 2.19: Two measures of error: cusp height δ and layer plane error ϵ.

Koc [20] proposes a marching algorithm which samples the actual error. The sampling is taken along an intersection curve, created by intersection of a plane and the nominal surface. The plane is spanned by the vector between two corresponding points P and P' and the average of $\vec{n}_P$ and $\vec{n}_{P'}$ as shown in figure 2.20. The curve itself is not calculated, but the sample points are obtained through the

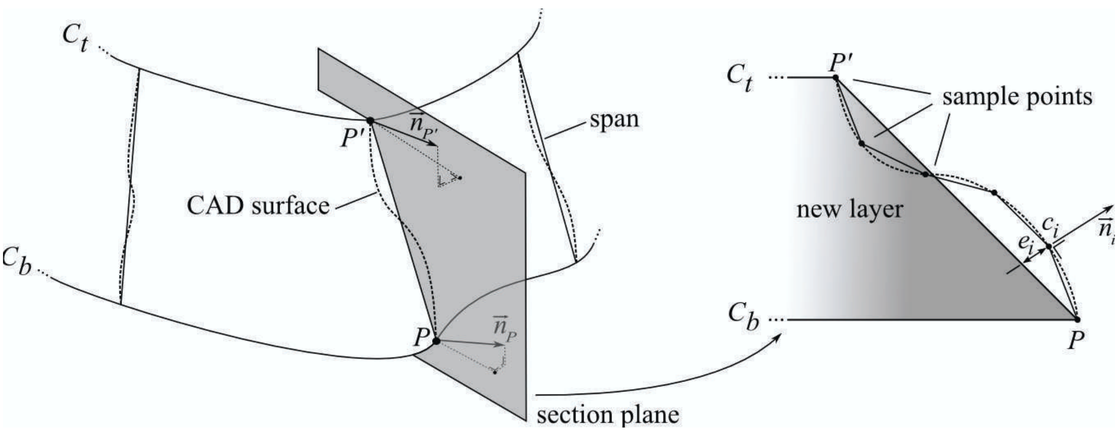


Figure 2.20: Marching algorithm

differential properties of the surface. The samples are taken from the base section to the top section. Sampling size and distribution do not receive much attention.

Kumar and Choudhury [22] proposes a different approach. The ruled surface is approximated by a set of parametric bilinear patches, also known as cubic spline patches. For each bilinear patch, the corresponding model surface patch is identified and sampled in u and v direction. Next, the error along the nominal model surface normal $\vec{n}_S$ at the sample points is evaluated by line/patch intersection. The method is depicted in figure 2.21. Only the surface region 'bounded' by the four patch points is examined from the entire exact CAD surface. Disadvantage is the introduction of yet another approximation, this time of the ruled surface itself. The patches must be sufficiently small, increasing computational cost. Also, the

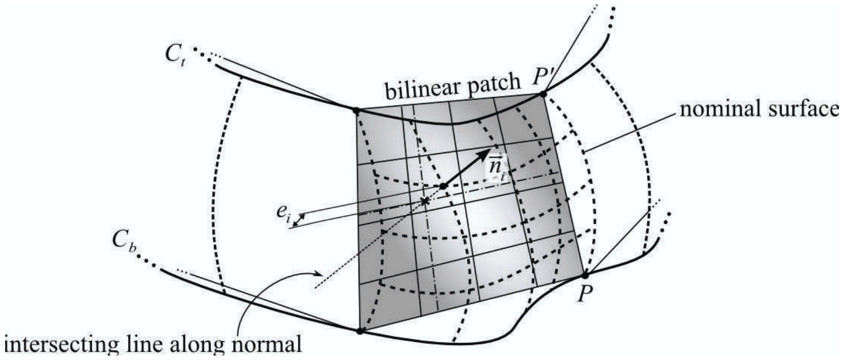


Figure 2.21: Bilinear patch approximation

risk of intersection with a patch point outside the patch area is present, possibly yielding misleading results. This effect should be taken into account. An alternative approach mentioned is to determine the error along the normal of the bilinear patch (not shown in the figure). It was observed that application of this approach yielded more and thus thinner slices than the situation with only the layer thickness estimation used for error estimation. This is as expected as layer thickness estimation has its limitations as discussed in section 2.7.

The difference in volume between exact CAD model and the sliced approximation also poses an error quantifier. This can be evaluated per layer or for the entire model. Geometric modelers usually offer tools in order to calculate this. However, this approach only gives global information. The difference in volume may be very small, indicating valid solution, yet locally, the actual error may exceed user-defined limits.

The angle between the ruled surfaces of two subsequent layers may also serve as a measure of error. When a smooth body is required with little post-processing, this measure of error may be appropriate.

The methods above are mostly based on sampling and thus represent an estimation of the true maximum error. Higher accuracy comes with increased computational cost. In order to come up with a suitable error estimation, the objective must be known first.

2.9 Physical fabrication

As mentioned in section 2.1, various hardware solutions can be found. Five-axis laser and water jet cutters are used. Most of the time, the table holding the layer sheet is rotated and the cutter head is kept static, as is shown in figure 1.6. The systems often allow a maximum rotation of 30 to 45 ° around each axis and layer diameter dimensions are in the order of 500 mm. HWC is used as well, but it suffers from the same angular restrictions, depending on portal distance and controlling software capabilities. EDM is used as well as shown before. EDM and HWC have the disadvantage of not being able to cut inner contours without intersecting the part itself, unless extra measures are taken, such as drilling holes inside inner contours and detaching the wire, putting it through the drilled hole, reattaching it to the machine, then cut, etcetera. Most of the time, machines do not facilitate this. It would lead to high operator cost when repeated often and cause tedious labor. Some systems are supplied with a sheet feeder, which automatically feeds the cutter with an uncut sheet of paper, foam of whatever material is used. When using adaptive slicing, this requires ordering of the sheets as sheets with different thicknesses follow each other.

The VLM-St project [1] mentions the use of guidance pin holes in the part for accurate alignment of the individual layers. Also, connectors can be generated by software where multiple disconnected contours are cut in one layer, just for proper positioning as well.

The fusing of the layers can be done manually or automatically, but literature does not mention much about this subject regarding TLOM systems.

Post processing can be subdivided in incremental and decremental. Decremental methods typically involve sanding or milling. Incremental methods may involve filling or coating. Both forms likely involve higher cost and should be avoided where possible. In general, first order slicing greatly alleviates the post-processing actions required when zero-order slicing would have been used.

Chapter 3

Analysis

This chapter contains the requirements for the design of a RP tool for the HWC process, based on the demands of 3EL-Company and additional limitations. Next, decisions are made regarding the best suitable methods to meet these requirements. First, cases are presented to give an impression of typical shapes used as input for slicing.

3.1 Typical cases

A few examples of various shapes from the 3EL-Company archives are shown in figure 3.1. An example of a wind turbine blade model is depicted at the bottom of the figure. Most of the examples are boat hulls, but the upper left example is the bare shape of an experimental aircraft body, which will serve as a demonstrator throughout this thesis.

3.2 General requirements

A set of requirements for the overall programme was set up and is presented here. In order to get an effective slicing tool, it must satisfy the following global requirements:

- generate a 2.5D ruled surface approximation of the exact CAD model,
- perform layer thickness optimization,
- comply with a user-specified tolerance field,
- preserve features such as sharp edges as much as possible,
- offer generation of a continuous part surface (no saw-tooth effect),
- offer user-defined uniform slicing,
- include feasibility checking of the wire angle,
- include surface quality check.

All of the requirements are derived from basic customer needs. In many cases, the customer does not give explicit tolerance values, yet often the maximum obtainable accuracy plays a role. Obviously, a better accuracy usually comes with increased cost through the larger number of layers and their assembly. It is advan-

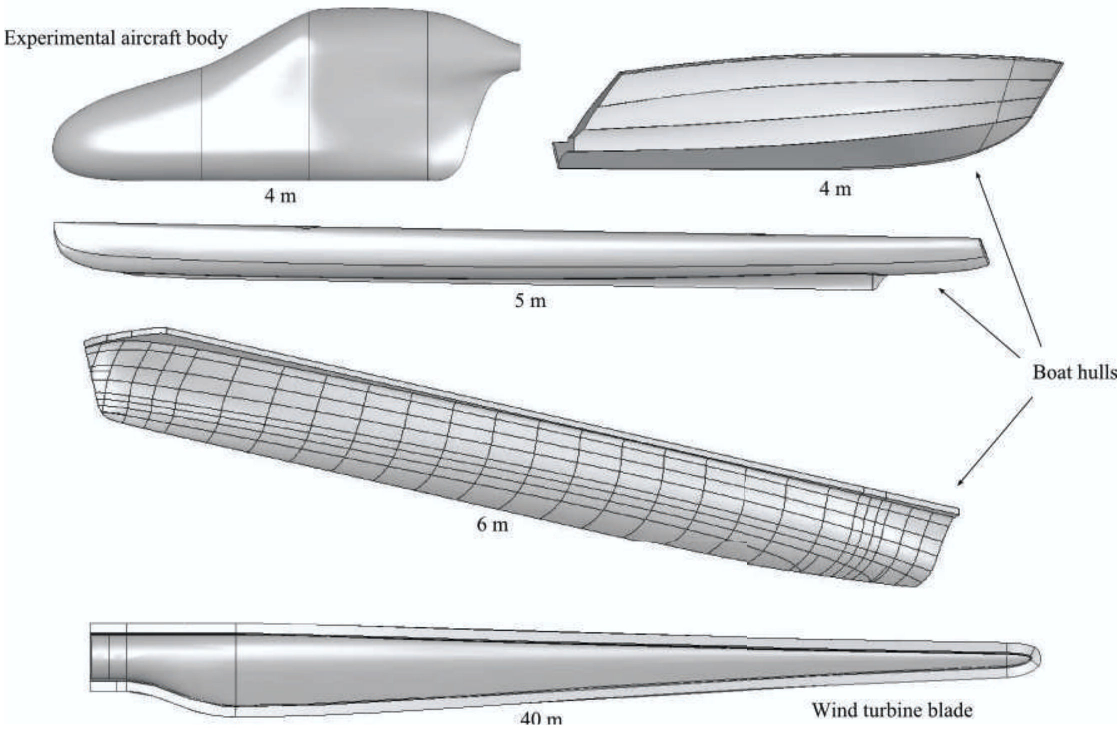


Figure 3.1: Various examples which have been fabricated with HWC. Note that the examples are drawn in different scales.

tageous for the HWC engineer to be able to, for example, simulate various slicing scenarios with variable tolerance values and evaluate the corresponding cost. However, the cost aspect is not analyzed here. Furthermore, the software tool can assist in decisions regarding manufacture feasibility.

3.3 Master project requirements

Decisions have been made regarding in- and excluded requirements for the Master project, based on fitness for automation and available development time.

Included requirements:

- offer first order surface approximation,
- offer adaptive slicing,
- offer user control over available layer thickness set,
- offer a user-defined build direction,
- preserve topology (sharp edges in correspondence solving),
- generate continuous part surface (no saw-tooth or staircase effect),
- offer actual error evaluation: normal deviation,
- offer a check of the maximum machine wire angle,
- offer a check of surface quality,

Most important non-included requirements:

- CAD model decomposition and segmentation (e.g. from end features),
- automatic calculation of optimal build direction,
- varying build direction through the model,
- zero order slicing,
- feasibility checking w.r.t. part size and -location in machine,
- branching in build direction,
- correspondence solving with user interaction,
- zero in- or outside tolerance fields,
- generation of pilot pin-and-hole and connectors,
- specialized CAM file format (with correspondence info),
- graphical user interface.

Both decomposition and segmentation have been excluded from the master project for the reasons mentioned in section 2.3: high expected development cost and limited value for the proof of principle.

The determination of the optimal build direction is assumed to be a relatively simple task for the user, which is plausible with respect to shapes such as shown in figure 3.1. It has therefore no primary priority.

The build direction is kept the same through the segment because of increasing complexity of technical issues, both practical and computational.

Zero order slicing is left out as it does not make good use of the four-axis HWC process, although it could be implemented with relative ease once computational first-order slicing is possible.

Part size and location in the machine also affect feasibility and are left to the user's responsibility as well. It would, amongst others, require the software to have knowledge of the overall machine dimensions and kinematics, which could be variable when the tool is used for production on different machines, demanding a proper machine database. Also, the location in the machine can be optimized for, just as optimally nesting the layers in a source sheet.

Branching is left out as it can be avoided easily by manual segmentation. It would also require specific attention as mentioned in section 2.4: Many of the 'most intelligent' algorithms fail miserably in even the more simple cases. Apart from this, no primary demand is present as can be derived from figure 3.1.

Correspondence solving with user interaction would also require further analysis, yet this is expected to be manageable. However, the possibly complex user/automation interaction would be of primary interest here, which does not contribute to general insight into the slicing process in this Master Project.

Zero in- or outside tolerance are excluded as it requires specific attention. It requires methods like offsetting section contours. However, the solution used also depends on the requirement whether or not a continuous outer surface is required or not, as explained in section 2.6. A possible workaround can be offsetting the original CAD model outer surface with the maximum allowed cusp height. Next, the offset model can be sliced using the same cusp height as a measure of maximum allowable normal deviation.

Pilot pin-and-hole and other facilitating features are left out as the layers are assumed to be alignable by themselves. When one or more sharp edges are present, this is generally the case, although it must be treated with great caution. Usually, alignment shapes can be integrated with the part design.

Finally, a specialized CAM file format design is left out. This file should contain enough information to provide the machine controlling soft- and hardware with cutting data and therefore requires cooperation of the HWC machine manufacturer.

3.4 Practical requirements and constraints

The machine used for the project is a Step Four®PC-CUT 5000 series HWC machine, made in Austria. It is shown in figure 1.2.

PC-CUT 5000 machine properties:

- four independently controlled axes,
- manual variable distance between the portals (250-5000 mm),
- max. dimensions in the layer plane: 5400 x 1460 mm (equals raw block sizes),
- max. wire angle: set at 45 ° (in software).

Machine software properties:

- dedicated control,
- very basic 2D CAD environment,
- simulation (along 2D contours),
- DXF (2D contours) / HPGL (very basic) import,
- Step Four SCF-file format with correspondence information,
- operating system: DOS.

The 2D drawings can best be imported through the DXF interface, but remain without any information regarding correspondence relationships or KWC. Correspondence is subsequently applied either manually or automatically by contour matching using a minimum change in contour angle between two successive polyline segments. This easily leads to solutions that are not intended by the user or even unacceptable surfaces. It would require extra manual effort for the more subtle regions of correspondence. Tool path calculation is done using linear arc-length parameterization as described in section 2.6.

Step Four has developed its own SCF cutting data file format which stores both the contours correspondence data and other HWC-specific information. However, this file format has not yet been used. Instead, the DXF file format is used. Hence, at the moment, no practical way exists to import correspondence information into the Step Four software, rendering this information lost after the contours have been imported. This leads to the restriction that for the time being, correspondence should be easy to reconstruct by the Step Four software operator. Once the file format is put at 3EL-Company's disposal, this issue should be resolved to a large extent.

In order to review the computational slicing process, the user must obtain a file with the results. This requires that the slicing process documents its progress in a log file, allowing the user to change input parameters based on the log file data.

To summarize, the following additional requirements are defined:

- offer DXF file export option,
- offer log file generation.

It was decided to let the user set a maximum allowed wire angle θ_{max} which is the maximum angle between the wire and the normal of the portal planes as shown in figure 1.1. The actual wire angle θ must not exceed this value. The largest actual wire angle is denoted by θ^*. When no layer thickness solution is found for $\theta^* < \theta_{max}$, the part is defined as not feasible. Other solutions can be proposed instead, e.g. involving cutting with lower wire angle than actually needed, resulting in additional post-processing. These solutions have been omitted here.

3.5 Methods

In this section, the chosen solutions for the various steps in the RP chain are presented and substantiated.

3.5.1 CAD model import

The choice between direct and tessellated slicing caused quite some discussion with various arguments put forward. The essential characteristics of both are captured in section 2.2. It was finally decided that the direct slicing method is favorable. The most important reasons are:

- chosen method for correspondence solving (see below),
- strongly reduced development/implementation time.

3.5.2 Correspondence problem

The solution chosen for the correspondence problem is the topology traversal method as presented in section 2.5.2, using linear arc length interpolation between the spans. Direct slicing greatly facilitates this because exact topology is present. This method will is supposed to best suit the requirement of feature preservation. A broad range of objects like the mentioned cases is expected to be successfully produced using topology traversal. Zheng's method could complement this approach, but is not implemented. When used without topology traversal, it poses the risk of getting stuck at a local minimum instead of reaching the global minimum in the optimization scheme. This very well might happen at sharp corners, which are often characterizing features of the resulting foam structure and should therefore be accurately described. Solutions exist to part of these problems, but automated results should still be regarded with great caution.

In general, user interaction remains inevitable for obtaining the correct solution of the correspondence problem when confronted with geometries are more complex. Assuming not only contour data but also the original surface to be present, this statement still holds, although more information is available. The problem remains ill-constrained.

3.5.3 Tool path generation

Once the correspondence has been determined, the tool path is generated assuming a one-to-one correspondence. For layer L, define a base- and top section contour C_b and C_t, both parameterized by arc lengths s_b and s_t, respectively. Assume a known mapping from s_b to s_t, defined by the correspondence solution:

$$F : s_b \mapsto s_t. \tag{3.1}$$

A contour generally consists of multiple contour segments. A segment is described by a parametric curve, bounded by a start and end parameter. Let F take this into account when evaluating. At the i^{th} location of s_b, the coordinates on base- and top contour are, respectively,

$$\vec{r}_{b,i} = C_b(s_{b,i}) \tag{3.2}$$

and

$$\vec{r}_{t,i} = C_t(F(s_{b,i})), \tag{3.3}$$

Define $\vec{c}$ as the unit ruling vector (see section 2.5.1 for the definition). At $s_{b,i}$, we get

$$\vec{c}_i = \frac{\vec{r}_{t,i} - \vec{r}_{b,i}}{|\vec{r}_{t,i} - \vec{r}_{b,i}|}. \tag{3.4}$$

Using this approach in a discrete or continuous sense, the tool location and orientation can be evaluated along the entire boundary of L. That is, without any kerf width correction applied.

3.5.4 Error analysis

Sampling of the normal deviation of the nominal surface S is chosen to represent an estimation for the actual error e. The maximum absolute actual error is denoted by e^*. The normal deviation or the distance along normal $\vec{n}_{S,i}$ to the ruled surface intersection point is calculated at a user-defined sampling rate (see figure 3.2). Define R_{set} as the set of rules surfaces approximating S at layer L. When the thickness of L and correspondence solution have been established, an intersection half way the section planes of C_b and C_t is created, yielding a contour C_m on S which is then sampled by arc length s. The choice of the section to be half way the layer is chosen because that is the location where e is assumed largest, given the fact that the normal error equals zero at C_b and C_t. At each sample point P_i, the surface normal $\vec{n}_i$ is evaluated. P_i and $\vec{n}_i$ can now define a line l which is then used to intersect the surfaces in R_{set} of L. The distance along l between P_i and the geometric intersection $\vec{x}_i$ of R_{set} then represents the local actual error e_i. Note that the line can intersect R_{set} multiple times so the correct intersection should be selected.

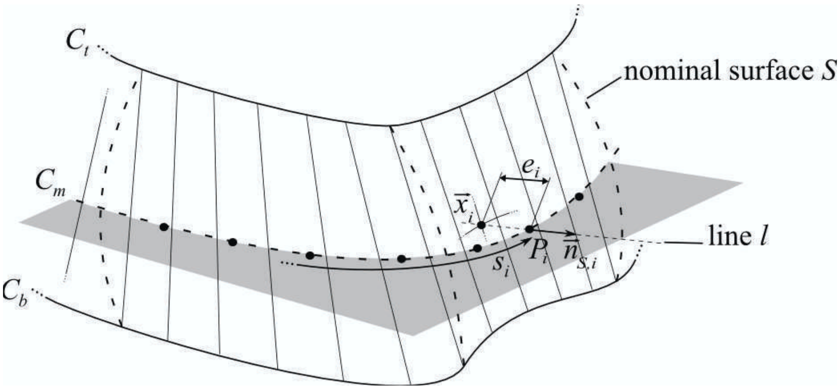


Figure 3.2: Normal deviation sampling along nominal surface normal

The error check can be stopped once $e_i > \delta$. The procedure can be repeated for multiple section heights.

3.5.5 Surface quality

Using HWC, manufactured part surface regions with relatively low local wire speed can be characterized by a higher surface roughness, higher surface foam density or excessive burning, depending on orientation with respect to gravity, causing buoyancy of hot air to form 'chimneys' in the foam in extreme cases. The cause of this phenomenon is schematically illustrated in figure 3.3. The amount of melting

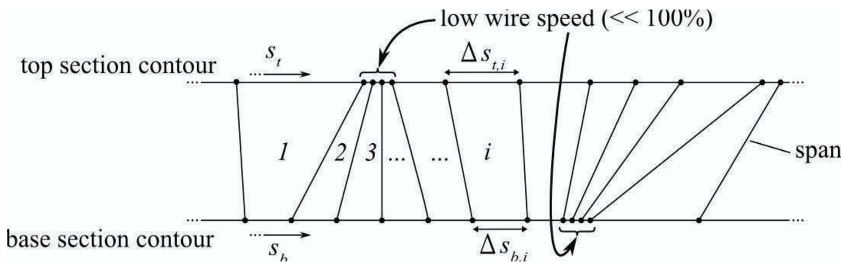


Figure 3.3: Schematic correspondence solution including regions of low wire speed

can be characterized by the arc length ratio $Æ$. The arc length ratio is defined by

$$Æ_i \equiv \frac{\min\{|\Delta s_{t,i}|, |\Delta s_{b,i}|\}}{\max\{|\Delta s_{t,i}|, |\Delta s_{b,i}|\}} \tag{3.5}$$

and the minimum arc length ratio at a single layer is defined by

$$Æ^* \equiv \min_i\{Æ_i\} \tag{3.6}$$

Surface roughness typically significantly increases when $\mathcal{AR}$ is in the range of 0.6 - 0.2. At lower values, which approach the case of one-to-many correspondence, excessive melting can occur at the contour with the smaller arc length.

A part of the solution is to compensate for the kerf width. However, KWC cannot always fully compensate for excessive melting behavior. An accurate way of KWC calculation is important, but left out here.

Another solution to excessive burning is, as mentioned in section 2.5, applying digital filtering techniques to reduce low wire speeds. However, this will probably result in a surface that is further away from the optimum, possibly canceling out some of the optional previous correspondence optimization effort. This solution is particularly useful when the ratio approaches zero or infinity, values where the wire will progressively burn into the foam surface.

It was decided to let the user define a minimum allowable arc length ratio $\mathcal{AR}_{min}$, based on his experience. When this constraint cannot be met, the part is defined as not feasible. In other words, this constraint tries to force the solution into one that avoids a one-to-many correspondence.

3.5.6 Layer thickness estimation

In order to arrive at an optimum layer thickness as fast as possible, an accurate thickness estimation is needed. The layer thickness estimation is made with the method from section 2.7. Using an iterative process, the optimal layer thickness is determined.

It was decided to limit the thickness solution space to a user-defined set of available sheet thicknesses. This yields the following advantages and limitations.

Advantages:

- the user can choose and limit the thicknesses for practical constraints
- the algorithm can be kept simple

Limitations:

- a custom thickness (to take end features into account) is not available
- human factor in choosing thickness set
- sub-optimal resulting layer thicknesses

In practice, using the current machine setup, use of too many different thicknesses is error-prone. Until present, most products have been manufactured using just one or two different thicknesses. It could be the case that after the computational slicing process, the user is not satisfied with calculated layer thicknesses for various reasons. In this case, the set can be modified using the new insights and the slicing process can be run again.

Denote a 'sheet thickness database' containing the available user-defined thicknesses by

$$T_{DB} = \{T_1, T_2, \ldots, T_N \mid T_i > 0 \wedge T_{i+1} > T_i \wedge N \geq 1\}. \tag{3.7}$$

Let a sample be denoted by subscript s. In principle, a layer thickness estimation, denoted as t_s, can be evaluated at n samples on base contour C_b for each iteration, according to the curvature relations from section 2.7. A proper sample set might be generated automatically. Here, the samples are be homogeneously distributed at a user-defined rate. Define the sample set as

$$X = \{s_1, s_2, \ldots, s_n \mid s_i \geq 0 \wedge n \geq 1 \wedge s_n < \sup(s_b)\}. \tag{3.8}$$

An iteration is denoted by subscript i. Using the foregoing relations and data, the i^{th} iteration for layer L can described. In order to do this, the direction vector $\vec{d_s}$ (section 2.7) must be known. Now use substitution

$$\vec{d_s} = \begin{cases} \vec{e}_z & \text{if } i = 1 \\ \vec{c}_s & \text{if } i > 1. \end{cases} \tag{3.9}$$

Note that at $i = 1$, correspondence F (equation 3.1) and hence ruling vector $\vec{c}(s_b)$ are unknown. Hence, $\vec{c}(s_b)$ is approximated by $\vec{e}_z$. In the next iterations, F is known so $\vec{c}_s$ can be evaluated using equation 3.4. After t_s-evaluation at each sample, t^* is obtained from equation 2.14 as $t^* = \min\{t_s \mid s = 1 \ldots n\}$. For this iteration, a curvature-based T_{DB}-index is then conservatively obtained by

$$\tilde{I}_i = \begin{cases} 1 & \text{if } t_i^* < T_1 \\ j \mid T_j \leq t_i^* \wedge T_{j+1} > t_i^* & \text{if } T_1 \leq t_i^* < T_N \\ N & \text{if } t_i^* \geq T_N \end{cases}, \tag{3.10}$$

where $\tilde{I}$ denotes the t^*-based list index of T_{DB}.

Define I_i as the final T_{DB}-list index of iteration i. Although $\tilde{I}_i$ is now known, it may or may not be chosen for I_i. Apart from $\tilde{I}_i$, a second proposal $\hat{I}_i$ is defined. Its value is $I_{i-1} + 1$ or $I_{i-1} - 1$. In general, $\hat{I}_i$ may differ from $\tilde{I}$. $\hat{I}_i$ is determined by values $Æ_{i-1}^*$, e_{i-1}^* and θ_{i-1}^* while comparing them to the constraints imposed by $Æ_{min}$, e_{max} and θ_{max}:

$$\hat{I}_i = \begin{cases} \tilde{I}_i & \text{if } i = 1 \\ I_{i-1} - 1 & \text{if } e_{i-1}^* > \delta \vee Æ_{i-1}^* < Æ_{min} \vee \\ & \left(\theta_{i-1}^* > \theta_{max} \wedge \dfrac{\partial \theta_{i-1}^*}{\partial t_L} > 0\right) \\ I_{i-1} + 1 & \text{if } \left(e_{i-1}^* \leq \delta \wedge \theta_{i-1}^* \leq \theta_{max} \wedge Æ_{i-1}^* \geq Æ_{min}\right) \vee \\ & \left(\theta_{i-1} > \theta_{max} \wedge e_{i-1}^* \leq \delta \wedge \dfrac{\partial \theta_{i-1}^*}{\partial t_L} \leq 0\right) \\ \text{undefined} & \text{otherwise,} \end{cases} \tag{3.11}$$

with t_L as the layer thickness of L. The choice for a unit change is based on the assumption that $T(I_{i-1})$ lies near the optimum layer thickness, although this is not necessarily the case. It is also assumed that

$$\frac{\partial e^*}{\partial t_L} > 0 \tag{3.12}$$

and

$$\frac{\partial Æ^*}{\partial t_L} < 0, \tag{3.13}$$

which are both assumed to hold at least in the limit $t_L \downarrow 0$. To be determined is

$$\frac{\partial \theta^*}{\partial t_L}, \tag{3.14}$$

which is derived from the sign of the κ_n and normal vector orientation at the sample point at C_b containing θ^*.

Finally, the T_{DB}-index for iteration i of L is determined:

$$I_i = \begin{cases} \tilde{I}_i & \text{if } \mathrm{sign}(\tilde{I}_i - I_{i-1}) = \mathrm{sign}(\hat{I}_i - I_{i-1}) \wedge |\tilde{I}_i - I_{i-1}| > 1 \\ \hat{I}_i & \text{if otherwise} \end{cases} \tag{3.15}$$

After thickness determination, a new section is created at an offset from the base section by the new thickness. This section can then checked for a single closed contour. Next, the correspondence for L can be solved. Finally, $Æ_i^*$, e_i^* and θ_i^* can be evaluated and stored together with the iteration i.

Notes:

In some other RP implementations, the layer thickness is unchanged when $e_i^* \leq \delta$. Although the layer is acceptable, it may not be optimal. It is possible that t_i^* an hence also $T(\tilde{I}_i)$ are too small, based on the limitations of κ_n-based thickness estimation. However, here, thickness is increased. This may eventually save cutting time because thicker sheets may be used.

When $\theta^* > \theta_{max}$, the thickness must be changed in direction of decreasing wire angle. Usually, this means that a thinner layer is desired. However, this is not necessarily the case, as near the nose of the upper leftmost case in figure 3.1. Building from left to right using the same case, the surface at the nose becomes less steep with respect to build direction. In that case, a larger layer thickness reduces θ yet may increase e^*. When, in this case, already $e^* > \delta$, $\hat{I}_i$ becomes indeterminate and the iteration deadlocks, rendering the geometry not feasible for production. A better manual start plane location, further away from the nose tip can solve this problem.

The next example is given to illustrate the merit of multiple κ_n-based thickness estimations in the consecutive iterations. Take a small patch which resembles a piece of a cylinder (see figure 3.4) from nominal surface S. At sample point P, one would intuitively predict a thickness of infinity, based on the given shape, build direction and expected tool path, denoted by the plotted spans. However, iteration 1 determines the κ_n-direction using substitution $\vec{d_1} = \vec{e}_z$. This yields $\kappa_{n,z} > 0$, locally yielding a finite (too small) thickness estimation. Next, the correspondence is solved for, parameter evaluations are performed and iteration 1 is finished.

Iteration 2 might offer a better estimation. Using the solved correspondence from iteration 1, depicted by the spans, substitution $\vec{d_2} = \vec{c}_P$ is applied, yielding $\kappa_{n,c} = 0$ and infinite estimated layer thickness. Note that topology should then dictate a maximum to the thickness.

Analogously, it can be argued that the sample thickness estimation could also be too large in iteration 1. The effect discussed here is further referred to as the

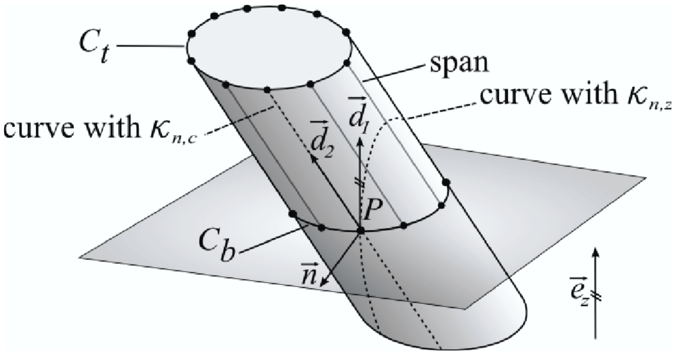


Figure 3.4: Approximation of surface (not shown) around P by a cylinder. The subscripts denote the first two iteration numbers

'skew cylinder effect'. Remember that t^* is ultimately used for layer creation. This means that the local over- or under-estimation of layer thickness does not necessarily lead to a bad thickness estimation.

It was decided to omit further use of improved κ_n-based thickness estimations for the third iteration onward. Instead, only unit increment DB thickness changes are used. This decision is based on the assumption that the directional change of $\vec{d}$ from iteration to iteration is relatively small after iteration 2. In other words, $\vec{c}$ is assumed not to change significantly. In general, this might not be the case. However, this was not investigated. It strongly depends on the shape and topology of the input model. Also, it should be remembered that a κ_n-based thickness estimation has its limitations as explained in section 2.7.

Chapter 4

Implementation

This chapter describes how the chosen methods are implemented in a custom-built program attached to an existing CAD software package. The program is called 'Solidfoam'. The program structure is presented in a Class diagram. Next, the program execution flow is described using charts. A more exhaustive set of flow charts is presented in appendices C and D.

4.1 Programming approach

It was decided to use an existing Windows®based CAD package to implement the RP tool. CAD software developers have been realizing for a long time that customers need customization to the package and therefore offer an API (Application Programming Interface) to tailor and extend the software as needed. Solid Edge®(SE) from UGS®and Solidworks®(SW) from 3DS®, two affordable mid-range CAD packages, were candidates. Solid Edge is used by 3EL-Company and Solidworks by the University of Twente. Both use the same Parasolid geometry modeling kernel. Although Solid Edge was preferred as it is the primary CAD package at 3EL-Company, Solidworks was eventually chosen for implementation, using the arguments in table 4.1.

feature or facility	**SW**	**SE**
no. of spans (using API)	> 20	3 (max)
3D drawing possible	yes	no
record macro's	yes	no
API user basis	small	big

Table 4.1: Differences between Solidworks and Solid Edge which affect implementation

The program is designed using an object-oriented approach. The language chosen to implement the slicing application is C# (pronounced as 'C sharp'). Main reasons are that it is fully object-oriented, having the possibilities comparable to C++ and the ease of use from Visual Basic. The development environment is free of charge. Disadvantages are its limited popularity in the CAD field and fewer available geometry-related libraries than for C++, along with slightly reduced speed and sometimes difficult interoperability with legacy COM components. The chosen system architecture is depicted in figure 4.1. It is a rather conventional architecture. The proxy is implemented as a class, working as an interface for all external API-calls. It also wraps API functions by more intuitive or more suitable defined functions. Note that the geometric kernel cannot be approached directly but through a restrictive API. Alternative architectures with different operating system or CAD-system architectures could be more suitable for research purposes, but this discussion is omitted here due to the practical context in which the project was set up.

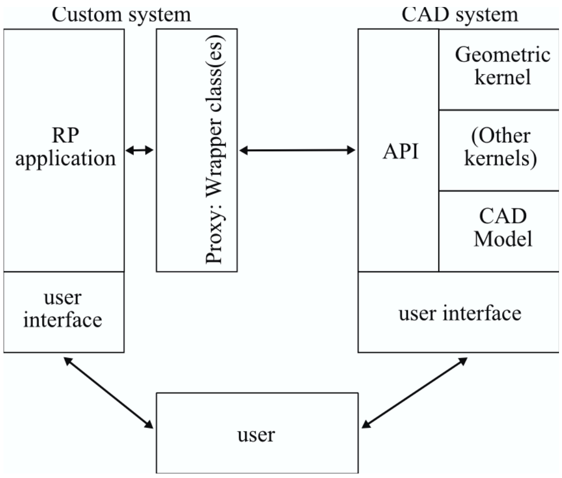


Figure 4.1: Conventional software architecture

Note that here, no graphical user interface was developed. Parameters can be changed in the source code only, which, together with several other restrictions, leaves the application in a prototype phase. All operations are performed in the Solidworks part environment using a single part.

4.2 Application Class diagram

The program data structure can be represented by a class diagram. It represents reality in a relatively intuitive manner, as many classes can be compared to what they represent physically, geometrically or mathematically. Figure 4.2 shows the simplified diagram containing the most important classes, ignoring functions and less relevant attributes. The left column of classes represent Solidworks classes that are referenced by Solidfoam. The diagram is commented in a top-down manner. Starting with the ‘SlicingManager’ class which controls the slicing process, it is shown that it has (a reference to a collection of) RPLayer-objects. The prefix RP denotes association with the RP process. LM (Layered Manufacturing) denotes the namespace in which the types are defined.

An RPLayer object or ‘layer’ references a base- and top RPSection object or simply ‘section’. Each section is shared by its lower and upper layer, except, of course, for the bottom- and topmost section. A layer also contains a reference to a surface loft feature. The ruled surface is represented by this feature, which

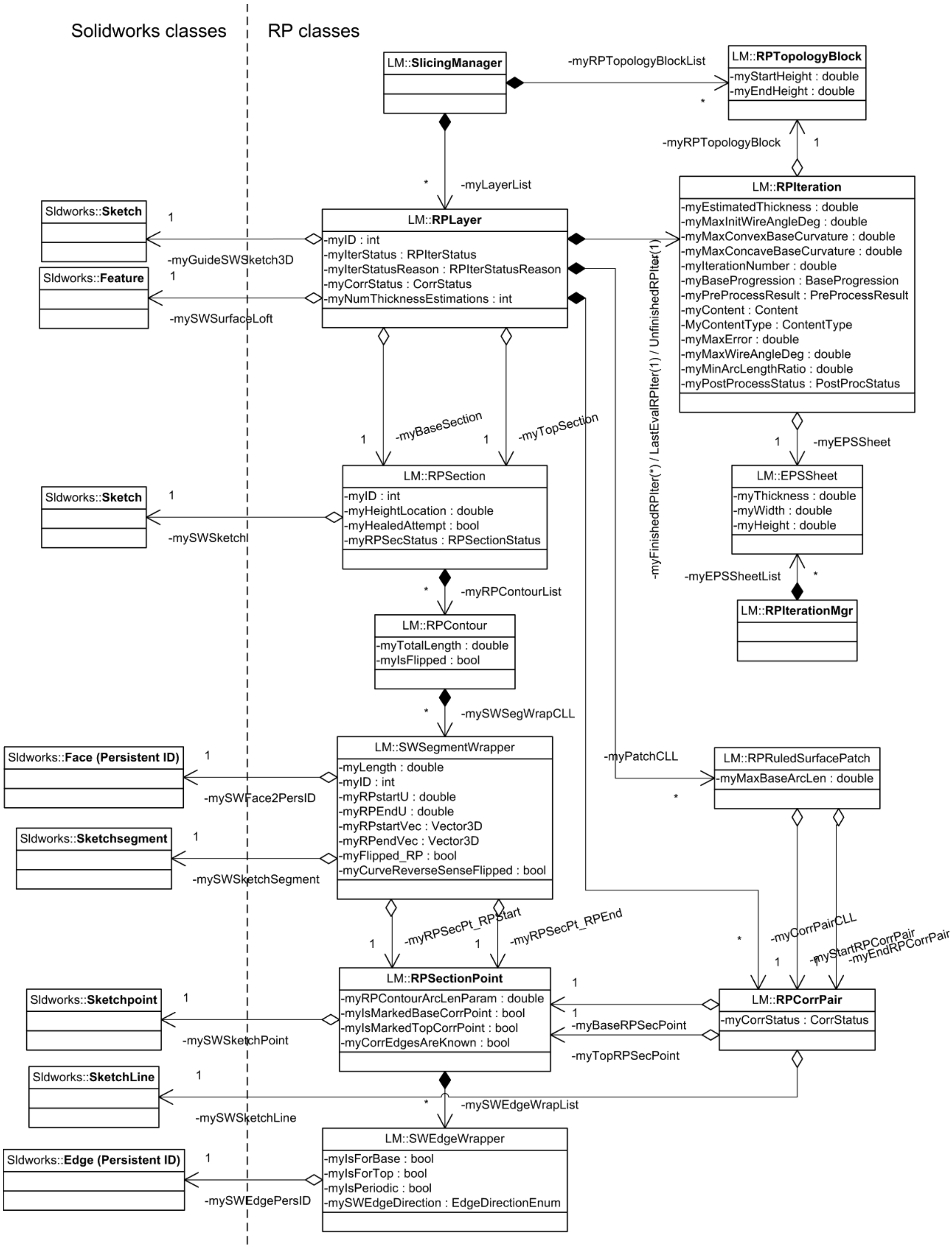


Figure 4.2: Solidfoam class diagram

comes standard with most CAD packages. This prevents the need to define the parametric ruled surface definition from scratch. It is assumed that Solidworks uses arclength parameterization, based on observations.

A set of two corresponding points, also known as a 'span', is represented by the RPCorrPair class. The graphical and geometry-topological equivalent of a span is represented by a 3D line segment, which is part of the guide-curve 3D-sketch as referenced by a layer. All correspondence information regarding location of spans is contained in this 3D sketch. Each RPCorrPair object has a reference to its 3D line segment.

A layer also has a list of ruled surface patches, which in turn contain no more than two references to their start- and end spans. The layer also has a list of all of its spans, an ID to identify it uniquely, and some enumerations about its status in the iteration process and correspondence. It also has a reference to (the list of) iterations is was built with.

Each RPSection has a reference to a SW sketch, each containing one closed contour, with an option for multiple contours when extending the application to include branching. The sketch contains the intersection from the sketch plane with the nominal CAD model. It serves as provider for both the underlying intersection curves and visual representation while not altering the CAD topology.

A section contour must be oriented to conform the 'right hand rule' (RHR) w.r.t. the $\vec{e}_z$. This explains the presence of a boolean indicating whether it is flipped or not. It also contains a sorted circular linked list of segment wrappers it is made out of.

The segment wrapper class contains a reference to the actual sketch segment in SW and the nominal CAD face it was derived from by the intersection. Furthermore, it contains information about the segment like its start- and end parameters, orientation w.r.t. its underlying curve and more. It also has references to its start- and end RPSectionPoints. This could later be extended by intermediate points.

The RPSectionPoint wraps a Sketchpoint in SW which is always present at the intersection of the RPSection sketch plane and a CAD model edge or vertex. It therefore has a list of references to the edges it intersects. The case of multiple edges can occur when the section sketch plane intersects a CAD vertex.

The SWEdgeWrapper class references a CAD model edge. An object of this class is instantiated at each edge intersection, so a single CAD model edge can be referenced by several wrapper objects, each corresponding to a specific RPSection-point. The wrapper also knows if the edge intersection applies to the layer below or above. Usually, it will apply to both. The direction property denotes the way the edge is intersected: tangent to the section plane y/n and in build direction y/n (four cases).

The RPIteration references an EPSSheet with a certain thickness. An iteration contains pre process and post process information. Processing, in short, denotes the top section shifting, correspondence solving and subsequent ruled surface generation. Pre process information is primarily based on information taken from the base section contour C_b of the nominal CAD model. This contains estimated thickness, estimated maximum wire angle, maximum curvatures and progression. The latter can indicate something about the change in CAD surface steepness or wire angle w.r.t build direction. Post process information contains the largest actual error e^*, largest actual wire angle θ^* after tool path generation and minimum arclength ratio $Æ^*$ to indicate bad surface quality. The status information facilitates the iteration process.

The RPTopologyBlock class can be used to subdivide a part into sections with constant topology in build direction using end feature detection. However, this has not been implemented further. It might make more sense to reference a block from the RPSection, but (top) RPSection objects are created and destroyed during iteration, whereas RPIteration objects are persistent.

4.3 Application workflow

The main program flow is shown in figure 4.3. The meaning of the shapes is explained in appendix C, along with more detailed charts concerning the key features of the program. More detailed implementation details are left out. Solidworks runs with an opened part that is to be sliced. For now, a plane must be user-defined as the start plane, using the plane normal as build direction with the option to flip it in case the normal points in the wrong direction. This plane is used for the base section of the first layer. Next, layers are sequentially created until the part is finished. That occurs when the program terminates due to an exception or artificial halt. Finally, the results are presented in a log file containing the iteration steps and a DXF file containing each contour in its own layer. Polylines are used to approximate the contours with a chordal error at least an order smaller than minimum achievable error at physical production. The meshing facilities in SW are used to achieve this.

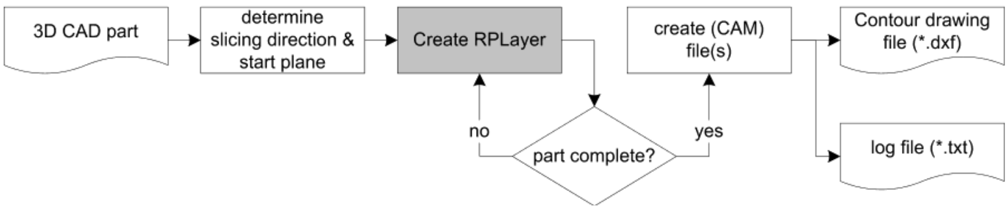


Figure 4.3: Main application flow

4.3.1 Iteration scheme

The iteration flow chart for a new RPLayer object creation is depicted in figure 4.4. Its black box processes are further shown and treated in Appendix C. In each iteration, a pre process phase precedes the processing phase ('RPLayer update'). The pre processing phase contains algorithms for calculation of a new layer thickness as discussed in section 3.5.6. It also has the means of controlling the further execution of the iteration itself. The 'update RPLayer' processing contains the shifting of the top RPSection height and new correspondence solving. The processing phase is followed by a post processing phase, containing the actual error evaluation as discussed in section 3.5.4, wire angle evaluation using the relations from section 3.5.3 and arc length ratio evaluation as presented in section 3.5.5, all corresponding to the current RPIteration object. This phase has influence on the iteration process itself as well.

4.3.2 Convergence

The iterative algorithm for finding the optimal layer thickness according to section 3.5.6 is shown in action in figure 4.5. The following approach is used. In the first iteration, the κ_n-based critical thickness t^* is determined and is denoted by a cross. Next, the closest smaller available DB sheet thickness is picked for establishing the top section height according to equation 3.10. Finally, the layer is processed and post processed.

The second iteration repeats the first iteration with an improved direction vector for t^*-deterination, leading to the second DB thickness. In the third iteration onwards, unit DB thickness increments are used, as expressed by equation 3.11.

During the iteration process, search space gets smaller, based on the change in layer thickness. This is depicted by shading. The process is brought to a halt when oscillation is detected or when a T_{DB}-limit is encountered. When either is the case, the optimal, that is, the thickest *valid* RPIteration object is searched for in the current RPLayer's RPIteration history and chosen as solution when available. The values of this optimum are copied to a new RPIteration object which is finally processed at the layer, thereby finishing a single layer. When no optimum is found, the part is simply defined not feasible for the HWC process with the current set of constraints. Thus, the iteration will either converge to a state where an optimum is found or to a state where the part is considered not feasible. Hope et al. [15] use a scheme which is comparable.

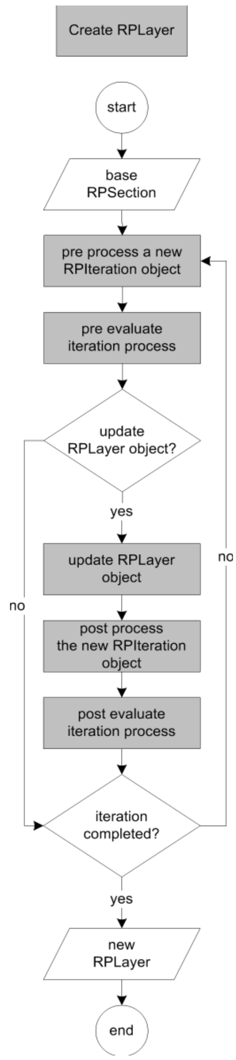


Figure 4.4: Layer creation iteration scheme

4.3.3 Marking points for correspondence

When an intersection curve is created, the curve sketch points are automatically marked for correspondence based on the local sharpness. Figure 4.6 shows an example. The user can decide whether tangent points should be used for correspondence solving. For sharp points, the program can be provided with a minimum angle α which the two connected curves at that point must at least have in order to be marked as a point for correspondence solving. Apart from the sketch points, no other points are introduced for correspondence yet. The marked points are then each wrapped an a RPSectionPoint object.

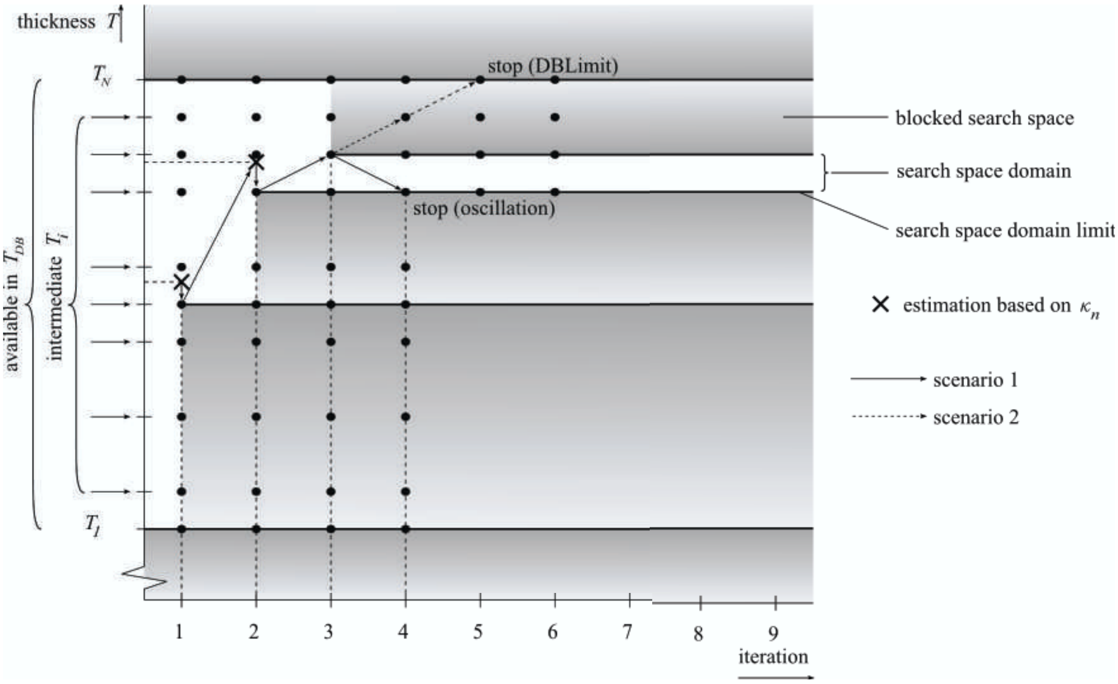


Figure 4.5: Layer thickness determination examples: two scenarios in the optimal thickness search algorithm

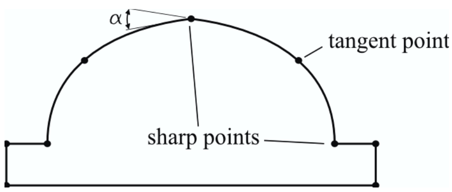


Figure 4.6: CAD model section

4.3.4 Correspondence solving

Each sketch point on an intersection curve was created by an nominal CAD model edge that was intersected by the section plane. After being marked for correspondence, it receives a reference to its edge. Next, topology traversal can start. The CAD model edges are traversed from base to top section, starting at the intersected base edge, in order to define a correspondence solution. A practical implementation is further proposed in appendix E. An obtained correspondence solution for the entire layer may be unacceptable. A check is first made to see whether it is fully defined. That is, every marked base and top section point must have a one-to-one correspondence with a section point on the opposite section. Theoretically, a fully defined solution does not guarantee an uncrossed solution. That is, solutions where the tool path arc length velocity changes sign at one of the sections. So a second check is performed to check for crossed spans. If the correspondence is found to fail a check, the program halts. User interaction could solve the errors but this interaction possibility has not been implemented.

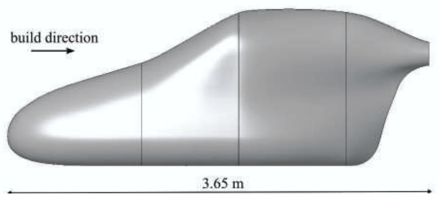
Chapter 5

Experimental results

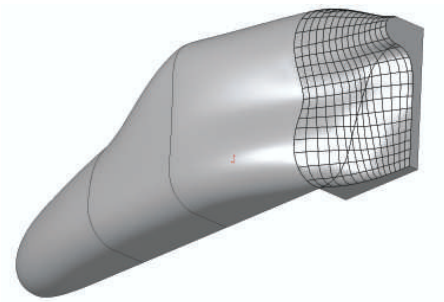
This chapter describes how the slicing process is employed from start to end, using an experimental aircraft body model as demonstrator. In many literature cases, axisymmetric geometries are used for validation of code. This has been done as well for debugging purposes but has been omitted here. In the following, the body represents a more general CAD model shape.

5.1 Computational slicing

To test the slicing program, a fuselage CAD model of an experimental aircraft called the 'Skymobil', as shown in figure 5.1, is computationally sliced. A surface extension was added at the tail in order to avoid solutions with extreme wire angles, as shown in figure 5.1(b). Although only one half of the body was modeled,



(a) Exact CAD model



(b) Extended surface

Figure 5.1: One half of the Skymobil body as nominal CAD model. The aircraft nose is located at the left. The conical shape on the right is suited for assembly of a push-propeller.

the extended surface causes a non-symmetric model. The excess material is cut off after physical manufacturing, leaving exactly one half of the aircraft body as shown in figure 5.1(a). The model contains no high-frequency surface information, small features or need for complex correspondence solutions. The guidance notches for aligning the layers in the physical assembly stage were not incorporated in the CAD model while computationally slicing, reducing complexity. As is shown in figure 5.1, an additional edge has been added by projection of a straight line onto the surface near the tail. This has been done in order to 'artificially' improve the local correspondence solution.

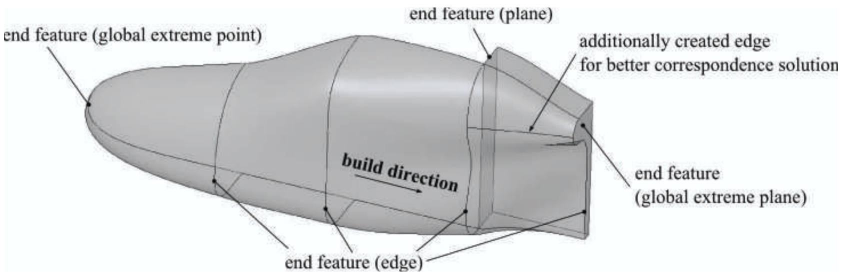


Figure 5.2: body with additional curve for correspondence solving

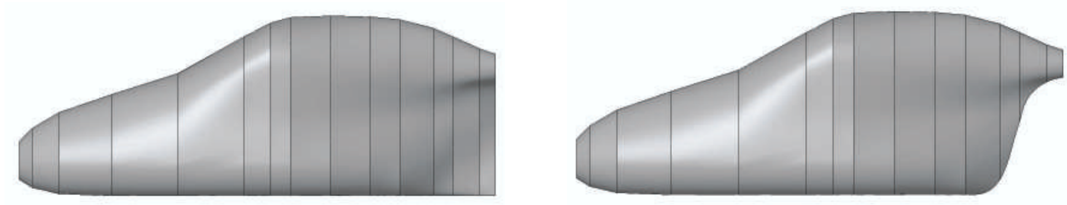
The slicing process has been executed using the following properties and constraints:

- $\delta = 10$ mm, expected to keep the amount of layers manageable,
- $\theta_{max} = 55\,°$,
- $Æ_{min} = 0.5$,
- T_{DB} ranges from 50 to 800 mm with 50 mm increments,
- sampling rate: 50 samples (homogeneously distributed over C_b),
- CAD model scaling for slicing: full scale,
- direction of $\vec{e}_z$: along the roll axis, pointing in air flow direction,
- start plane location: just behind the nose at a user-defined location,
- amount of section planes for calculation of actual error e: 3.

The sampling set is acceptable in terms of computational time, yet expected fine enough to yield a proper estimation. For evaluation e, $Æ$ and θ, every sample point is processed. That is, the check does not halt when exceeding constraint limits. This is computationally more expensive, yet it could give more insight. The chosen θ_{max} and $Æ_{min}$ may seem quite non-conservative, but allow the process to continue without interruption. The final results are manually checked for exceeding values.

5.2 Computational slicing results

The approximating set of ruled surfaces which is obtained after slicing is shown in figures 5.3 and 5.4. The entire slicing task was spit up using three segments to accommodate end features manually because those have not yet been taken into account in implementation. The iteration traversal for each individual layer is shown in figure 5.5. The corresponding iteration process log files for each layer are presented in appendix F. At the end of segments 2 and 3, custom layer thicknesses have been used to preserve the end features. At the end of segment 2, the slicing has been performed in reversed build direction to check if the custom thickness would not violate the imposed constraints.



(a) Resulting ruled surfaces

(b) Ruled CAD model with surface extend chopped off

Figure 5.3: Skymobil body as ruled CAD model

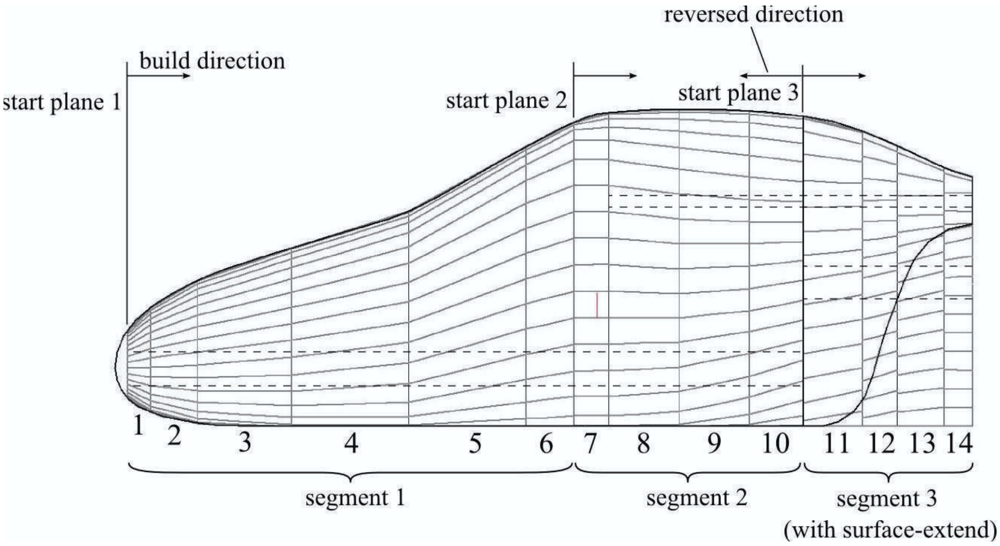


Figure 5.4: Skymobil body including layer numbers

As can be seen from figure 5.5, the final layer thickness often strongly deviates from the initial κ_n-based thickness estimation, as in layer 5. This can be easily explained by the fact that the base section of layer 5 is located in an area of high local κ_n-value. However, it quickly diminishes to a much lower value in build direction, rendering the κ_n-based thickness estimations too conservative. Here, further iteration continues by successfully picking larger thicknesses, instead of taking an acceptable yet too small thickness while assuming that the iteration process has come to an optimal solution.

The second item of interest is the effect of the improved (second) κ_n-based thickness estimation which uses the previous correspondence solution. Layers 6 and 12 are the only instances where the improved estimation changes by a large amount, but only at layer 6 leading to a larger than unit database thickness increment. Examining the geometry locally at the base section of layer 6, the thickness underestimation could be explained by skew cylinder effect from figure 3.4.

The critical wire angle θ^* remains amply below $45\,^\circ$ in all layers, except for layer 1 with wire positions near $45\,^\circ$, which is at the limit of feasibility. The critical arc length ratio Æ^* is another point of interest. The ratio at layers 11 through 13

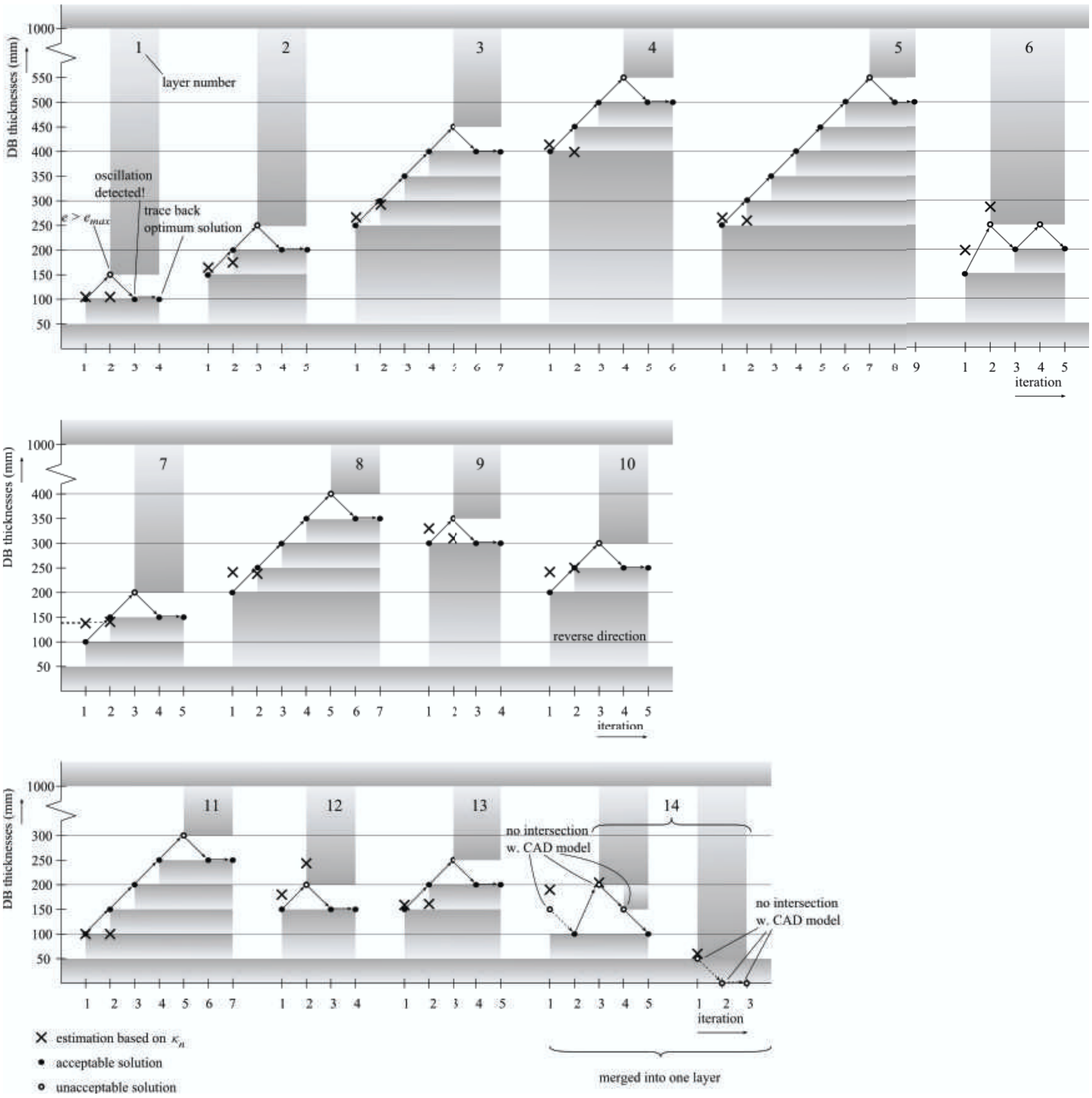


Figure 5.5: Graphical representation of iteration processes

is quite low: down to 50%, but this occurs at the bottom of the part which is chopped off later on, thus no cause for concern. The other instance is layer 1 with Æ = 0.51%. This might cause the cusp height to be exceeded.

The resulting values for e^* are shown in figure 5.6. All final actual errors are below the imposed threshold of 10 mm. At the last layer, one of the iterations yielded no intersection with the CAD model, represented by an infinite error. Layer 6 has an infinite error which was caused by the fact that the e^* exceeded half the length of the line segment along which the actual error was evaluated. The Segment is kept short enough to avoid intersection of non-relevant ruled surface patches in the ruled surfaces set.

In order to check if the sampling rate was fine enough in circumferential and build direction, an additional audit was made. Both the nominal (figure 5.1(a)) and ruled (figure 5.3(b)) CAD model were tessellated using maximum chordal errors of 0.01 mm, aligned with each other and compared by ‘shortest distance’

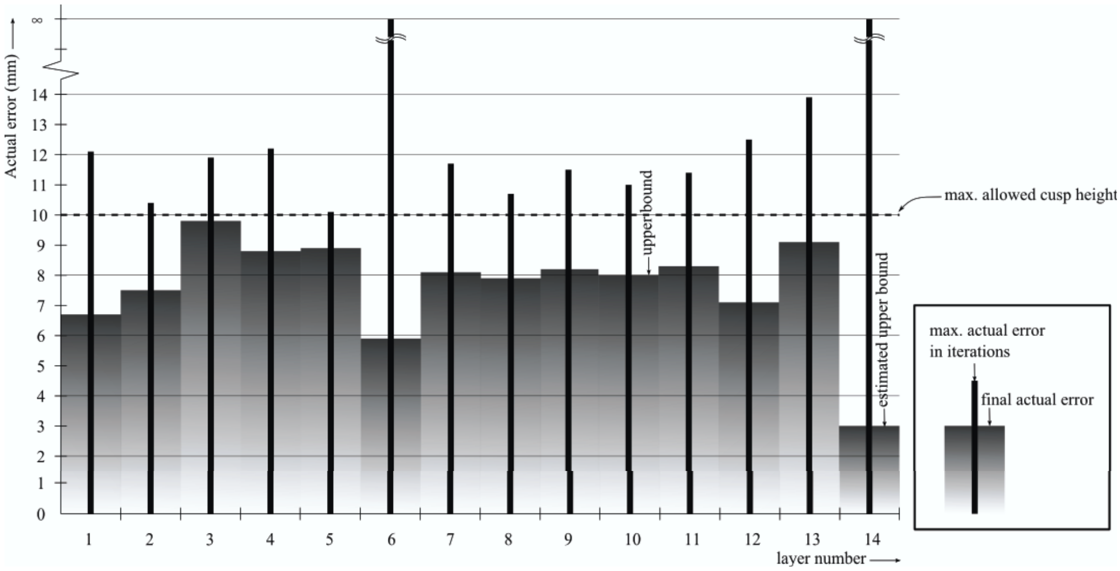


Figure 5.6: Graphical representation of actual error

of each node on the tessellated ruled model to the nominal model. Results are shown graphically in figure 5.7. The deviations remain within the imposed limit of 10 mm, which indicates that for this specific case, the sampling rates seem to be good. Note that no interval of confidence has been established for this observation. This is considered justified by the fact that, in the worst case, the error in normal deviation is in the order of the tessellation tolerance, which is an order smaller than the errors of interest.

Finally, it must be remarked that the processing time is very long. Calculation times in the order of 30 minutes per layer are not rare. The estimated average duration for a single iteration lies in the order of 5 minutes as can be derived from appendix F. Per κ_n-calculation or error-evaluation sample, approximate calculation times in the order of 0.3 seconds are observed. The soft- and hardware used here are presented in table 5.1.

CPU	Intel Core2 Duo T7500 (2x 2.2 GHz)
memory (RAM)	2 GB
operating system	MS Windows XP Professional SP 2
.NET framework version	2.0.50727
Solidworks version	2006 SP 4.1
Solidfoam version	1.1.6.3

Table 5.1: Used soft- and hardware

5.3 Physical slicing

After DXF export of the section contours from the slicing software as in figure 5.8, the file is transferred to the program controlling the machine. Next, the cor-

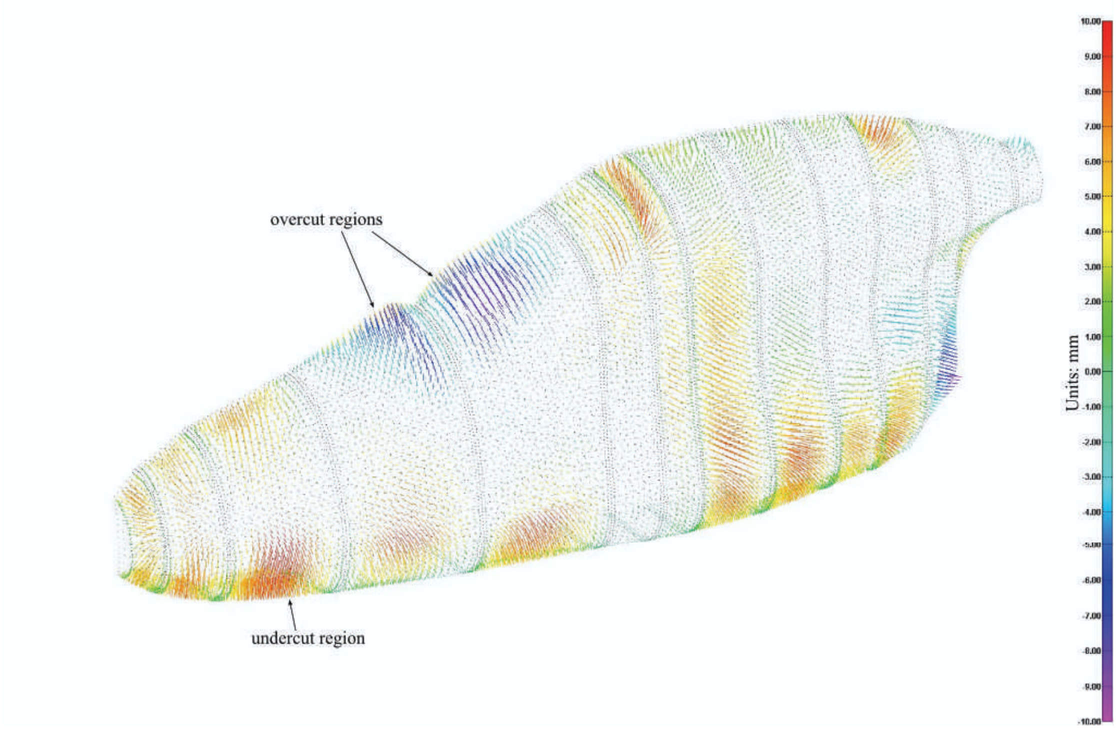


Figure 5.7: Post-slicing audit evaluation of normal deviations. The maximum absolute normal deviation equals 9.6 mm. The deviation vectors are magnified by a factor 10 for better visibility

respondence is manually created again because it got lost during the data transfer through DXF files.

Further data for manufacturing include:

- scale factor: 0.5 to keep the production size more practical,
- kerf width: corrected by step four software,
- Material: EPS 150, age > 6 months and assumed free of after-shrinkage,
- full calibration of machine.

To reduce the risk of foam part misalignment in the machine, the ruled surface and the planar base section surface were cut in the same setup, avoiding movement of the part between these cutting phases. The planar base section surface is cut first with a vertical wire, followed by the contours with the horizontal wire as shown in figure 1.1. The vertical wire is aligned perpendicularly to the horizontal wire with the latter in its horizontal orientation. The top section was cut back to its final thickness by the horizontal wire in a second setup.

Layer number 1, located at the nose section, is not produced because once scaled, the software was not able to correctly generate the carriage tool paths. The resulting layers are shown in figure 5.9. After Assembly and cutting off the tail extension, the result is as shown in figure 5.10. The most notable is the fact that the layers did not have a tight fit with respect to each other. At each transition from layer to layer, discontinuities in the order of 2 mm were present. The cause for this phenomenon is not yet known. In the next section, accuracy is analyzed further.

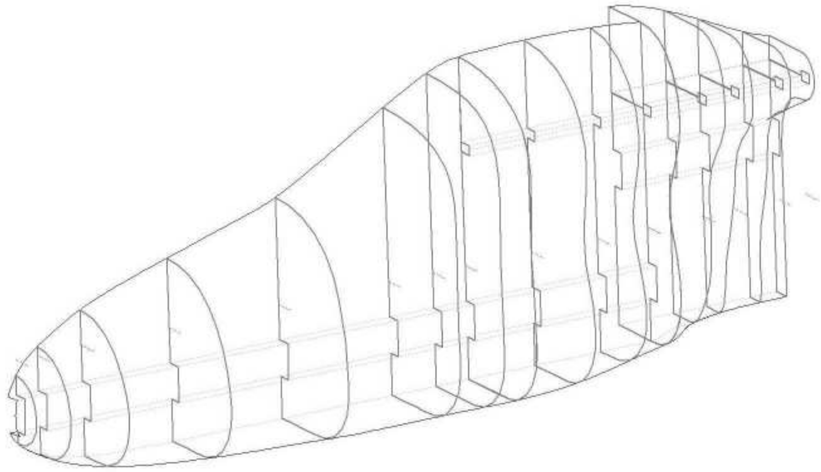


Figure 5.8: 2D sketches as exported to DXF. The sketches include geometry for alignment of the layers after production

(a) (b)

Figure 5.9: Layers after manufacturing

5.4 Accuracy analysis

The Step Four HWC machine has a claimed contour tool path accuracy of 0.2 mm/m [36]. However, the conditions for this number are not given so it assumed to be a general, low wire angle, high arc length ratio number. This is typically the case when cutting 2D objects. However, in this project, generating 2.5D layers, this is generally not the case at all. In order to get a quantitative measure of process accuracy, the demonstrator as described in the foregoing was measured and the measurement data have been compared to its reference ruled-surface CAD model. Also, the precision and accuracy of the measurement method are evaluated. Appendix G contains all information about the measurement method, setup and the actual measurements. The results are summarized in table 5.2. These contour cutting accuracy figures represent a measure the normal deviation of 'as manufactured' surfaces with respect to the 'as designed' *ruled surface approximations.* Care

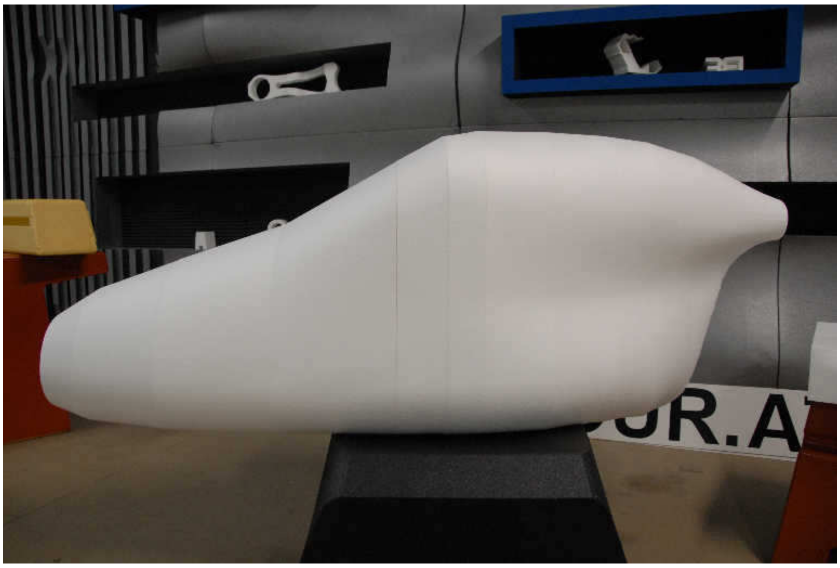


Figure 5.10: Assembled Skymobil body

Accuracy indications	
	units: mm/m
2D contour cutting accuracy (RMS)	0.3
layered object - contour accuracy (RMS)	1.2
layered object - accuracy (σ)	1.1
layered object - layer thickness accuracy (glued)	0.1
	units: mm
layered object - tolerance (gross errors included)	8.8
layered object - tolerance (gross errors excluded)	5.8

Table 5.2: Accuracy summary

should be taken when using these values because the number and extent of the experiments is very limited. It merely offers a coarse quantitative rule of thumb. The 2D precision seems in the same order as presented by the manufacturer.

Figure 5.11 shows the maximum *physically measured* normal deviations per layer from the *nominal CAD model* together with the *expected* normal deviations for each layer. The physical measurement sample points are the same as used for obtaining the layered assembly accuracy data in table 5.2. The expected values, as obtained from computational slicing, are the same as depicted in figure 5.6, scaled by a factor of 0.5 as was done with the CAD model before production. A trend seems not to be present. Most layers remain within the tolerance. Three layers have normal deviations exceeding the user-defined limit, which can lead to rejection of the product in commercial cases. The large error in layer 11 can be explained by a gross human error as explained further in appendix G. This layer has deficient manually applied correspondence. The difference between measured and calculated is in the order of 1 mm.

The precision values are a limited representation of quality. More information

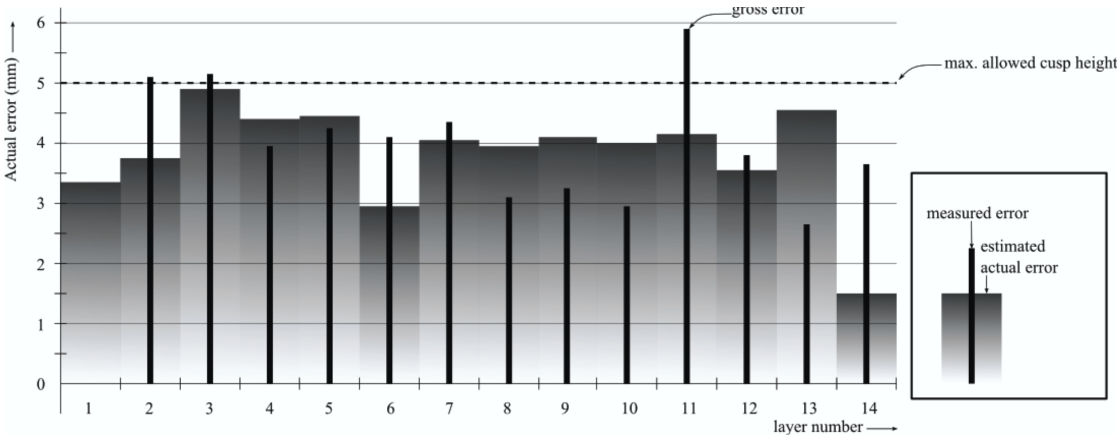


Figure 5.11: Chart with the maximum *measured* and the maximum *calculated* normal deviations per layer.

can be used from even more measurements to obtain a feasible tolerance field value. For this purpose, the 'process feasibility' C_p is used often [17]. It is defined by

$$C_p = \frac{T}{6\sigma}, \tag{5.1}$$

where T is the allowed tolerance field and σ the standard deviation of the measured error at *many* layered products. The fraction denotes the ratio of the tolerance field versus (a de facto chosen) $\pm 3\sigma$. The latter contains 99.73% of the measurements. In practice, $C_p \leq 1.33$ is used to call the process 'statistically controlled'. When $1 \leq C_p \leq 1.33$, a large risk of failure exists. Using this for a statement about the process feasability comprising 'as manufactured' versus the 'as designed' ruled surfaces, the assumption of σ to be in the order of 1 mm can be proposed while choosing $C_p = 1.33$, yielding T in the order of 8 mm.

Chapter 6

Discussion

This chapter discusses whether the master project objective is met properly, mentioning the most significant drawbacks, limitations, together with comment on manufacture accuracy. This concerns tool development and the demonstrator. Other HWC-process related issues are not treated here.

6.1 Thickness estimation and iteration

In the iterations, decisions are made regarding the direction of thickness change. However, no information is given about the *magnitude* of the change. Information from the region *between* the base and top section can be used to this extent. The trend, for example, of the actual error can be extrapolated after the second iteration to estimate a new thickness. This was omitted, sometimes resulting in a lengthy iteration sequence.

The user-defined increment(s) between the thickness values in the thickness database or -set and its set size also strongly influence the convergence speed of the iteration process. Unit thickness changes are used in the iteration process because the initial κ_n-based thickness estimations are assumed to yield a solution in the vicinity of the optimum. In general, this might not be the case. Together with a large set of thicknesses, this could lead to a lengthy iteration sequence. However, the practical fact of using a very limited amount of thicknesses relieves this issue to some extent. Binary search algorithms or other methods could be faster in the case where the assumption of the near-optimum κ_n-based thickness estimation is dropped, although this still depends of the thickness database size.

The κ_n-based thickness estimations are calculated with brute force. This could be improved as illustrated by the following. Take a sample point X_i on C_b during the first two iterations of a layer as shown in figure 6.1. The tangent plane on the nominal surface at X_i is denoted by T_i. The following takes place at sample point X_i:

Iteration 1:

- substitution $\vec{d_1} = \vec{e}_z$ is performed,
- Suppose $\vec{t_1}$ as the normalized projection of $\vec{d_1}$ on T_i,
- κ_n is evaluated corresponding to the direction of $\vec{t_1}$.

Iteration 2:
- substitution $\vec{d_2} = \vec{c}$ is performed,
- Suppose $\vec{t_2}$ as the normalized projection of $\vec{d_2}$ on T_i,
- κ_n is evaluated corresponding to the direction of $\vec{t_2}$.

Define ϕ as the angle between $\vec{t_1}$ and $\vec{t_2}$. In many and probably most cases, this angle is not changed by a significant amount. In order to save unnecessary calculations, ϕ can be calculated in advance. In other words, C_b is checked for skew cylinder effects first. Provided that ϕ exceeds a specific threshold, κ_n and the new thickness estimation are allowed to be recalculated at X_i. However, the rate of change of κ_n with the direction of $\vec{t}$ is yet unknown, possibly leading to a difficult choice for the threshold value.

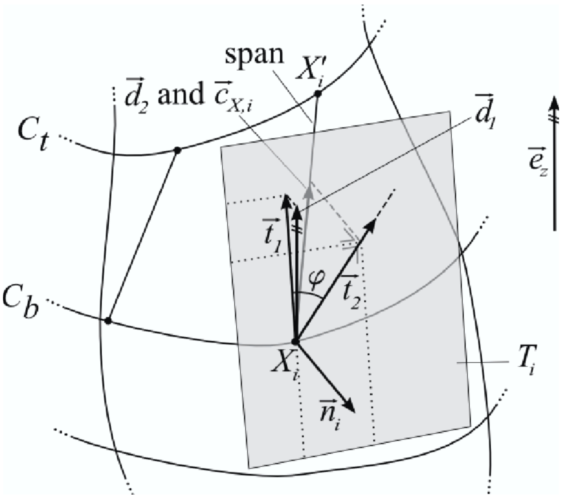


Figure 6.1: Change in direction for κ_n-calculation in the tangent plane

Apart from the initial κ_n-based thickness estimations, other remarks must be made regarding determination of layer thickness. Assumption 3.12, with t_L as the layer thickness of L, repeated here as

$$\frac{\partial e^*}{\partial t_L} > 0,$$

states that the critical actual error increases with increasing layer thickness, which does not hold in general. However, this has not been taken into account. A local minimum satisfying $e^* \leq \delta$ may exist at $t > 0$, but are not likely for the relatively 'simple' shapes as shown in 3.1.

An analogue discussion could be held for assumption 3.13, repeated here by

$$\frac{\partial \AE^*}{\partial t_L} < 0,$$

stating that the critical arc length ratio decreases with increasing layer thickness. Here, a local maximum satisfying $\AE^* > \AE_{min}$ may exist for $t > 0$. However, this is assumed very unlikely.

The calculation of equation 3.14, expressed by

$$\frac{\partial \theta^*}{\partial t_L}, \tag{6.1}$$

results in information about the change in wire angle for the critical wire angle θ^* only. This information is then used in expression 3.11 to determine in which direction the layer thickness is adjusted. However, there may be locations other than the critical one where equation 3.14 has the opposite sign and possible a larger magnitude. This information is ignored, which, in general, might lead to a thickness change in the wrong direction.

The last comment here is the initial condition for the iteration process, defined by the location of the start plane. It is user defined here, but automation could yield a more optimal location. However, this is expected to increase computational cost considerably.

6.2 Direct slicing

The choice for direct slicing avoids the need for the development of intersection algorithms. However, in many cases, intersections of the nominal CAD model coinciding with model edges often resulted in geometrical errors. At intersections in the vicinity of these edges, that is, even several orders in magnitude larger than the machine accuracy ϵ, this was often the case as well. As a solution, the intersection plane can be offset from the error location by a large enough distance. This avoids many edge intersection cases as presented in appendix E.

Another point of interest are the resulting intersection curves which result from plane-surface intersections. In some cases, the resulting intersection curve was self-intersecting although this is not visible. Bad parameterization in the geometry kernel could be a cause, but this has not been investigated. The problem is solved best by approximating the intersection curve with a fitted spline, using a tolerance of an order smaller than δ.

The implementation according to the scheme in 4.1 works for an application in prototype phase, but the speed of the total system is very low. This is reflected, for example, by the process times shown in appendix F. API-calls might trigger processes in the CAD program which causes unknown overhead. More direct access to the kernel might offer improved speed.

6.3 Correspondence solving

A more detailed local optimized correspondence solution like Zheng's spring model method from section 2.5.2 can yield thicker and less layers compared to the use of topology traversal alone. Topology traversal is a good solution to preserve features, but further local optimization can reduce the actual error e, especially for twisted and 'concave' geometries. Zheng uses a dense sampling for the creation of spans, although just a few spans might, combined with linear arc length interpolation, may reduce the error significantly already, allowing the use of thicker layers.

When extension of the tool is desired to take the spring model method or user interaction into account, the question arises whether or not the application design was set up properly in order to implement these. New insights can change part of it or even render certain approaches too cluttered or inadequate.

When the correspondence problem needs user interaction and multiple iterations are used to obtain an optimal layer thickness, it is very inconvenient to the user as for every iteration, manual correspondence solving is needed. This should be avoided, but poses new challenges.

6.4 Error estimation

Many measures of error estimation were presented in section 2.8. However, the de facto measure of error used in inspection software involves no more than the error:

- along a user-defined vector,
- in a user-defined plane,
- to the 'closest point'.

The latter calculates, for each point in a point cloud set, the distance from that point to the closest point on the nominal surface set. This option was used in the accuracy analysis. For simple geometry as in figure 6.2 on the left, the normal deviation equals the closest distance deviation. However, when the shape has, for example, inflection points, this equality does not hold. Therefore, it should be kept in mind that accuracy analysis results such as performed in this work should be treated with caution.

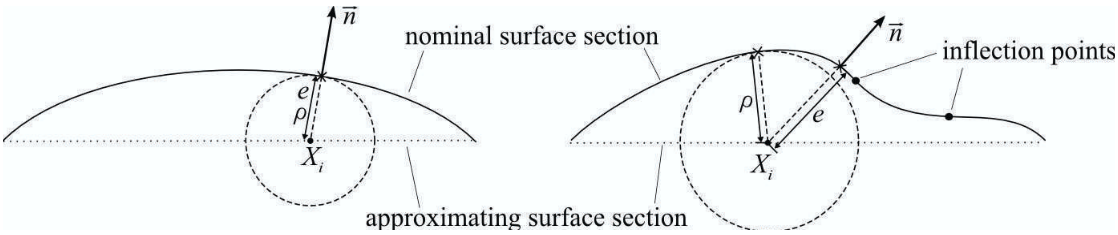


Figure 6.2: Normal deviation from nominal surface, denoted by e and 'closest point' deviation, denoted by ρ

Although different measures of error exist, κ_n-base layer thickness estimation used here requires the cusp height δ. This could require the need to express non-cusp height measures of error to be translated into an equivalent cusp height or a different kind of thickness estimation.

The implementation for error estimation evaluation comprised a brute force algorithm. For each sample, at least one line-surface intersection with a parametric surface is calculated. This approach can be replaced by a discrete variant, using the tessellated approximation of the nominal model, generated using a chordal error which is an order smaller than the maximum allowed error. This would involve significantly less API calls because the tessellation set of the model is queried once. As a result, the set of triangles can be accessed directly instead of through the API.

6.5 Physical fabrication

The accuracy of fabrication strongly depends on the machine construction, software and machine calibration. The calibration strongly depends on human factors and is not automated, which could cause unsuspected results when high accuracy is demanded. However, a separate investigation into the factors affecting accuracy has not been performed and is beyond the scope of this work. Kerf width correction is one of the most important of these.

6.6 Accuracy analysis

The accuracy of the slicing depends very much on the sampling density chosen by the user. It would be convenient if the rate was defined automatically.

The precision values as output from the photogrammetry software are not fully defined. The software is a black box and the output is assumed to be correct. For further insight and references regarding PG precision, the reader is referred to [27].

Based on the magnitude of the measured tolerance, it seems sensible to improve the accuracy of the HWC process in order to manufacture layered objects with high accuracy. As can be observed from figure 5.11, the measured error exceeds the maximum allowed error in the layer plane. However, alignment errors in the order of 1 mm could change these values by an amount in the order of 1 mm. An additional safety margin for the user-defined cusp height is therefore advised. However, this can lead to thinner layers, possibly increasing the number of adhesive interfaces and their corresponding error.

Accuracy analysis for single layered 2.5D shapes was not done. Information about the manufacturing process, including information about kerf width correction accuracy, can be obtained from such analysis.

Chapter 7

Conclusions and recommendations

This chapter summarizes the conclusions that can be drawn from the master project, followed by recommendations for further research.

7.1 Conclusions

The main problem as identified in chapter 1 describes technological disadvantages of HWC compared to milling by the lack of automation and accuracy control. A tool that computationally subdivides a CAD model into feasible layers, taking accuracy, process limitations, surface quality and the intent of the designer into account, does not yet exist for the HWC process in its current state. Such a tool was therefore proposed to be developed by combining and extending existing methods from RP literature. The following items involve conclusions that can be drawn from the master project.

- Program:

A prototype software tool was successfully developed and deployed, yielding satisfactory results for non-branching input models with relatively simple topology. It automatically calculates both the optimal layer thicknesses and the approximate tool paths by solving for correspondence. The user defines the maximum allowable cusp height δ, maximum wire angle θ_{max} and the minimum allowable arc length ratio $Æ_{min}$. A sampling of the actual normal deviation e was implemented as an example of error estimation where $|e| \leq \delta$ should hold. The actual wire angle θ is sampled to estimate if $\theta \leq \theta_{max}$ holds. The minimum arc length ratio $Æ$ is evaluated to ensure $Æ \geq Æ_{min}$. An iteration scheme for optimal thickness calculation was created, taking these constraints into account as much as possible. The program also indicates when a model is considered not feasible.

- Correspondence:

Direct slicing was chosen instead of tessellated slicing. It best suits the demand of feature preservation, using topology traversal to solve the correspondence problem. It is suited for relatively simple CAD models only, using the assumption that topology correctly represents correspondence. However, user interaction likely remains to be required in a variety of cases. The ruled surface approximation remains under constraint and therefore different solutions may exist. It can be proposed that in general, a fully automated correspondence solver will never guarantee to yield the intended solution, unless additional design rules are added or the allowed shape and complexity of input models are severely restricted in advance. In those cases, solutions are more likely to match the original intent. Solving and optionally optimizing the correspondence in each single iteration is feasible for simple shapes that can be handled by automation but may become unacceptable when user interaction for correspondence solving is required.

- Curvature-based thickness estimation:

Layer thickness optimization is computationally expensive in the current implementation. A good estimation based on κ_n can reduce the amount of further iterations. However, the added value of a second κ_n-based thickness estimation, using the correspondence from the previous iteration, is limited to regions where the evaluation direction of κ_n significantly changes. This depends on the combination of input model shape, geometric topology and build direction. Based on the results of the demonstrator, use of more than one estimation per layer does not seem justified.

- Intersection:

Intersection of a CAD model by a plane is difficult when intersecting through vertices or edges and should be avoided where possible. Intersection does not always yield a valid intersection contour. Checks are necessary in order to ensure a valid non-self-intersecting parameterization and approximated by new splines where possible, even when the nominal model seems perfect.

- Error estimation:

The method used for error estimation should be based on its purpose, but cusp height δ is best suited to give an initial estimate of the layer thickness. The normal deviation sampling used here for actual error estimation is computationally expensive. However, accuracy control is far more reliable compared to manual slicing, provided that it is used correctly. That is, the error sampling density should not be chosen too low. It should therefore be taken into account that the human factor has not left the process yet.

- Performance:

The iteration process takes an exceptional amount of calculation time. The chosen software architecture might cause considerable overhead, which may be justified in experimental or prototype phase, but not in future commercial exploitation. Although the limited amount of practical available layer thicknesses alleviates the problem, the cause remains and the amount of iterations should therefore still be kept as small as possible.

- Accuracy:

Based on a few measurements, the fabrication accuracy of a layered object now has an estimated accuracy in the order of 1 mm (RMS) in the layer plane and 0.1 mm per adhesive interface. The measured tolerance is in the order of 6 mm, excluding gross errors.

7.2 Recommendations

The following items involve the most relevant recommendations for further development.

- Layer thickness estimation:

Apart from the determination of the *direction of change* for a new layer thickness value, the *magnitude* of the change can be estimated as well, based on, for example, the trend of the actual error throughout the previous iterations. Other properties of the nominal surfaces could be used as well. Further analysis and research of these possibilities can be promising.

- Correspondence:

Another promising subject lies in the application of the spring model for improved local or global correspondence solving, possibly increasing the resulting layer thickness. Sharp edges can be retained as features when using topology traversal. However, when the edges are not available or when topology does not lead traversal to a good solution, the spring model might offer an interesting alternative. It is originally used at high resolution, but a few spans might be enough in many cases. However, the location of the spans may become a variable to be solved for. Linear arc length interpolation can still describe the correspondence between the spans.

After export to DXF, correspondence is lost after computationally slicing. Manual reconstruction introduces the chance of incorrectly applying correspondence the second time. Export to a suitable file format should avoid this and is recommended for implementation with high priority.

- Intersection:

Intersection of end features and other vertices and edges should be avoided automatically where possible. Automatically shifting the intersection location by a small distance could avoid intersection problems.

- Error estimation:

A promising gain in speed could be obtained from actual error evaluation using a *tessellated approximation* of both the nominal and ruled surface sets while still using direct slicing. A lot of literature exists regarding algorithms querying large mesh sets. A comparison with the current approach regarding computational time might yield large differences. Volume difference, based in some way on the value of δ, may also be used for a rough error estimation in the first iterations to speed up the process. However, this is a global measure of error, likely ignoring large local deviations and should therefore be treated with care.

- Performance:

The architecture-related influence on calculation time is another point of interest. Direct access to geometric kernels could improve performance, but this has not been investigated.

- Accuracy:

The kerf width correction has received little attention in this work, yet is essential for high accuracy of the product. The current kerf width correction models as used in the Step Four software are supposedly too limited for 2.5D layers with respect to accuracy. The accuracy of the HWC machine and software used should be investigated further using far more basic 2.5D ruled surface layers in order to further pinpoint the cause of dimensional deviations.

Appendix A

Layer thickness estimation

The following geometric considerations are used to obtain a layer thickness estimation t. It is based on a circular approximation of a planar curve in the neighborhood of a point, where the curve is the intersection of a plane and the CAD model surface. That point is depicted as point E in figure A.1. The thickness t equals the projected length of $\overrightarrow{EB}$ on the unit build direction vector $\vec{e}_z$. Point E is used, together with known normal vector $\vec{n}_s$ and the known unit surface tangent vector $\vec{t_c}$, to construct a local coordinate system, allowing the vector $\overrightarrow{EB}$ to be expressed in this CS. However, we need distances a and b in order to do this. Using three times the Pythagorean theorem at the geometry of figure A.1, three basic relations are obtained from which the distances between the points can be solved. The relations are the following:

$$\triangle ABC\colon \rho^2 = a^2 + (\rho - b)^2 \tag{A.1}$$

$$\triangle ABD\colon \rho^2 = \left(\frac{m}{2}\right)^2 + (\rho - \delta)^2 \tag{A.2}$$

$$\triangle BCE\colon m^2 = a^2 + b^2 \tag{A.3}$$

Knowns are radius of curvature ρ and maximum allowed error δ; Unknowns are distances a, b and m. Now, A.1 through A.3 can be solved for the unknowns. Note that δ represents a distance and NOT an angle.

$$A.1 \Leftrightarrow b = \begin{cases} \rho - \sqrt{\rho^2 - a^2} & \text{if } -\frac{\pi}{2} < \theta < \frac{\pi}{2} \\ \rho + \sqrt{\rho^2 - a^2} & \text{if } \frac{\pi}{2} \leq \theta \leq \frac{3}{2}\pi \end{cases} \tag{A.4}$$

$$A.2 \Leftrightarrow m^2 = 4\rho^2 - 4(\rho - \delta)^2 \tag{A.5}$$

Using A.5, we can easily derive that

$$t = 2\sqrt{2\rho\delta - \delta^2}\cos\theta, \tag{A.6}$$

which is frequently encountered in literature Kulkarni and Dutta [21], Banerjee et al. [2], Hope et al. [15], de Jager et al. [8], Kumar and Choudhury [22]. However, θ is still unknown, so this expression is not satisfactory. For equation A.4, it is useful to know when the $\theta = \frac{\pi}{2}$ transition is crossed. This is the case when $2\left(\frac{m}{2}\right)^2 = \rho^2$, as can be easily deducted from the figure, and hence $\delta = \left(1 - \frac{1}{\sqrt{2}}\right)\rho$.

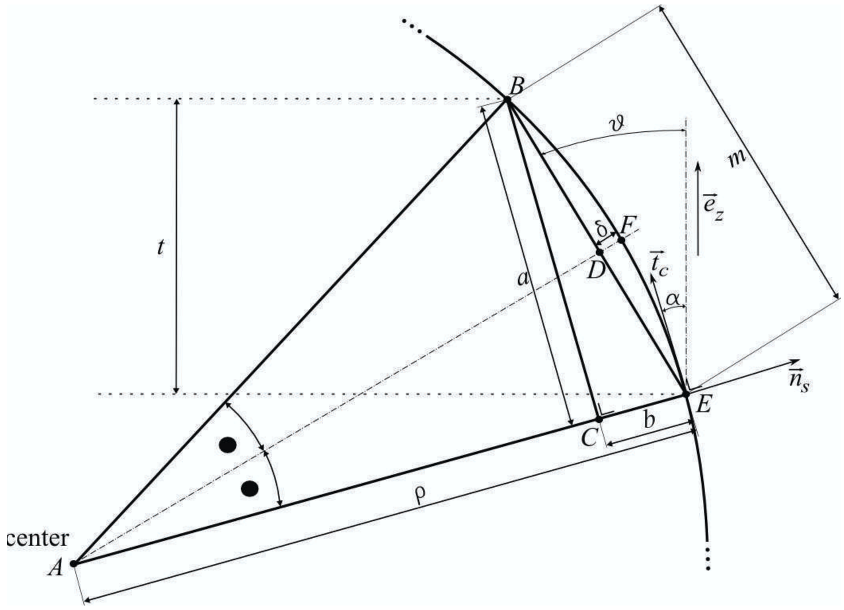


Figure A.1: Circular approximation of cross-sectional CAD-model geometry. Note that δ represents a distance and not an angle.

Substituting A.4 and A.5 in A.3, we get

$$
\begin{aligned}
4\rho^2 - 4(\rho-\delta)^2 &= a^2 + [\rho \pm \sqrt{\rho^2 - a^2}]^2 \Leftrightarrow \\
4\rho^2 - 4(\rho^2 - 2r\delta + \delta^2) &= a^2 + [\rho^2 \pm 2\rho\sqrt{\rho^2 - a^2} + (\rho^2 - a^2)] \Leftrightarrow \\
8\rho\delta - 4\delta^2 &= 2\rho^2 \pm 2\rho\sqrt{\rho^2 - a^2} \Leftrightarrow \\
a &= \left[\rho^2 - \left(\frac{8\rho\delta - 4\delta^2 - 2\rho^2}{-2\rho}\right)^2\right]^{\frac{1}{2}} \Leftrightarrow \\
a &= \left[\rho^2 - \left(\rho - 4\delta + \frac{2\delta^2}{\rho}\right)^2\right]^{\frac{1}{2}} \qquad \text{(A.7)}
\end{aligned}
$$

Now the term between braces is expanded:

$$
\begin{aligned}
\left(\rho - 4\delta + \frac{2\delta^2}{\rho}\right)^2 &= \rho^2 - 4\delta\rho + 2\delta^2 - 4\delta\rho + 16\delta^2 - 8\frac{\delta^3}{\rho} + 2\delta^2 - 8\frac{\delta^3}{\rho} + 4\frac{\delta^4}{\rho^2} \\
&= \rho^2 - 8\delta\rho + 20\delta^2 - 16\frac{\delta^3}{\rho} + 4\frac{\delta^4}{\rho^2}.
\end{aligned}
$$

Using this expanded expression, A.7 can be simplified to

$$
a = \left[8\delta\rho - 20\delta^2 + 16\frac{\delta^3}{\rho} - 4\frac{\delta^4}{\rho^2}\right]^{\frac{1}{2}} \qquad \text{(A.8)}
$$

Now, the layer height t can be calculated using this very basic expression. Using point E together with the known CAD surface normal vector and curve tangent

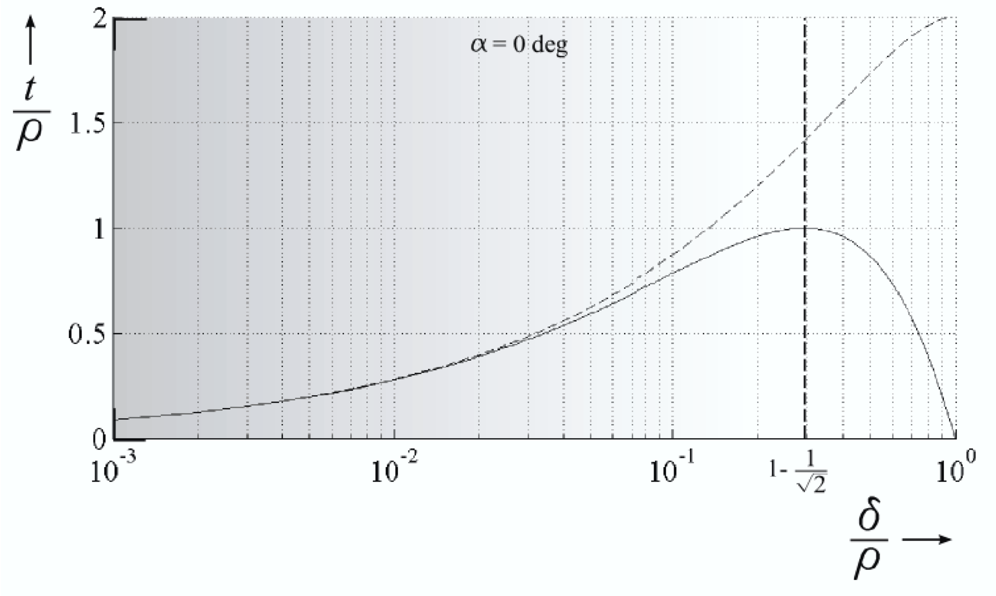


Figure A.2: Equation A.6 (dashed) compared to equation A.10 (solid).

vector to construct a local coordinate system, vector $\overrightarrow{EB}$ can be expressed in this CS:

$$\overrightarrow{EB} = -b\vec{n_s} + a\vec{t_c}, \tag{A.9}$$

Now a layer thickness estimation t is obtained:

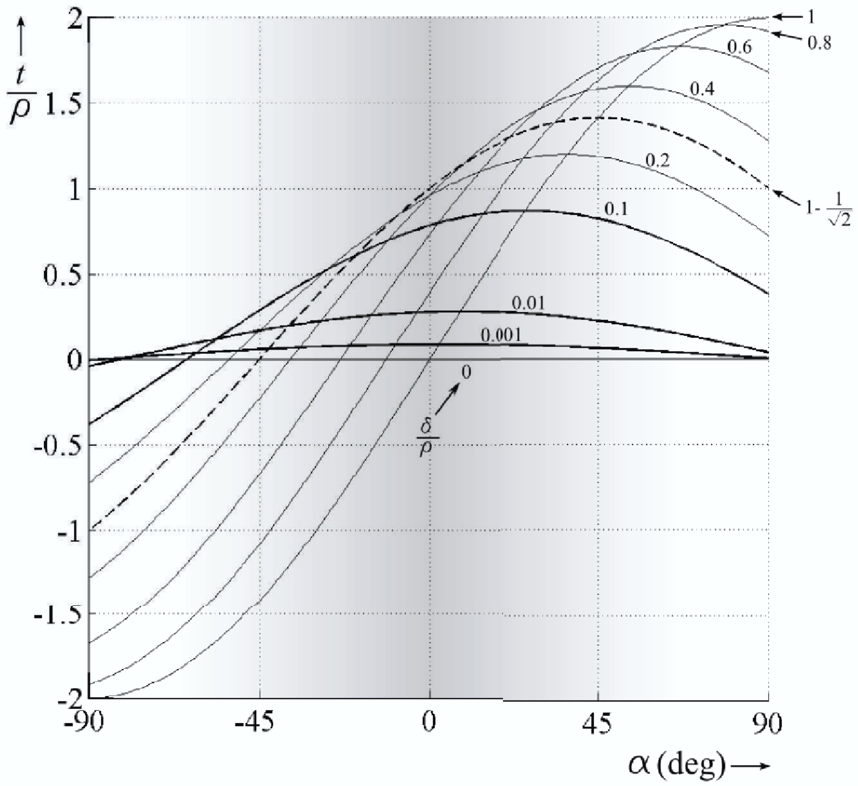
$$\begin{aligned} t &= \overrightarrow{EB}\cdot\vec{e_z} \Leftrightarrow \\ t &= \begin{cases} (\rho - \sqrt{\rho^2 - a^2})\vec{n}_s\cdot\vec{e}_z + a\vec{t_c}\cdot\vec{e}_z & \text{if } -\frac{\pi}{2} < \theta < \frac{\pi}{2} \\ (\rho + \sqrt{\rho^2 - a^2})\vec{n}_s\cdot\vec{e}_z + a\vec{t_c}\cdot\vec{e}_z & \text{if } \frac{\pi}{2} \le \theta \le \frac{3}{2}\pi \end{cases} \end{aligned} \tag{A.10}$$

where $\vec{e}_z$ is the unit build direction vector. When $\delta \ll r$ is assumed, equation A.8 can be approximated by

$$a = 2\sqrt{2\delta\rho} \tag{A.11}$$

However, this negligence of 'higher-order terms' is not recommended when maximum allowed error δ and radius of curvature ρ approach each other in order of magnitude, which is quite possible in many situations.

Note that in general, the build direction vector does not necessarily coincide with the cross-section plane, although depicted as such in the figure for convenience. Equation A.10 can be visualized as is done in figure A.3. Dark areas and thicker lined graphs represent domains that are likely to be encountered often. Note that the dashed line represents the $\theta = \frac{\pi}{2}$ situation in the upper two plots. Also note that the thickness estimation can be negative in extreme cases, although this would correspond to an unlikely domain. Nevertheless, this should be taken into account. Figure A.2 compares approximation equation A.6 to equation A.10 on a different scale.



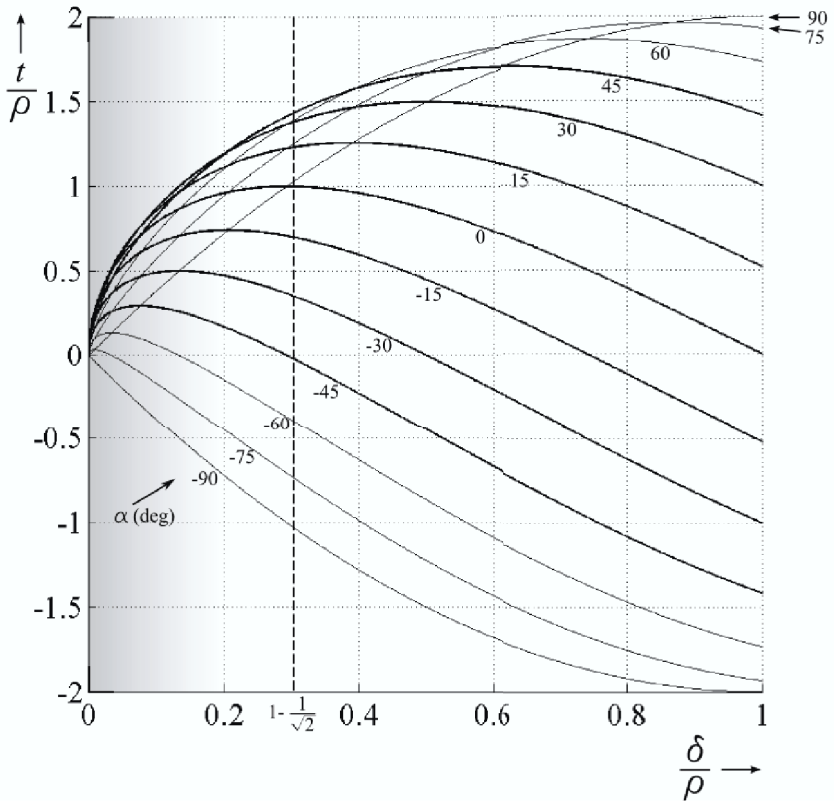


Figure A.3: Upper: normalized thickness estimation a.f.o. angle. Lower: normalized thickness a.f.o. maximum allowed cusp height.

Appendix B

Normal curvature

In this appendix part, normal curvature is derived using basic differential geometric descriptions of curves and surfaces. Before examining the normal curvature, the curvature of a single parametric space curve is derived. In order to do this, A few other formulas are of the essence and therefore treated first. Vectors are denoted bold instead of arrows on top.

B.1 Radius of a circle through three points

Given three planes $\mathbf{N}_1 \cdot \mathbf{p} = d_1$, $\mathbf{N}_2 \cdot \mathbf{p} = d_2$ and $\mathbf{N}_3 \cdot \mathbf{p} = d_3$, the intersection of these planes is given by

$$\mathbf{r} = \frac{d_1(\mathbf{N}_2 \times \mathbf{N}_3) + d_2(\mathbf{N}_3 \times \mathbf{N}_1) + d_3(\mathbf{N}_1 \times \mathbf{N}_2)}{\mathbf{N}_1 \cdot (\mathbf{N}_2 \times \mathbf{N}_3)}, \tag{B.1}$$

provided that the denominator is not zero. This means no set of two planes can be parallel. Given a circle through three points as shown in figure B.1, the radius can be calculated by using equation B.1. As can be seen derived the figure, the three planes are given by $\mathbf{p} \cdot (\mathbf{a} \times \mathbf{b}) = 0$ and bisecting planes $\mathbf{p} \cdot \mathbf{a} = \frac{|\mathbf{a}|}{2}$ and $\mathbf{p} \cdot \mathbf{b} = \frac{|\mathbf{b}|}{2}$ It can be shown Faux and Pratt [9] that

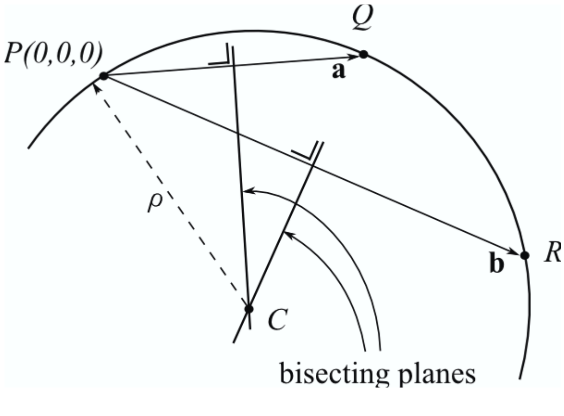


Figure B.1: Circle through three points

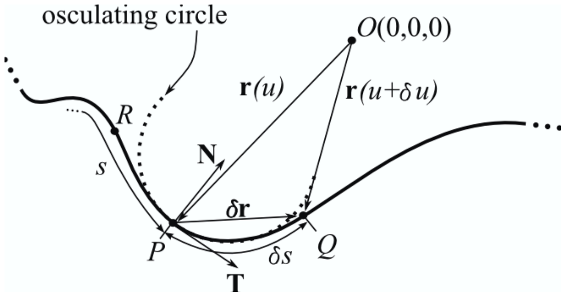


Figure B.2: Space curve

$$\rho = \frac{|\mathbf{a}||\mathbf{b}||\mathbf{a}-\mathbf{b}|}{2|\mathbf{a}\times\mathbf{b}|} \tag{B.2}$$

B.2 Curvature of a space curve

Consider a space curve $\mathbf{r}(u)$ as depicted in figure B.2. Take s, the arc length, as parameter.Then $|\delta\mathbf{r}|$ equals δs in the limit and

$$\frac{d\mathbf{r}}{ds} = \lim_{\delta s\to 0}\frac{\delta\mathbf{r}}{\delta s} = \mathbf{T} \tag{B.3}$$

is the unit tangent vector along the curve at P. Now,

$$\frac{d\mathbf{r}}{du} = \frac{d\mathbf{r}}{ds}\frac{ds}{du} \text{ or } \frac{d\mathbf{r}}{du} = \mathbf{T}\frac{ds}{du}, \tag{B.4}$$

also denoted by

$$\dot{\mathbf{r}} = \dot{s}\mathbf{T} \tag{B.5}$$

Further, the principal normal vector $\mathbf{N}$ is defined using

$$\frac{d\mathbf{T}}{ds} = \kappa\mathbf{N} \tag{B.6}$$

with $\mathbf{N}$ perpendicular to $\mathbf{T}$. Using the convention that $\kappa > 0$, $\mathbf{N}$ determines the sense of $\dot{\mathbf{T}}$:

$$\dot{\mathbf{T}} = \frac{d\mathbf{T}}{du} = \dot{s}\kappa\mathbf{N} \tag{B.7}$$

The plane spanned by the tangent and normal vector is called the osculating plane. The normal of this plane is called the binormal vector:

$$\mathbf{B} = \mathbf{T}\times\mathbf{N} \tag{B.8}$$

In the osculating plane, the osculating circle coincides with the curve at P as $\delta u \to 0$. The reciprocal of the osculating circle radius, κ, then represents the curvature, as will be shown later.

It will now be shown that the circle actually lies in the osculating plane: Suppose P, Q and R have parameters u, $u+\delta u$, and $u-\delta u$, respectively. Then these points coincide with the circle in the limit, which should yield $\overrightarrow{PQ} \times \overrightarrow{PN}$ to be parallel with $\mathbf{B}$. Using Taylor series, this can be proved:

$$\overrightarrow{PQ} \times \overrightarrow{PR} = [\mathbf{r}(u+\delta u) - \mathbf{r}(u)] \times [\mathbf{r}(u-\delta u) - \mathbf{r}(u)] \Leftrightarrow \tag{B.9}$$

$$\overrightarrow{PQ} \times \overrightarrow{PR} = [\delta u\ \dot{\mathbf{r}}(u) + \frac{\delta u^2}{2}\ddot{\mathbf{r}}(u)] \times [-\delta u\ \dot{\mathbf{r}}(u) + \frac{\delta u^2}{2}\ddot{\mathbf{r}}(u)] + O(\delta u^4) \Leftrightarrow \tag{B.10}$$

$$\overrightarrow{PQ} \times \overrightarrow{PR} = \delta u^3[\dot{\mathbf{r}}(u) \times \ddot{\mathbf{r}}(u)] + O(\delta u^4) \tag{B.11}$$

Now, from B.5 we derive using the chain rule and by substituting B.7:

$$\ddot{\mathbf{r}} = \ddot{s}\mathbf{T} + \dot{s}\dot{\mathbf{T}} = \ddot{s}\mathbf{T} + \dot{s}^2\kappa\mathbf{N} \tag{B.12}$$

Substituting B.5 and B.12 in B.11, after some manipulation using B.8, we get

$$\overrightarrow{PQ} \times \vec{PR} = \delta u^3 \dot{s}^3 \kappa \mathbf{B} + O(\delta u^4) \tag{B.13}$$

Hereby it has been shown that $\vec{PQ} \times \vec{PR}$ is parallel to $\mathbf{B}$.

Now it is shown that the osculating circle curvature equals the curvature of curve $\mathbf{r}$ at P. Write

$$\overrightarrow{PR} = -\dot{s}\mathbf{T_P}du + O(\delta u^2) \text{ and} \tag{B.14}$$

$$\overrightarrow{PQ} = \dot{s}\mathbf{T_P}du + O(\delta u^2) \tag{B.15}$$

Now substitute these expressions in the numerator of relation B.2 using $\mathbf{a} = -\overrightarrow{PR}$ and $\mathbf{b} = -\overrightarrow{PR} + \overrightarrow{PQ}$. Expression B.13 can, after rearranging, be used for substitution as the cross-product in the denominator. In the limit of $\delta u \to 0$, we get

$$\rho = \frac{\dot{s}^3}{|\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|} = \frac{\dot{s}^3}{\dot{s}^3\kappa} = \frac{1}{\kappa} \tag{B.16}$$

So hereby it is shown that the osculating circle radius is the reciprocal curvature of $\mathbf{r}$. Using expression B.7, we arrive at the frequently used expression for the curvature:

$$\kappa = \frac{|\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|}{|\dot{\mathbf{r}}^3|} \tag{B.17}$$

B.3 Basic surface properties

Let S be a parametric surface, described by parameters u and v. Define $u = u(w)$ and $v = v(w)$ or $\mathbf{u} = [u(w), v(w)]^T$ as a curve C on S.Define $\mathbf{r}(w)$ as a point on C and $\mathbf{r}(u, v)$ as a point on S. The tangent vector to the curve is expressed as

$$\dot{\mathbf{r}} = \frac{\partial \mathbf{r}}{\partial u}\dot{u} + \frac{\partial \mathbf{r}}{\partial v}\dot{v} = \mathbf{A}\dot{\mathbf{u}} \tag{B.18}$$

where

$$A = \begin{bmatrix} \frac{\partial \mathbf{r}}{\partial u} & \frac{\partial \mathbf{r}}{\partial v} \end{bmatrix} \tag{B.19}$$

The squared tangent vector length is given by

$$\dot{s}^2 = |\dot{\mathbf{r}}|^2 = \dot{\mathbf{r}}^T\dot{\mathbf{r}} = \dot{\mathbf{u}}^T\mathbf{A}^T\mathbf{A}\dot{\mathbf{u}} = \dot{\mathbf{u}}^T\mathbf{G}\dot{\mathbf{u}} \tag{B.20}$$

with $\mathbf{G}$ the first fundamental matrix of the surface:

$$\mathbf{G} = \mathbf{A}^T\mathbf{A} = \begin{bmatrix} \frac{\partial \mathbf{r}}{\partial u}\cdot\frac{\partial \mathbf{r}}{\partial u} & \frac{\partial \mathbf{r}}{\partial u}\cdot\frac{\partial \mathbf{r}}{\partial v} \\ \frac{\partial \mathbf{r}}{\partial v}\cdot\frac{\partial \mathbf{r}}{\partial u} & \frac{\partial \mathbf{r}}{\partial v}\cdot\frac{\partial \mathbf{r}}{\partial v} \end{bmatrix} \tag{B.21}$$

This matrix is symmetric.

B.4 Normal curvature of a parametric surface

In this section, the general space curve curvature will be connected to curvature properties of a parametric surface. From section B.2, we equate expression B.12 containing κ with the second derivative of the surface, obtained by differentiation of expression B.18. In other words, two expressions are combined, both describing the second derivative of a specific curve path $\mathbf{u} = \mathbf{u}(t)$ on surface $\mathbf{r} = \mathbf{r}(u, v)$, where subscript c denotes the curve:

$$\ddot{\mathbf{r}} = \ddot{s}\mathbf{T} + \dot{s}^2\kappa_c\mathbf{N}_c = \frac{\partial^2 \mathbf{r}}{\partial u^2}\dot{u}^2 + 2\frac{\partial^2 \mathbf{r}}{\partial u \partial v}\dot{u}\dot{v} + \frac{\partial^2 \mathbf{r}}{\partial v^2}\dot{v}^2 + \frac{\partial \mathbf{r}}{\partial u}\ddot{u} + \frac{\partial \mathbf{r}}{\partial v}\ddot{v} \tag{B.22}$$

Suppose, we now project $\ddot{\mathbf{r}}$ on surface normal $\mathbf{N}_s$. Using the fact that $\mathbf{N}_s$ is orthogonal to $\mathbf{T}$, $\frac{\partial \mathbf{r}}{\partial u}$ and $\frac{\partial \mathbf{r}}{\partial v}$, we get:

$$\ddot{\mathbf{r}} = \dot{s}^2\kappa_c\mathbf{N}_c\cdot\mathbf{N}_s = \mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial u^2}\dot{u}^2 + 2\mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial u \partial v}\dot{u}\dot{v} + \mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial v^2}\dot{v}^2 \tag{B.23}$$

or

$$\dot{s}^2\kappa_c\mathbf{N}_c\cdot\mathbf{N}_s = \dot{\mathbf{u}}^T\mathbf{D}\dot{\mathbf{u}} \tag{B.24}$$

with

$$\mathbf{D} = \begin{bmatrix} \mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial u^2} & \mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial u \partial v} \\ \mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial v \partial u} & \mathbf{N}_s\cdot\frac{\partial^2 \mathbf{r}}{\partial v^2} \end{bmatrix}. \tag{B.25}$$

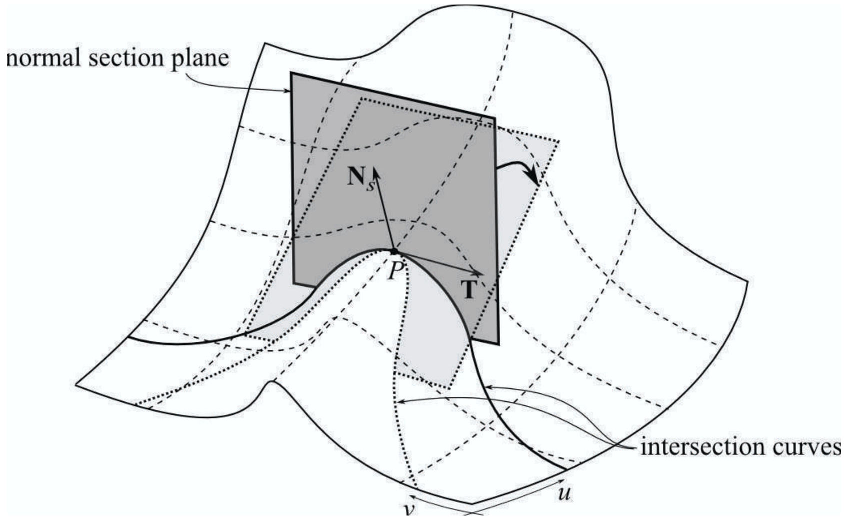


Figure B.3: Normal curvature

This matrix is the second fundamental matrix of the surface and is symmetric. At this time, the normal curvature is finally introduced as $\kappa_n \equiv \kappa_c \mathbf{N}_c \cdot \mathbf{N}_s$, which corresponds to the direction $\mathbf{A}\dot{\mathbf{u}}$ or simply the tangent of the surface curve through P. Substituting the final expression from B.20 in B.24 and rearranging, we get

$$\kappa_n \equiv \kappa_c \mathbf{N}_c \cdot \mathbf{N}_s = \frac{\dot{\mathbf{u}}^T \mathbf{D} \dot{\mathbf{u}}}{\dot{\mathbf{u}}^T \mathbf{G} \dot{\mathbf{u}}} \tag{B.26}$$

The projection of the surface curve normal on the surface normal at P actually gives the surface normal directional component of $\ddot{\mathbf{r}}(t)$. This explains the origin of the term 'normal curvature'. The other component, tangent to the surface, is called the geodesic curvature, but is not treated further here. The sign of κ_n is positive when the curve normal turns *toward* the surface normal. A corollary from B.26 is that κ_n solely depends on the tangent direction of curve $\mathbf{u}$ and that it is independent from the curve's principal normal vector direction:

The Meusnier theorem states that *'All curves lying on a surface S and having at a given point $P \in S$ the same tangent line have have at this point the same normal curvatures'*.

Graphically, this can be illustrated. Let a plane containing the surface normal at P intersect the surface through P as shown in figure B.3. This kind of plane is called a 'normal plane'. This yields an intersection curve between the normal plane and the surface, whose tangent $\mathbf{T}$ at P also lies in the normal plane. Now rotate the plane about $\mathbf{T}$ by an arbitrary angle. The intersection curve is thereby changed, also changing the curvature of the curve through P. However, κ_n remains unchanged as in the neighborhood of P, the curves run in the same tangent direction leaving $\dot{\mathbf{u}}$ at P unchanged. Note that this means that the geodesic curvature changes.

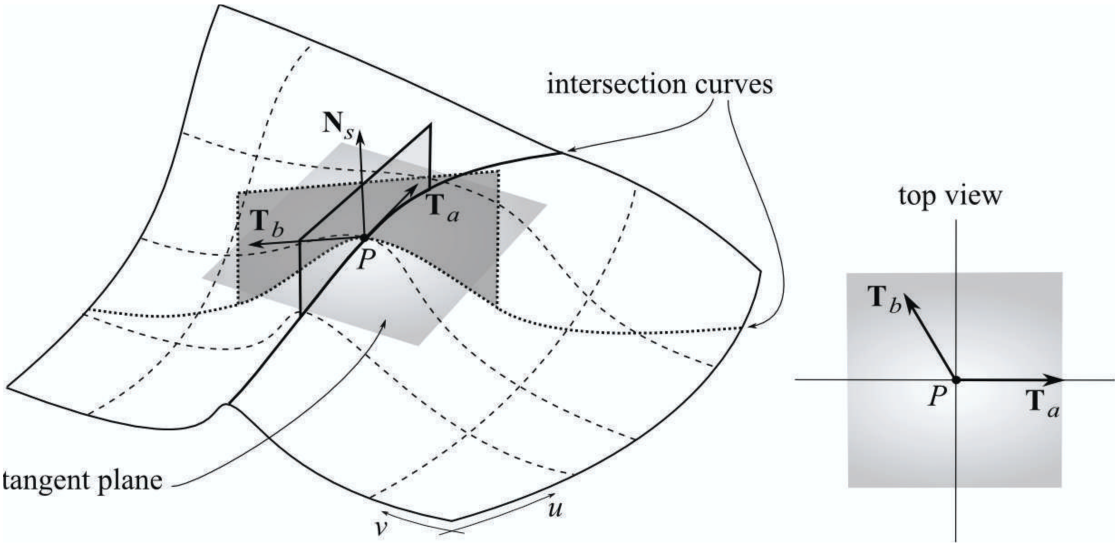


Figure B.4: Tangent plane and normal curvatures in different directions

Now denote $\mathbf{A}$ and $\mathbf{D}$ as

$$\mathbf{A} = \begin{bmatrix} E & F \\ F & G \end{bmatrix} \text{ and } \mathbf{D} = \begin{bmatrix} L & M \\ M & N \end{bmatrix} \tag{B.27}$$

Using this notation, a more frequently encountered form of B.26 can be established when canceling out the differential dt's, yielding

$$\kappa_n = \frac{Ldu^2 + 2Mdudv + Ndv^2}{Edu^2 + 2Fdudv + Gdv^2} \tag{B.28}$$

or

$$\kappa_n = \kappa_n \left(\frac{dv}{du} \right) = \frac{L + 2M\dfrac{dv}{du} + N\left(\dfrac{dv}{du}\right)^2}{E + 2F\dfrac{dv}{du} + G\left(\dfrac{dv}{du}\right)^2}. \tag{B.29}$$

Now fraction $\dfrac{dv}{du}$ determines the intersection curve direction in the tangent plane on the surface at point P. Figure B.4 shows two examples. Note that in general, the u- and v-direction along the surface are not necessarily orthogonal.

Appendix C

Application iteration flow charts

In this appendix part, the application flow is presented, preceded by the legend. Some charts include additional legends with abbreviations used for compact presentation. In some charts, the route straight down is the most likely encountered case. This will improve readability.

C.1 Flow chart legend

Following the de facto IBM-standards, the following symbols are used to denote program flow items.

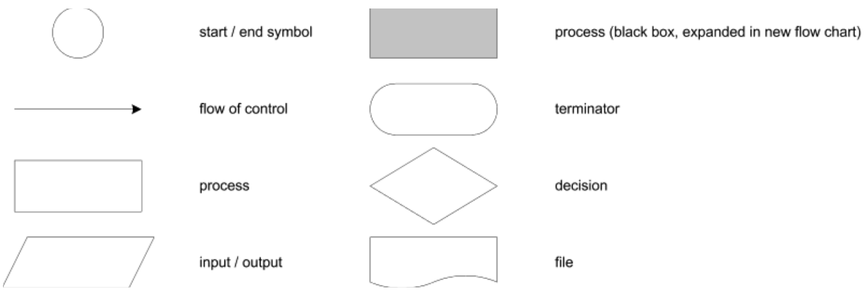


Figure C.1: Flow chart legend

C.2 Iteration flow charts

Figures C.2 and C.3 are repeated from chapter 4.

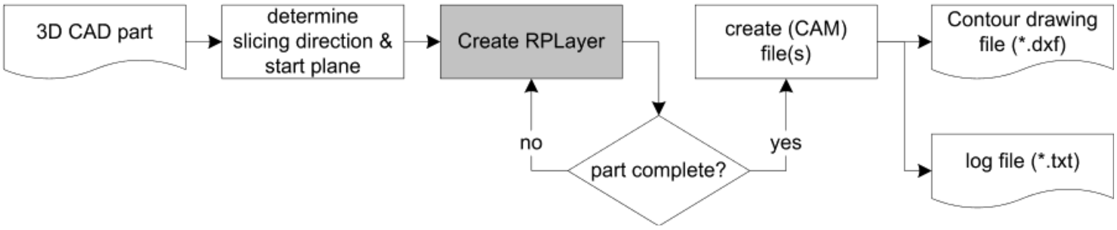


Figure C.2: Main application flow

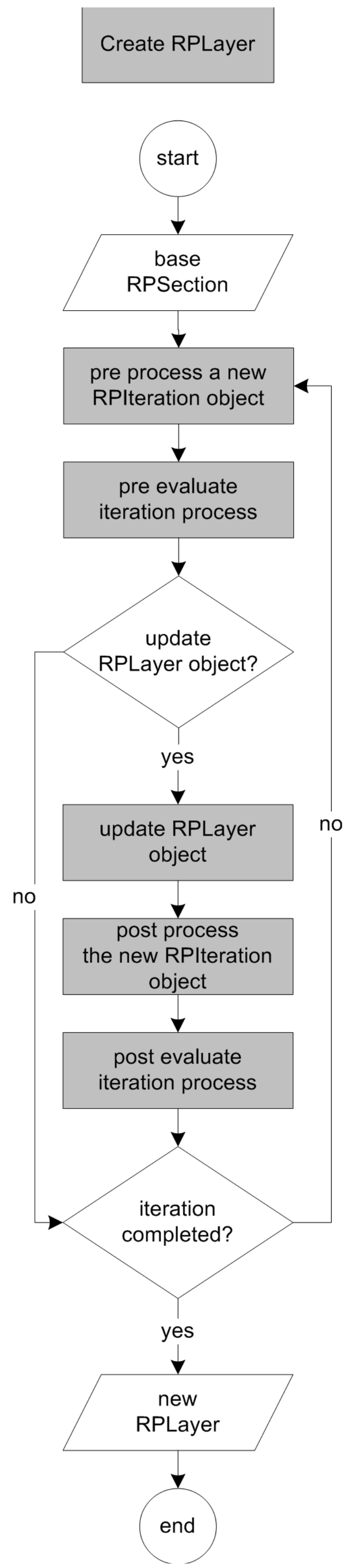


Figure C.3: Layer creation iteration scheme

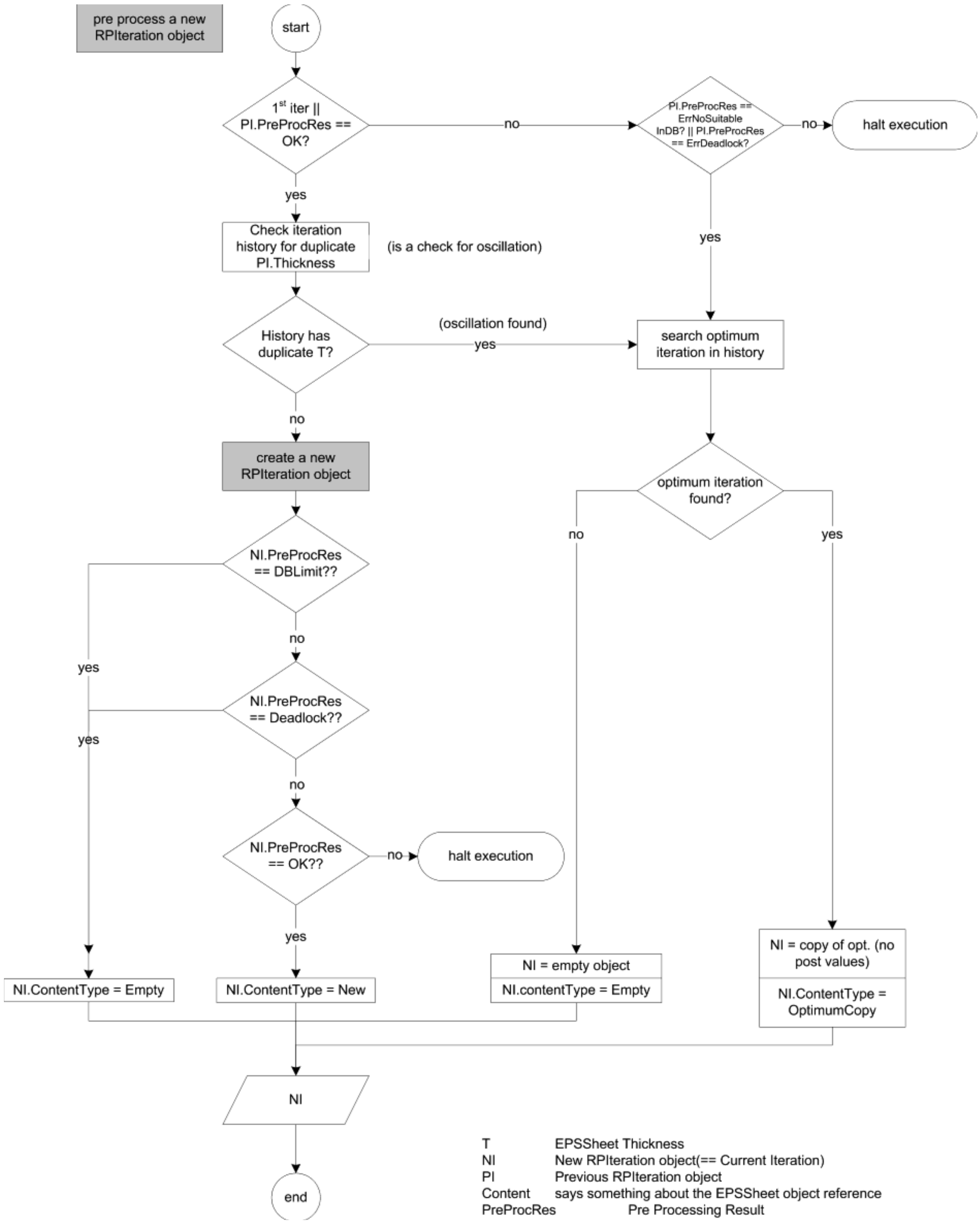


Figure C.4: Pre processing of an iteration

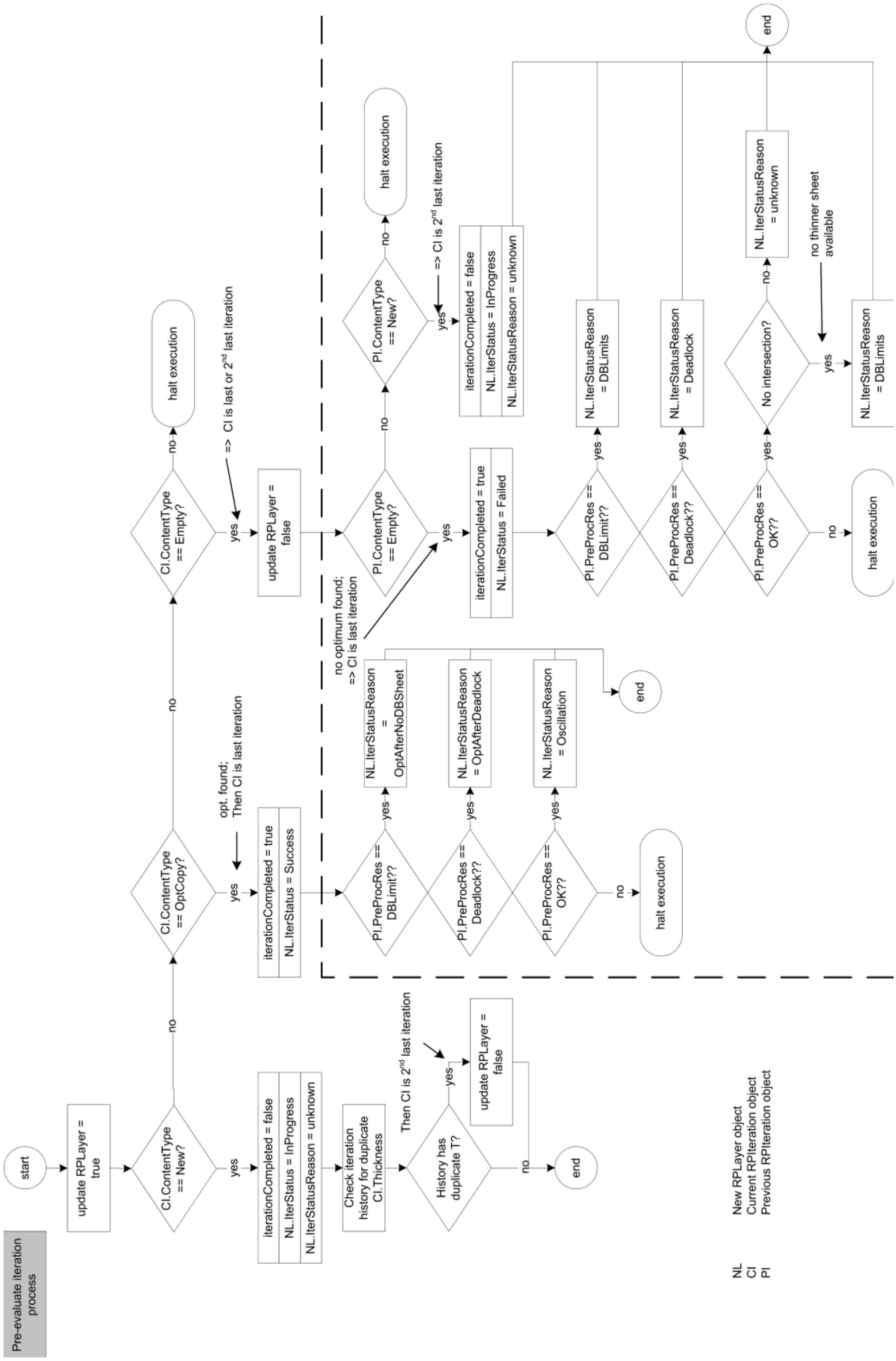


Figure C.5: Pre processing of the iteration process

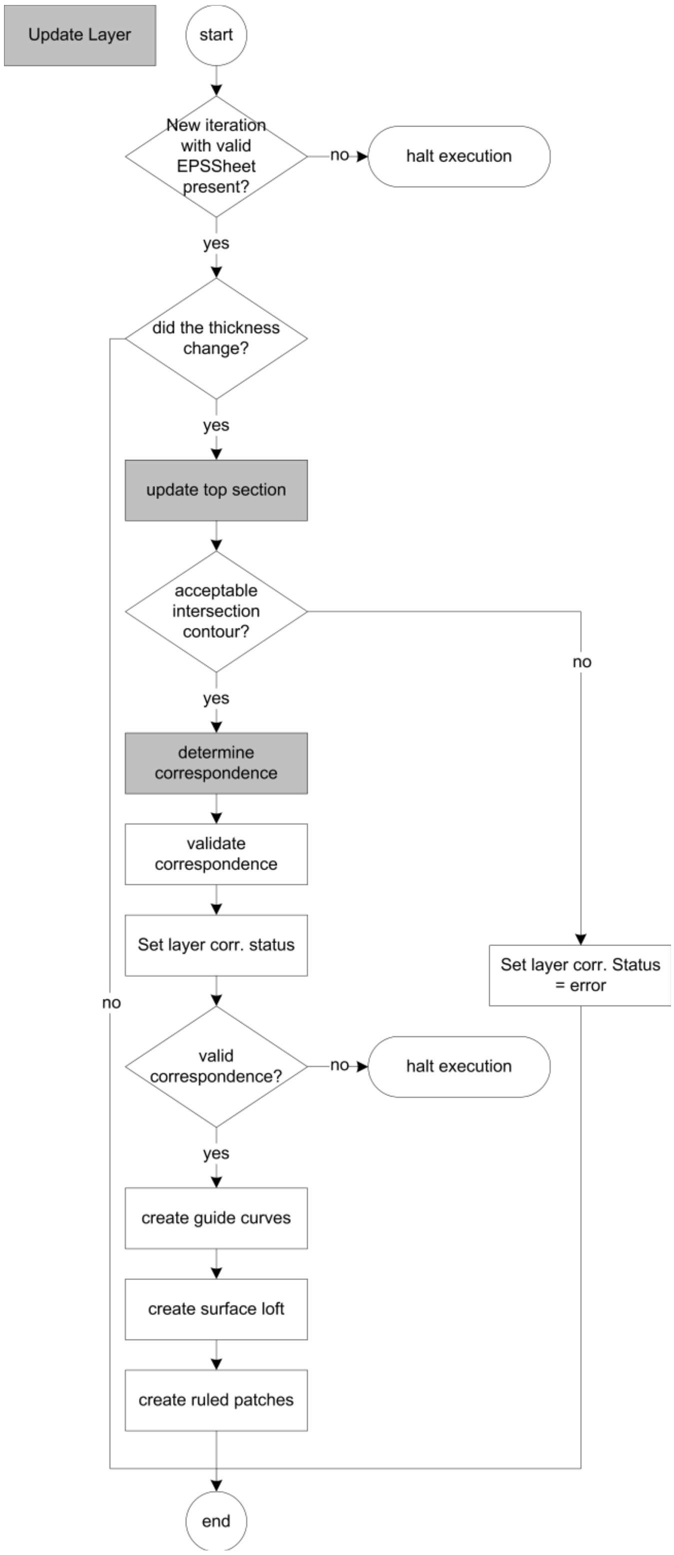


Figure C.6: Layer update

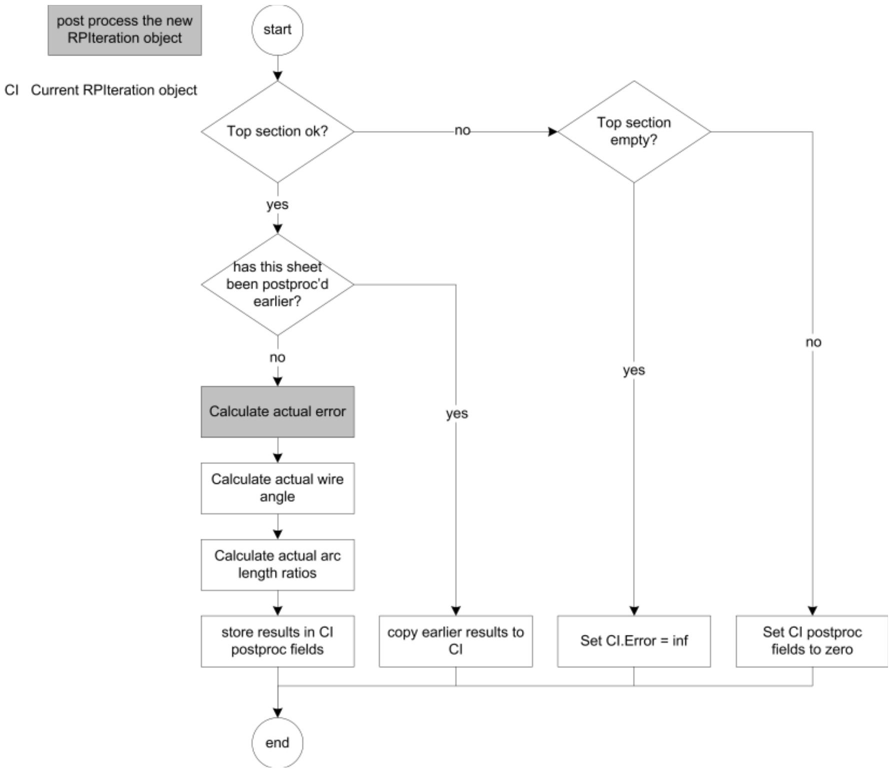


Figure C.7: Post processing of an iteration

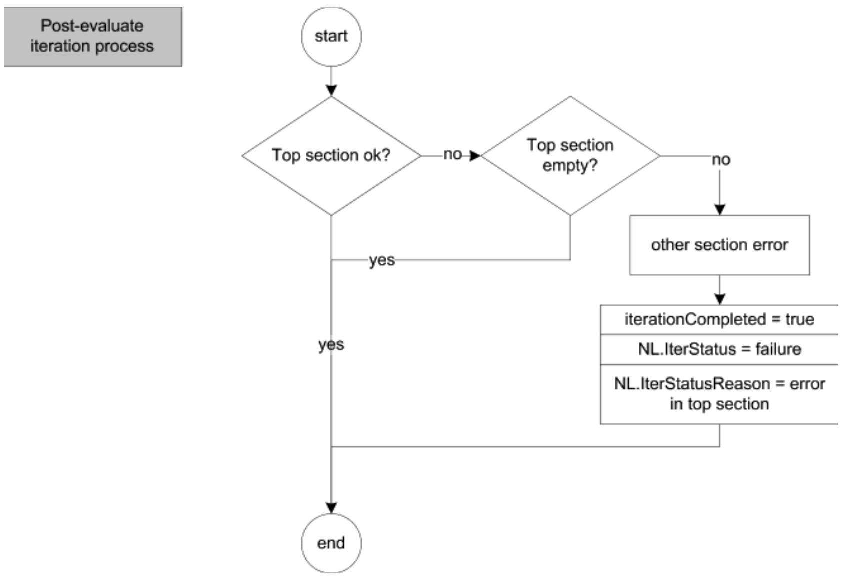


Figure C.8: Post processing of the iteration process

Appendix D

Application algorithm flow charts

D.1 Iteration flow charts (continued)

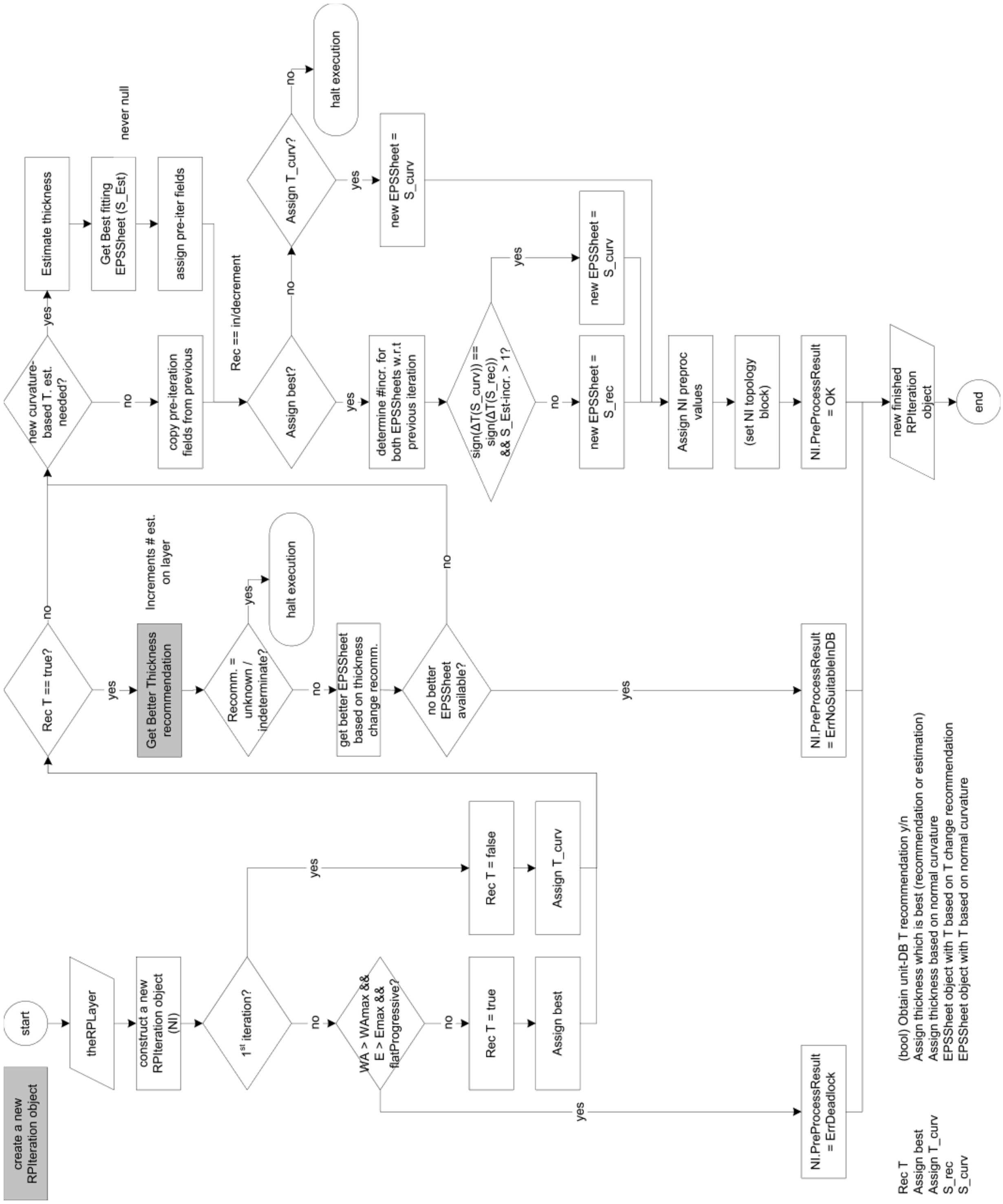


Figure D.1: Creation of an iteration

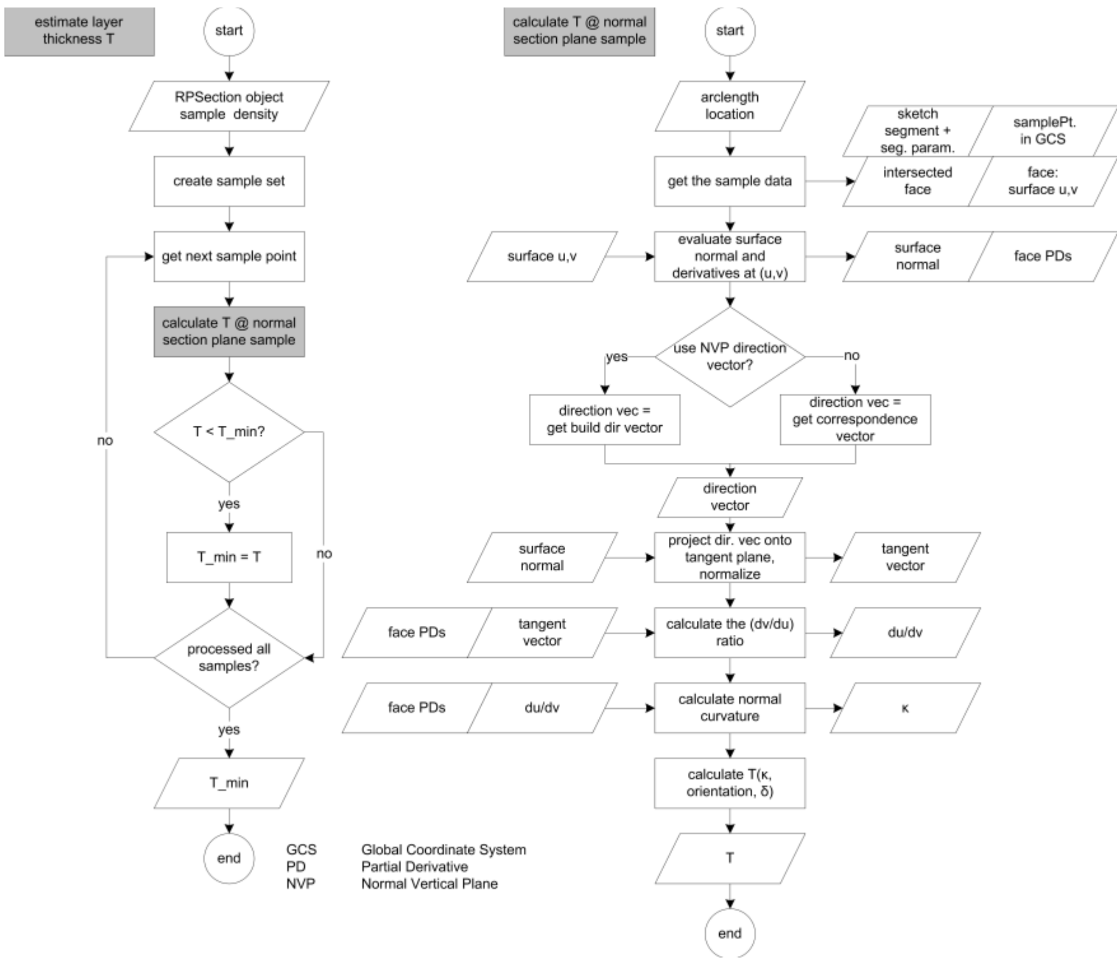


Figure D.2: Thickness estimation based on normal curvature

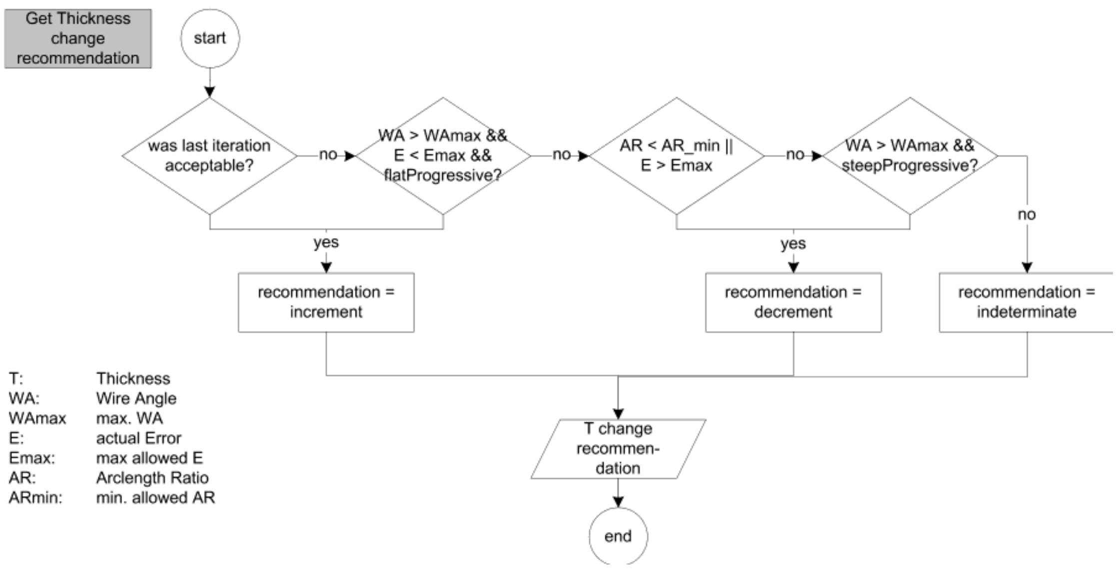


Figure D.3: Thickness change recommendation

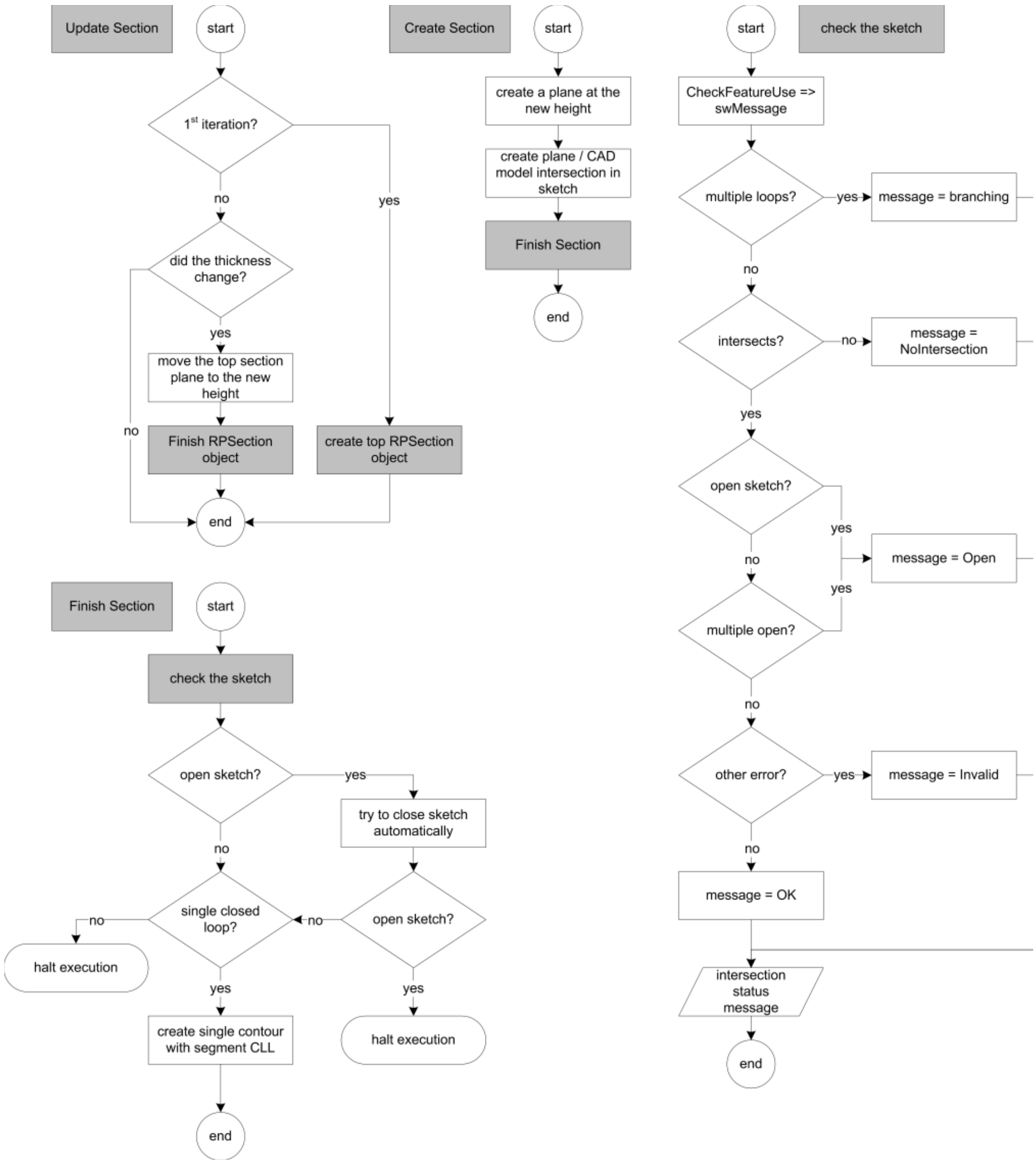


Figure D.4: Section update

D.2 Correspondence solving flow charts

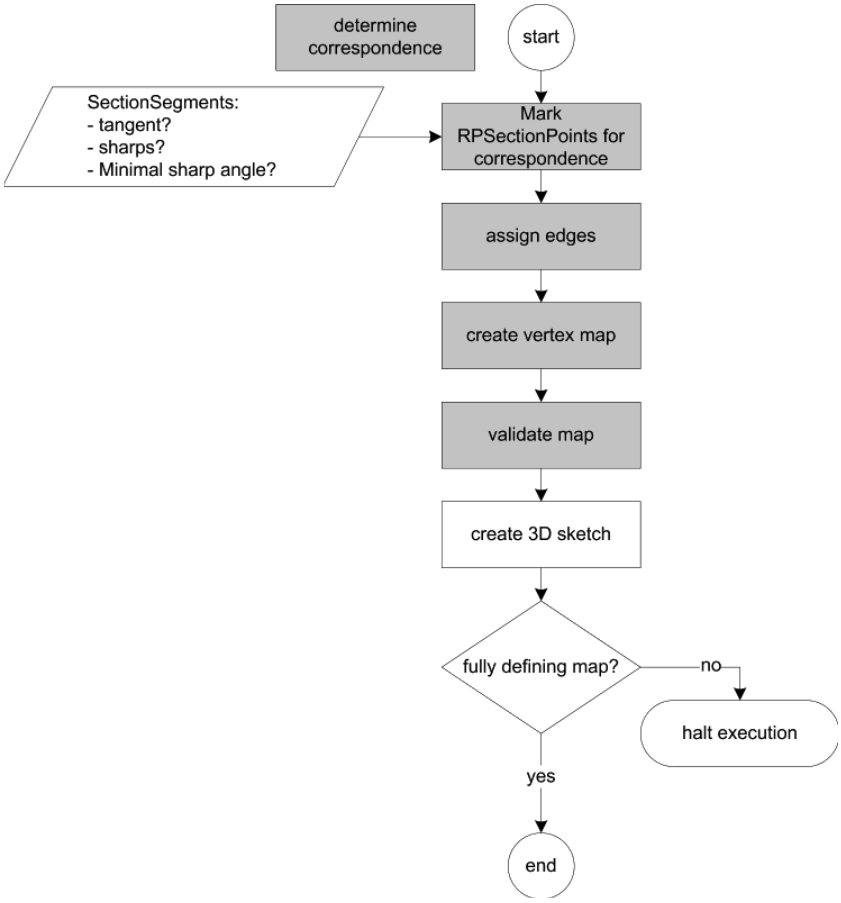


Figure D.5: Correspondence solving

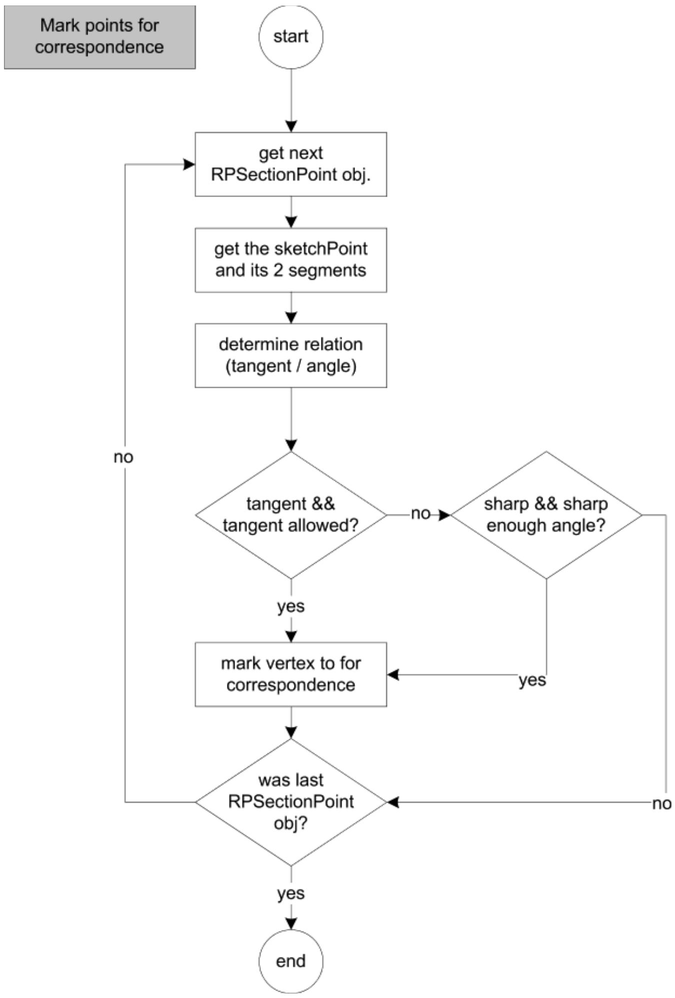


Figure D.6: Marking points for correspondence

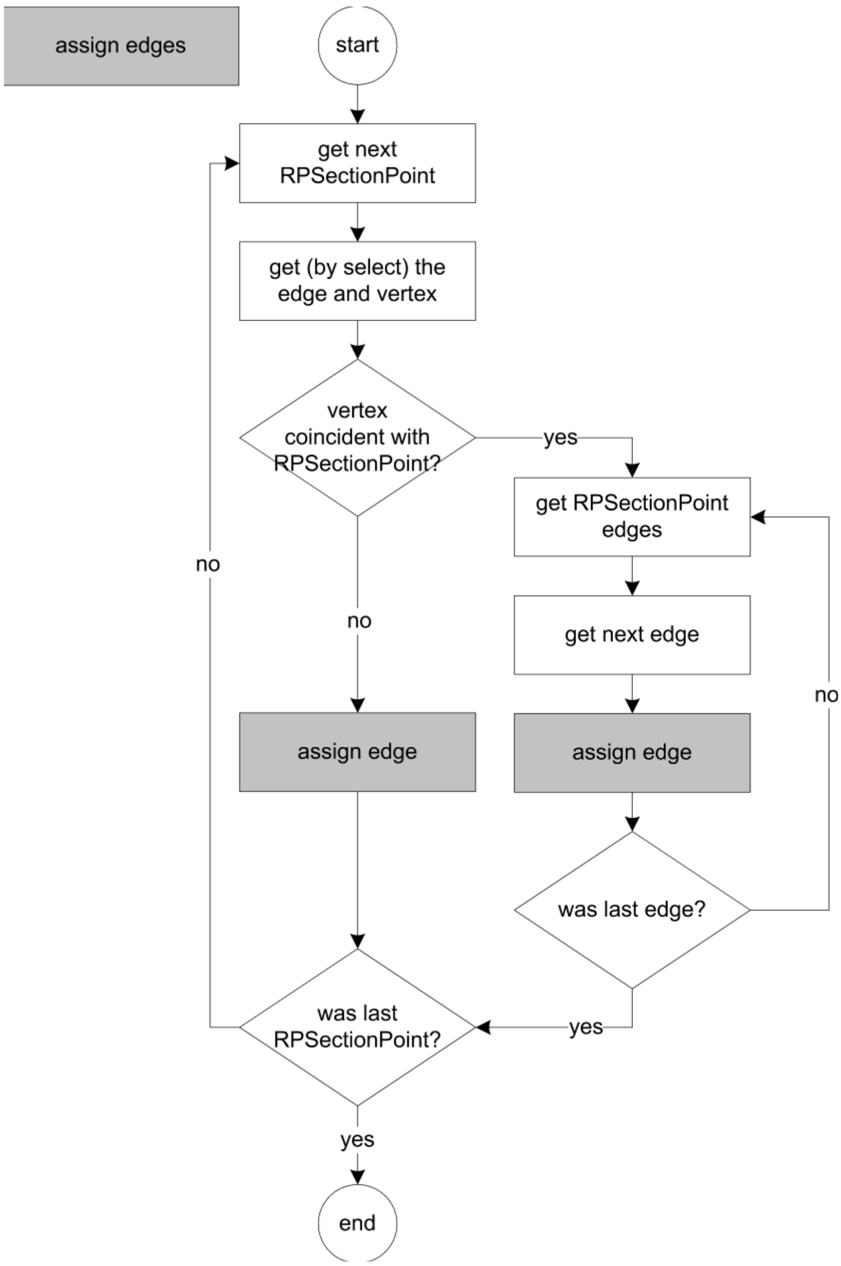


Figure D.7: Assigning edges to RPSection point objects

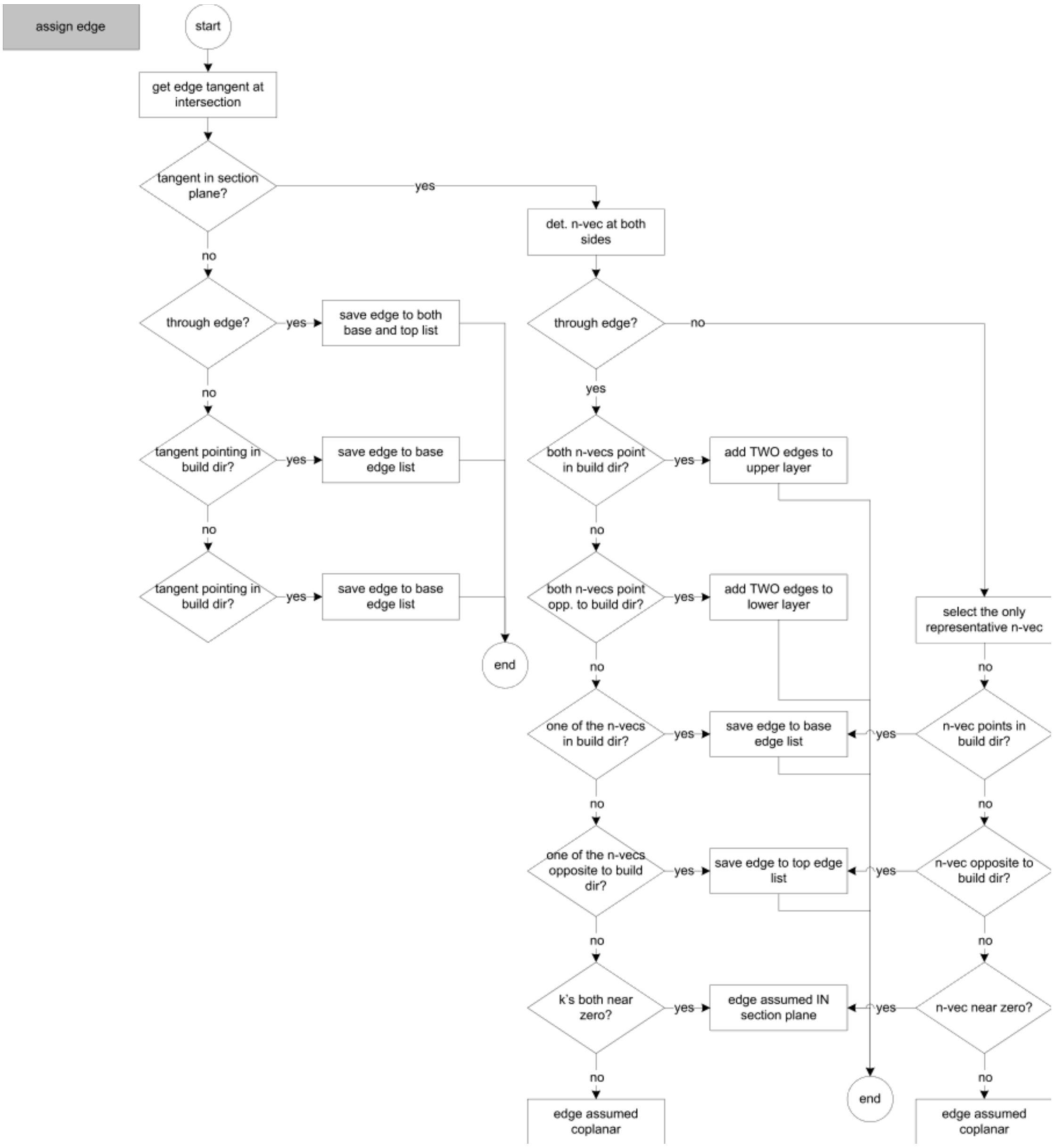


Figure D.8: Assigning single edge

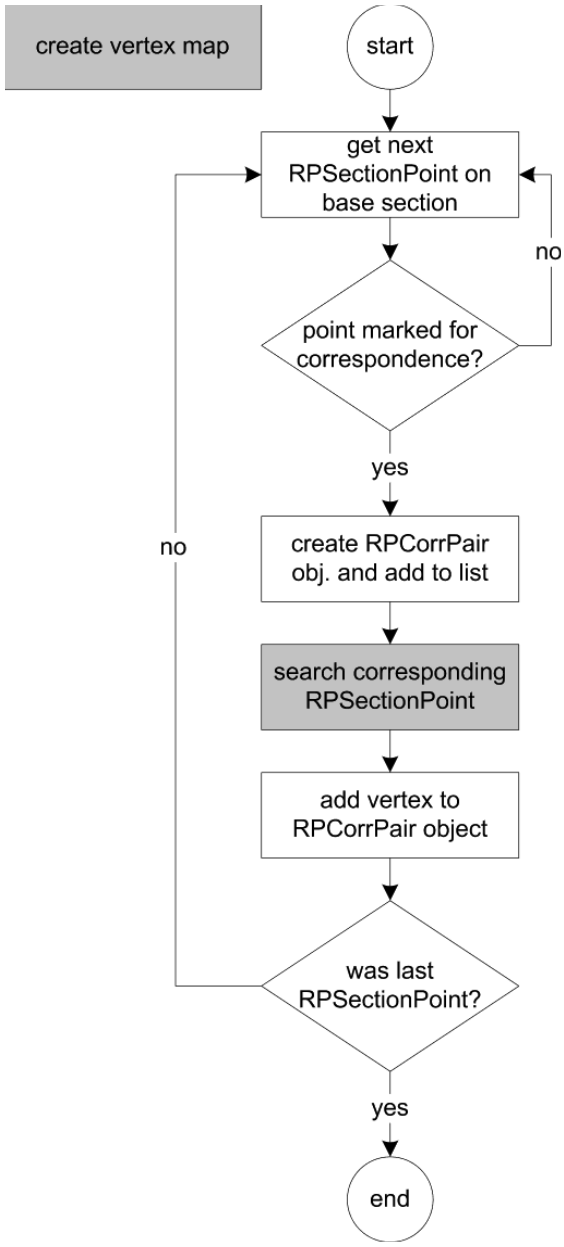


Figure D.9: Vertex mapping

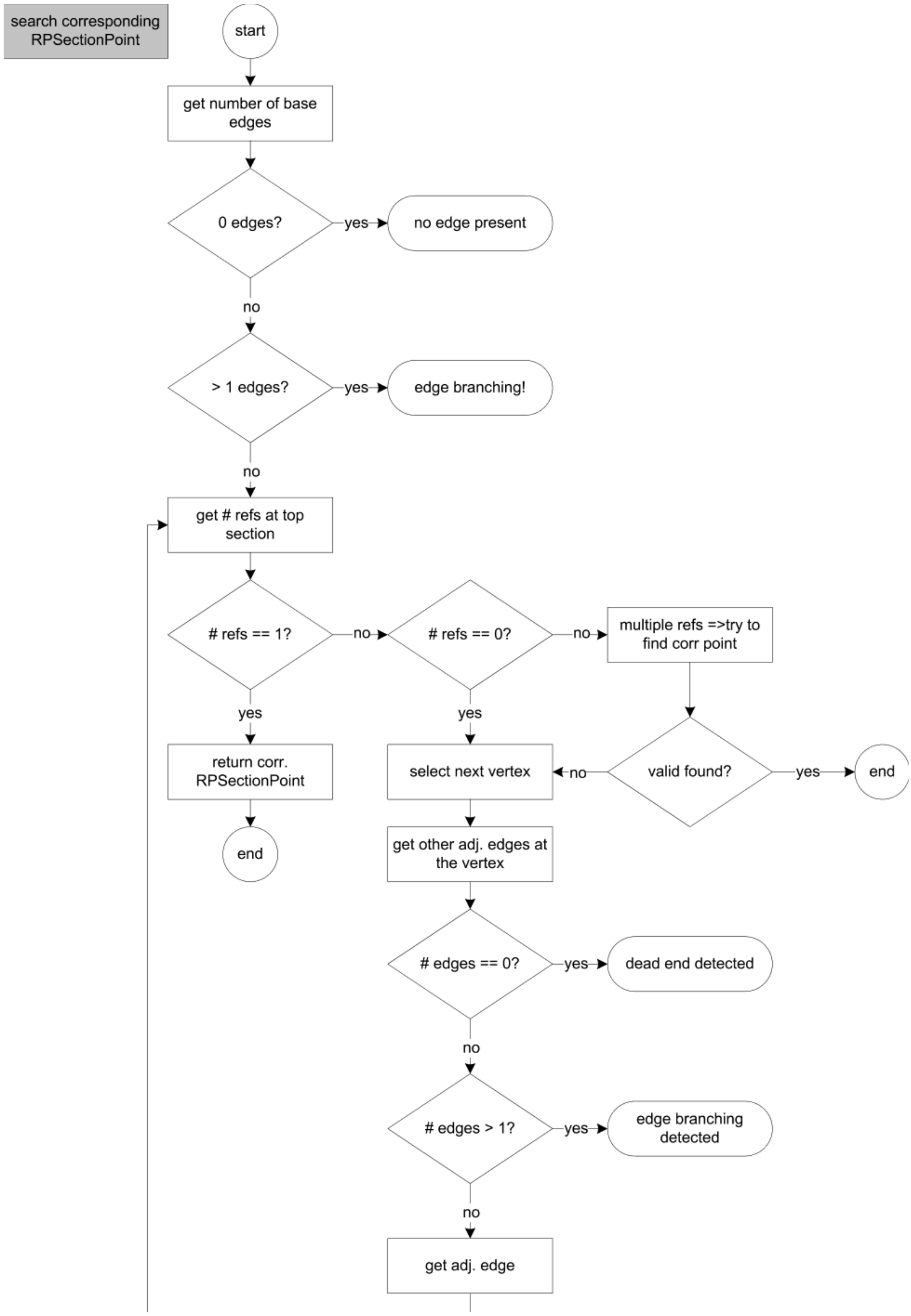


Figure D.10: Search for corresponding point

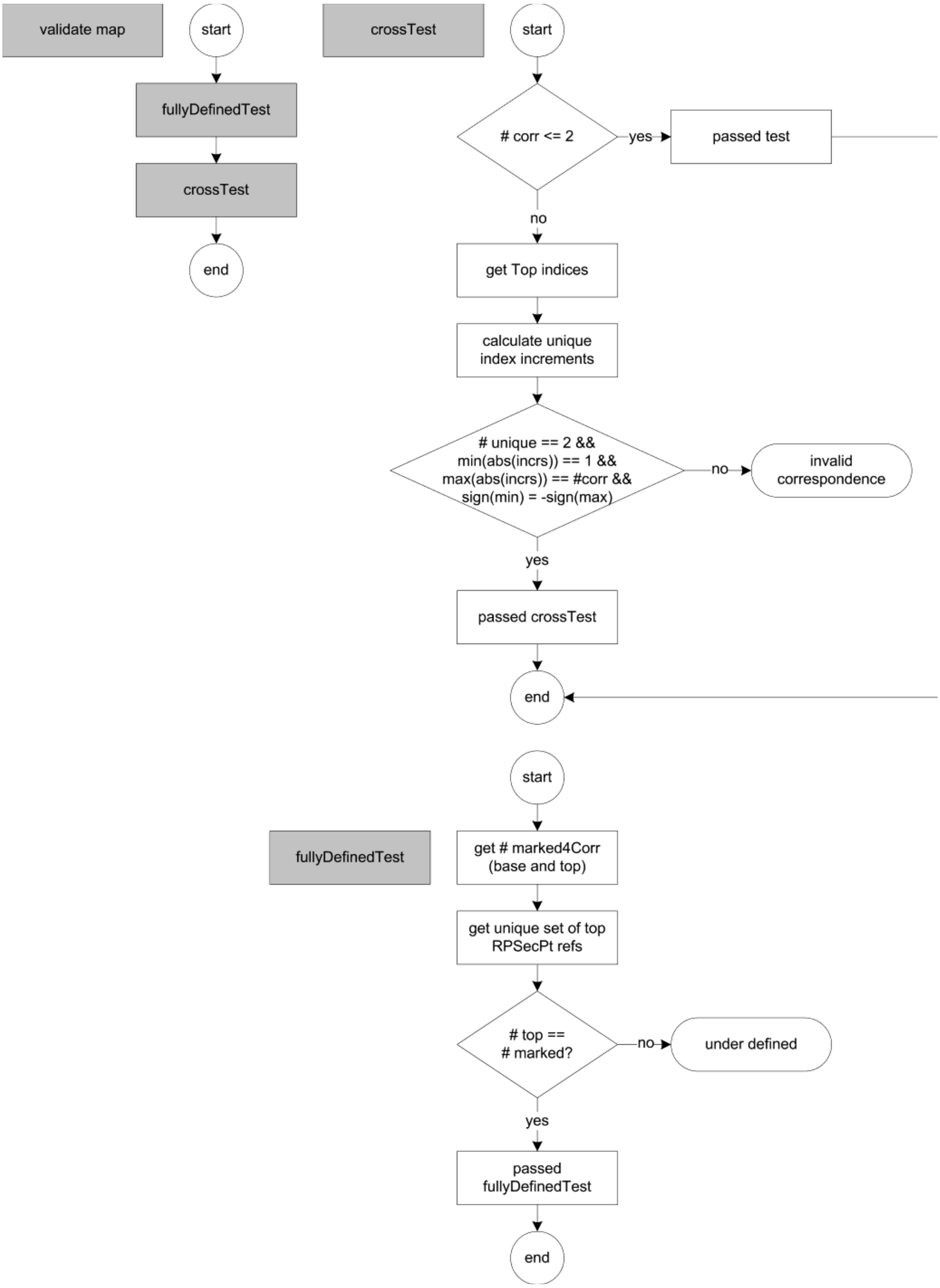


Figure D.11: Validation of found correspondence

D.3 Actual error flow charts

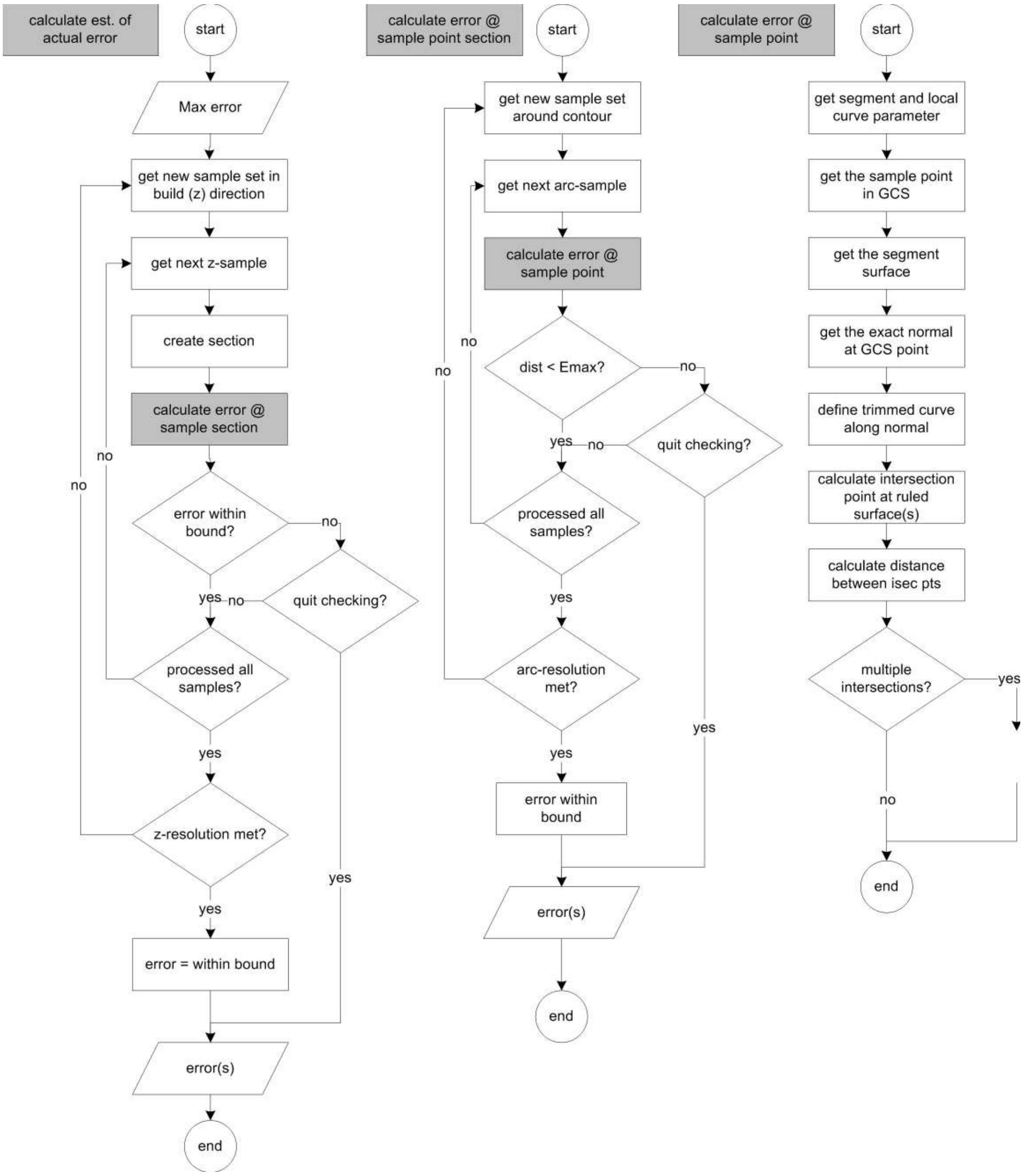


Figure D.12: Actual error sampling

Appendix E

Edge topology traversal

In this appendix part, the principle of topology traversal is explained for solving the correspondence problem.

E.1 Edge assignment for correspondence solving

When a cross-section from a CAD model is created at some location, intersection of a model face yields a curve and intersection of an edge yields a vertex. Intersection of a vertex, usually connecting multiple edges, also yields a vertex, which can be considered a special case of multiple edge intersection. In that case, the edge intersection is labeled as a 'non-through' edge as opposed to 'through' edge intersection, which will usually be the case. As edges are used for correspondence solving, these are of key interest. All forms of edge intersection are shown in figure E.1. It should be noted that some cases can result in problems in practice.

For each (sketch)point in a model/plane intersection, it must be determined if the intersected edge is available for correspondence solving to the layer above the section plane, below it, or both. This is called 'edge assignment'. Remember that a RPSectionPoint object is used to wrap the resulting sketchpoint of the edge's intersection. Edge assignment for an intersected edge is performed only if its RPSectionPoint object is *marked for correspondence* through the method explained in section 4.3.3. If not, the assignment is not necessary. The edge itself is wrapped in a SWEdgeWrapper object, containing fields indicating the assignment. The SWEdgeWrapper object is also made referenced by its RPSectionPoint. In the most likely cases, such as the upper left two examples of a through edge intersection, the intersected edge is assigned to both the lower and upper layer for correspondence solving. More difficult and error-prone cases are represented by the other cases in the figure. Decisions regarding assignment can result in subjective or undesired correspondence results. Nevertheless, it was decided to exploit the automation for correspondence solving as far and as generic as possible.

When analyzing which assignment is required, at first, the edge tangent $\vec{t}$ at the intersection point is evaluated in the edge sense. That is, in the direction from edge start point to edge end point. The sense can be determined by determining the sense of the edge w.r.t its underlying parametric curve representation, which is useful for periodic edges without start- and end point in particular. The projection $\vec{t}$ on $\vec{e}_z$ is determined next. If the projection magnitude is near zero ($< \epsilon_t$),

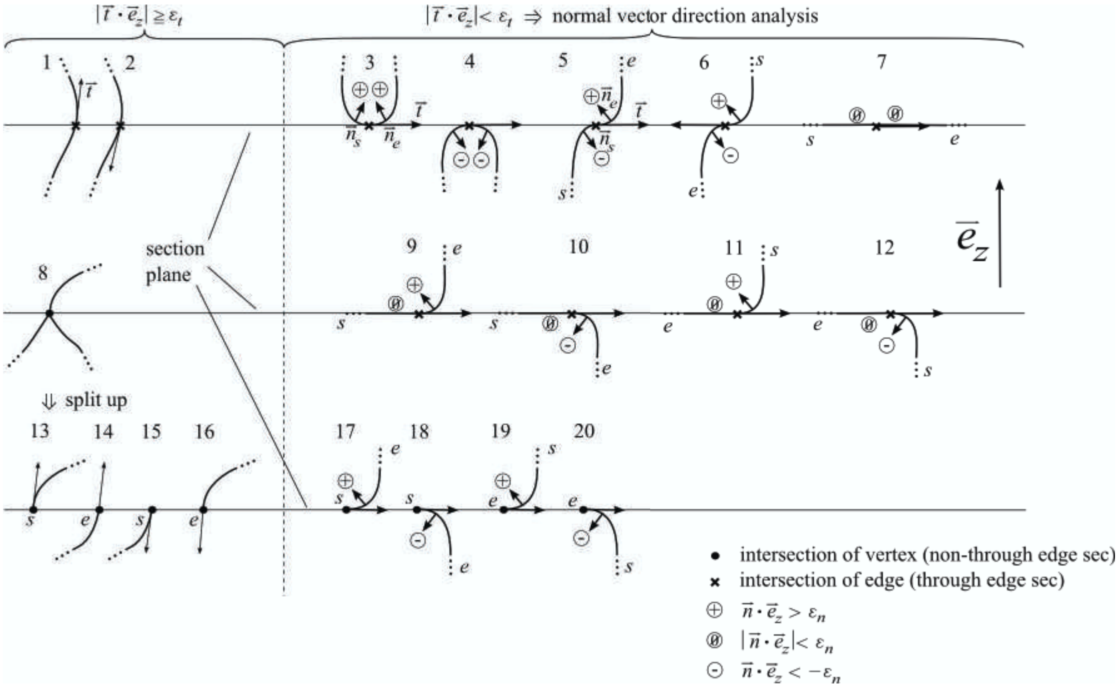


Figure E.1: Possible edge intersection cases

the direction of edge normal vector $\vec{n}$ in the neighborhood before and after the intersection location is analyzed. This information is all that is needed in order to determine assignment for correspondence solving. The idea is further shown in figure D.7 and D.8.

For cases 1 and 2, the edge is assigned for both upper and lower layer correspondence solving. Cases 3 and 4 will yield two assignments for the upper respectively lower layer. Cases 5 and 6 are treated as 1 and 2. 7 will not lead to assignments. Case 8 is decomposed in single edge intersection cases. cases 9, 11, 13, 16, 17 and 19 lead to upper layer assignment, 10, 12, 14, 15, 18 and 20 for lower layer assignment.

E.2 Topology traversal

The correspondence problem is solved using edge topology traversal. In order to elucidate the implementation, figure E.2 shows some of the more simple cases that represent the most likely encountered in practice. Case 1 shows the most likely case. Case 2 calls for a recursive edge topology traversal function, which follows the tangent edges from base section to the top section. The search is complete when an edge, intersected by the top-section, is found. This comes down to searching for the edge reference, which may exist at a RPSectionPoint object as described in the previous section. This case can happen very often when, for example, the CAD model is the output of a FEM program such as the diagonally drawn example depicted in figure 3.1. This is actually a case of edge branching. Other cases of edge branching are cases 4a, 4b and 5. These cases are highly ambiguous and usually require extra rules or user input in order to arrive at the intent of the designer.

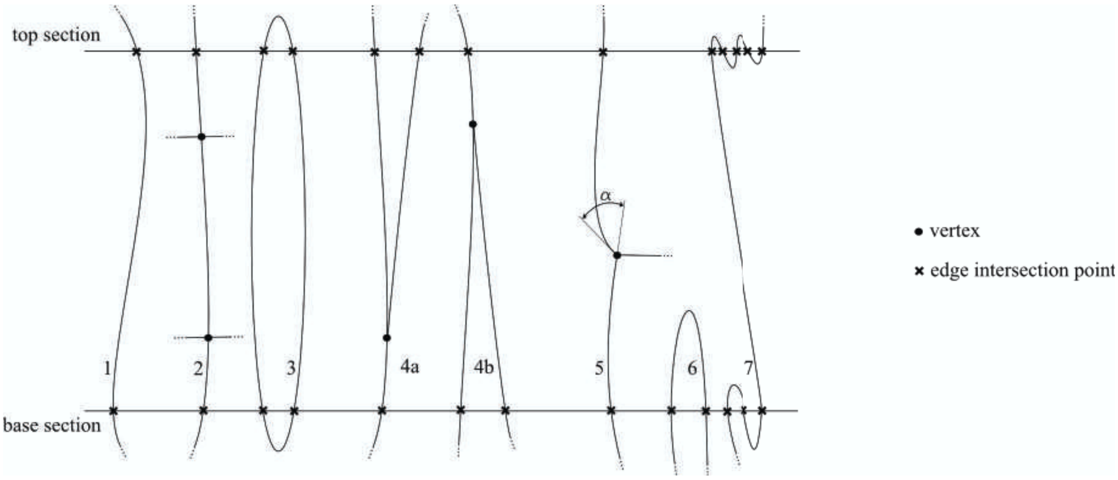


Figure E.2: Topology traversal cases

This is very hard to automate. Periodic edges can be present as well (case 3) or edges which are intersected multiple times (case 6, 7), which calls for inspection of the curve parameter values together with the edge sense in order to pick the correct intersection instance. Combinations are possible as well. Correspondence cases 1, 2 and 3 are solved well by the implemented methods, yet the other cases are not guaranteed to be treated robustly and usually require user interaction.

Appendix F

Demonstrator slicing results

In this appendix part, the slicing process is presented in tabular form. The table heading legend is shown in table F.1. The final layer numbers are added manually to the log files.

Symbol	**Meaning**
ID	layer ID
T_est	estimated layer thickness, based on κ_n
T_DB	layer thickness from the thickness database
AngMaxIni	estimated maximum wire angle at pre processing
kMaxConvex	maximum convex κ_n at the base section
kMaxConcav	maximum concave κ_n at the base section
ErrMaxAbs	maximum actual layer error
ArcMin	minimum actual arc length ratio
AngMax	maximum wire angle
PreProcRes	status message of pre processing

Table F.1: Log file legend

```
Slicing started at:             3-9-2007 9:07:48

Corr. pt. characteristics:
Tangent vertices allowed:       True
Min. angle:                     5,00 [deg]

Process constraints:
Max allowed wire angle:         55,00 [deg]

Accuracy constraints:
Max allowed chordal error:      10 [mm]
Min allowed contour ratio:      50,00 %

Sampling settings:
Number of samples per contour: 50
Number of z-error stages:       2

Stop error-eval when exceeds:   False
EPS Sheets available in DB:     50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 1000

   ID      T_est [mm]       T_DB [mm] AngMaxIni [deg] kMaxConvex[1/m] kMaxConcav[1/m]  ErrMaxAbs [mm]      ArcMin [%]    AngMax [deg]       PreProcRes

    1             104             100            56,4          -5,026           0,000           6,717          61,50%            44,9               OK
    2             105             150            44,9          -5,029           0,000          12,098          54,56%            42,0               OK
    3             105             100            44,9          -5,029           0,000           6,717          61,50%            44,9               OK
    4             104             100            56,4          -5,026           0,000           6,717          61,50%            44,9               OK
Layer number: 1     Thickness: 100     Status: Success         Reason: OptOscillation
Time Spent on layer: 00:13:55

    1             163             150            36,5          -2,710           0,000           4,788          75,37%            31,5               OK
    2             175             200            31,5          -2,324           0,000           7,468          71,07%            30,2               OK
    3             175             250            31,5          -2,324           0,000          10,366          67,58%            28,8               OK
    4             175             200            31,5          -2,324           0,000           7,468          71,07%            30,2               OK
    5             175             200            31,5          -2,324           0,000           7,468          71,07%            30,2               OK
Layer number: 2     Thickness: 200     Status: Success         Reason: OptOscillation
Time Spent on layer: 00:20:10

    1             263             250            24,4          -1,124           0,000           4,535          82,75%            20,0               OK
    2             291             300            20,0          -0,884           0,000           6,155          80,34%            19,6               OK
    3             291             350            20,0          -0,884           0,000           7,919          78,35%            19,3               OK
    4             291             400            20,0          -0,884           0,000           9,789          76,72%            19,1               OK
    5             291             450            20,0          -0,884           0,000          11,907          75,46%            19,0               OK
    6             291             400            20,0          -0,884           0,000           9,789          76,72%            19,1               OK
    7             291             400            20,0          -0,884           0,000           9,789          76,72%            19,1               OK
Layer number: 3     Thickness: 400     Status: Success         Reason: OptOscillation
Time Spent on layer: 00:36:33

    1             412             400            17,5          -0,462           0,011           6,274          86,43%            16,8               OK
    2             399             450            16,8          -0,486           0,009           7,423          84,79%            17,0               OK
    3             399             500            16,8          -0,486           0,009           8,806          82,84%            17,6               OK
    4             399             550            16,8          -0,486           0,009          12,212          80,75%            18,4               OK
    5             399             500            16,8          -0,486           0,009           8,806          82,84%            17,6               OK
    6             399             500            16,8          -0,486           0,009           8,806          82,84%            17,6               OK
Layer number: 4     Thickness: 500     Status: Success         Reason: OptOscillation
Time Spent on layer: 00:34:26

    1             267             250            24,3          -0,327           1,016           4,143          87,40%            30,2               OK
    2             258             300            30,2          -0,444           1,044           4,837          85,07%            30,6               OK
    3             258             350            30,2          -0,444           1,044           5,683          82,85%            30,9               OK
    4             258             400            30,2          -0,444           1,044           6,376          80,74%            31,1               OK
    5             258             450            30,2          -0,444           1,044           7,565          78,75%            31,2               OK
    6             258             500            30,2          -0,444           1,044           8,897          76,[illegible]    31,2               OK
    7             258             550            30,2          -0,444           1,044          10,097          75,14%            31,1               OK
    8             258             500            30,2          -0,444           1,044           8,897          76,38%            31,2               OK
    9             258             500            30,2          -0,444           1,044           8,897          76,38%            31,2               OK
Layer number: 5     Thickness: 500     Status: Success         Reason: OptOscillation
Time Spent on layer: 00:41:24

    1             198             150            31,6          -1,942           0,079           3,521          93,75%            30,2               OK
    2             287             250            30,2          -0,822           0,014        Infinity          90,47%            30,7               OK
    3             287             200            30,2          -0,822           0,014           6,859          92,07%            28,6               OK
    4             287             250            30,2          -0,822           0,014        Infinity          90,47%            30,7               OK
    5             287             200            30,2          -0,822           0,014           6,859          92,07%            28,6               OK
Layer number: 6     Thickness: 200     Status: Success         Reason: OptOscillation
Time Spent on layer: 00:28:18
```

Figure F.1: First segment iterations

Slicing started at: 3-9-2007 12:37:40

Corr. pt. characteristics:
Tangent vertices allowed: True
Min. angle: 5,00 [deg]

Process constraints:
Max allowed wire angle: 55,00 [deg]

Accuracy constraints:
Max allowed chordal error: 10 [mm]
Min allowed contour ratio: 50,00 %

Sampling settings:
Number of samples per contour: 50
Number of z-error stages: 2

Stop error-eval when exceeds: False
EPS Sheets available in DB: 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 1000

After Renumbering:

ID	T_est [mm]	T_DB [mm]	AngMaxIni [deg]	kMaxConvex[1/m]	kMaxConcav[1/m]	ErrMaxAbs [mm]	ArcMin [%]	AngMax [deg]		PreProcRes
1	138	100	23,5	-3,809	0,385	5,247	97,10%	18,9		OK
2	141	150	18,9	-3,642	0,370	8,112	96,92%	15,2		OK
3	141	200	18,9	-3,642	0,370	11,676	96,54%	12,6	**Layer 7**	OK
4	141	150	18,9	-3,642	0,370	8,112	96,92%	15,2		OK
5	141	150	18,9	-3,642	0,370	8,112	96,92%	15,2		OK

Layer number: 1 Thickness: 150 Status: Success Reason: OptOscillation
Time Spent on layer: 00:39:16

ID	T_est [mm]	T_DB [mm]	AngMaxIni [deg]	kMaxConvex[1/m]	kMaxConcav[1/m]	ErrMaxAbs [mm]	ArcMin [%]	AngMax [deg]		PreProcRes
1	241	200	13,8	-1,153	0,000	3,692	96,75%	18,4		OK
2	238	250	18,4	-1,187	0,000	5,606	95,58%	19,0		OK
3	238	300	18,4	-1,187	0,000	7,947	94,50%	19,5	**Layer 8**	OK
4	238	350	18,4	-1,187	0,000	10,682	93,23%	20,0		OK
5	238	300	18,4	-1,187	0,000	7,947	94,50%	19,5		OK
6	238	300	18,4	-1,187	0,000	7,947	94,50%	19,5		OK

Layer number: 2 Thickness: 300 Status: Success Reason: OptOscillation
Time Spent on layer: 00:31:15

ID	T_est [mm]	T_DB [mm]	AngMaxIni [deg]	kMaxConvex[1/m]	kMaxConcav[1/m]	ErrMaxAbs [mm]	ArcMin [%]	AngMax [deg]		PreProcRes
1	329	300	22,5	-0,668	0,000	8,265	90,60%	25,2		OK
2	310	350	25,2	-0,663	0,000	11,534	88,77%	25,7	**Layer 9**	OK
3	310	300	25,2	-0,663	0,000	8,265	90,60%	25,2		OK
4	329	300	22,5	-0,668	0,000	8,265	90,60%	25,2		OK

Layer number: 3 Thickness: 300 Status: Success Reason: OptOscillation
Time Spent on layer: 00:14:37

Figure F.2: Second segment iterations

Slicing started at: 3-9-2007 14:17:21

Corr. pt. characteristics:
Tangent vertices allowed: True
Min. angle: 5,00 [deg]

Process constraints:
Max allowed wire angle: 55,00 [deg]

Accuracy constraints:
Max allowed chordal error: 10 [mm]
Min allowed contour ratio: 50,00 %

Sampling settings:
Number of samples per contour: 50
Number of z-error stages: 2

Stop error-eval when exceeds: False
EPS Sheets available in DB: 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 1000

After Renumbering:

ID	T_est [mm]	T_DB [mm]	AngMaxIni [deg]	kMaxConvex[1/m]	kMaxConcav[1/m]	ErrMaxAbs [mm]	ArcMin [%]	AngMax [deg]		PreProcRes
1	243	200	32,8	-0,888	1,268	5,368	91,44%	33,4		OK
2	251	250	33,4	-1,039	1,163	8,011	89,63%	32,6		OK
3	251	300	33,4	-1,039	1,163	11,029	87,96%	31,8		OK
4	251	250	33,4	-1,039	1,163	8,011	89,63%	32,6		OK
5	251	250	33,4	-1,039	1,163	8,011	89,63%	32,6	**Layr 10**	OK

Layer number: 1 Thickness: 250 Status: Success Reason: OptOscillation
Time Spent on layer: 00:19:38

(will have cusom thickness)

Figure F.3: Third segment - reverse direction

```
Slicing started at:           3-9-2007 15:06:06

Corr. pt. characteristics:
Tangent vertices allowed:     True
Min. angle:                   5,00 [deg]

Process constraints:
Max allowed wire angle:       55,00 [deg]

Accuracy constraints:
Max allowed chordal error:    10 [mm]
Min allowed contour ratio:    50,00 %

Sampling settings:
Number of samples per contour: 50
Number of z-error stages:     2

Stop error-eval when exceeds: False                                                                          After Renumbering:
EPS Sheets available in DB:   50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 1000

  ID     T_est [mm]      T_DB [mm] AngMaxIni [deg] kMaxConvex[1/m] kMaxConcav[1/m]  ErrMaxAbs [mm]    ArcMin [%]   AngMax [deg]      PreProcRes

   1          101            100          32,7          -3,023          4,954          4,538         78,31%          35,7                  OK
   2          101            150          35,7          -2,605          4,945          5,939         66,52%          36,9                  OK
   3          101            200          35,7          -2,605          4,945          6,573         57,51%          37,8                  OK
   4          101            250          35,7          -2,605          4,945          8,264         50,76%          38,7     Layer 11     OK
   5          101            300          35,7          -2,605          4,945         11,336         45,12%          39,8                  OK
   6          101            250          35,7          -2,605          4,945          8,264         50,76%          38,7                  OK
   7          101            250          35,7          -2,605          4,945          8,264         50,76%          38,7                  OK
Layer number: 1    Thickness: 250    Status: Success       Reason: OptOscillation
Time Spent on layer: 00:35:49

   1          180            150          43,1          -2,083          0,264          7,105         74,70%          40,5                  OK
   2          244            200          40,5          -0,611          0,497         12,484         69,53%          37,9     Layer 12     OK
   3          244            150          40,5          -0,611          0,497          7,105         74,70%          40,5                  OK
   4          180            150          43,1          -2,08           0,264          7,105         74,70%          40,5                  OK
Layer number: 2    Thickness: 150    Status: Success       Reason: OptOscillation
Time Spent on layer: 00:17:41

   1          158            150          31,5          -0,66           1,735          5,193         72,52%          26,3                  OK
   2          162            200          26,3          -0,60           1,658          9,051         64,65%          27,1                  OK
   3          162            250          26,3          -0,60           1,658         13,885         57,77%          28,0     Layer 13     OK
   4          162            200          26,3          -0,60           1,658          9,051         64,65%          27,1                  OK
   5          162            200          26,3          -0,60           1,658          9,051         64,65%          27,1                  OK
Layer number: 3    Thickness: 200    Status: Success       Reason: OptOscillation
Time Spent on layer: 00:23:16

   1          188            150          31,1          -0,77           1,684       Infinity        100,00%           0,0                  OK
   2          188            100          31,1          -0,77           1,684          2,969         78,76%          36,4                  OK
   3          206            200          36,4          -0,81           1,596       Infinity        100,00%           0,0                  OK
   4          206            150          36,4          -0,81           1,596       Infinity        100,00%           0,0                  OK
   5          188            100          31,1          -0,77           1,684          2,969         78,76%          36,4       Layer 14   OK
Layer number: 4    Thickness: 100    Status: Success       Reason: OptOscillation
Time Spent on layer: 00:11:11                                                            (will have custom thickness)

   1           60             50          41,9          -5,05           2,840       Infinity        100,00%           0,0                  OK
   2            0              0           0,0           0,00           0,000          0,000          0,00%           0,0        ErrNoGoodInDB
   3            0              0           0,0           0,00           0,000          0,000          0,00%           0,0              Unknown
Layer number: 5    Thickness: 0      Status: Failure       Reason: ErrNoSuitableInDB
Time Spent on layer: 00:08:57

Slicing finished at: 3-9-2007 16:43:53
Total time spent: 01:37:47
```

Figure F.4: Third segment iterations

Appendix G

Accuracy analysis

This appendix introduces photogrammetry (PG) as dimensional measurement technique. It is followed by the measurement setup and photogrammetric measurements, starting with measurement of planarity of a granite measurement table which is used to obtain a measure of precision of the measurement method. Next, a simple 2D cut foam object is measured for accuracy in order to obtain a measure of accuracy for the HWC process. Finally, a more complex assembly of multiple layers is measured and analyzed.

G.1 Introduction to Photogrammetry

Photogrammetry is a remote sensing technology which allows, amongst others, to determine the 3D coordinates of points which are visible on multiple photographs. It can be regarded as inverse photography, although this would be a too narrow description. [37] proposes a more complete schema which is shown in figure G.1. When sufficient photos are used for input, photogrammetric methods are able to

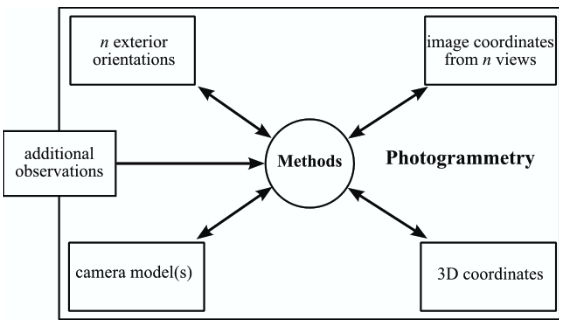


Figure G.1: Schematic view of photogrammetry

determine both the camera positions and orientations, denoted by the 'exterior orientation', as well as the 3D point coordinates and even lens distortions.

The main advantages of photogrammetry over other measurement methods are:

- relatively low cost,
- non-contact,
- simple measurement equipment: high-quality consumer photo camera,
- easily replaceable equipment,
- portability, on-site measurements,
- remote data processing possible.

The precision of this measurement method depends on factors such as the camera resolution, object size, amount of photos and point marking precision on the photos. In practice, a measurement precision of about 20 μm/m is feasible when using targets. According to [27], the technique is generally more accurate than laser scanning. Targets can be stickers with circular dots, which are fixed to the object and appear as ellipses in the photographs. See figure G.3(a) for an example. The center of the ellipses can then easily be determined by image processing at sub-pixel accuracy. When using coded targets, the software can automatically reference the same point across different photographs, which should otherwise be done manually. The measurement accuracy strongly depends on for example the accuracy of applying the targets to the object and compensation of target paper thickness.

G.2 Photogrammetry measurement setup

The measurement equipment as used throughout this work is presented in table G.1.

The camera calibration parameters used throughout the measurements is presented in table G.2. The σ-values denote the significance of the calibration parameters and should be at least an order smaller than the corresponding parameter values.

The accuracy analysis process is shown in figure G.2. The chart is not meant to be exhaustive. Note that the output from the PG software contains precision information about the measurement. The same applies to alignment. In figure G.2 and in the rest of this work, RMS denotes the 'root mean square' and σ the

Photogrammetry equipment	
camera:	Nikon D200
lens:	AF Nikkor 28 mm 1:2.8D
software	Photomodeler 6.0 with coded targets module
targets	Photomodeler targets, printed on plain paper
calibration sheet	Photomodeler calibration sheet (1 x 1 m)
scale bar	aluminium rule or retractable flexible rule

Table G.1: Photogrammetry equipment

standard deviation. Calculation of the RMS is expressed by

$$x_{RMS} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}. \tag{G.1}$$

It can be easily shown that

$$x_{RMS} = \bar{x}^2 + \sigma_x^2. \tag{G.2}$$

Nikon D200 camera with 28 mm lens Focus ring manually set to ∞		
Parameter	units: mm	σ
Focal Length	29.210749	5.5e-004
Xp princ. pt. X	11.829904	9.5e-004
Yp princ. pt. Y	8.138115	0.001
Fw format width	24.002822	2.2e-004
Fh format height	16.066116	n/a
	units: -	σ
K1 - radial distortion 1	1.449e-004	1.7e-007
K2 - radial distortion 2	-1.551e-007	9.7e-010
K3 - radial distortion 3	0.000e+000	n/a
P1 - decentering distortion 1	2.764e-005	3.2e-007
P2 - decentering distortion 2	-1.424e-005	3.5e-007

Table G.2: Photogrammetry calibration parameters

G.3 Photogrammetry accuracy

Accuracy denotes the difference between measurements and the 'true' value. Precision indicates the spread of the repeated measurement. Photogrammetry offers precision information about the obtained 3D point coordinates, obtained from the redundancy of image data. However, no information regarding accuracy is available.

In order to get a sense of the accuracy, a flat granite measurement block as shown in figure G.3(a) was marked, measured and processed using photogrammetry. In this experiment, 25 coded targets are used on the block sized 630 x 400 x 80 mm. Two targets were located at a predefined distance from each other in order to provide a scale for the point cloud. Eight photos were taken from different angles. After obtaining the 3D coordinates of the points using Photomodeler as shown in figure G.3(b), a plane was fitted through the points using a least-squares optimization. The resulting deviations from the plane are plotted in figure G.4. Numerical results are presented in table G.3.

In table G.3 The 'reference block tolerance' is a value indicating that all points on the block surface should lie between two parallel planes, separated by the tolerance value. Here, the measured tolerance exceeds the reference block tolerance. Figure G.4 shows a trend of the deviation, which seems to suggests that the block surface is slightly cylindrical in shape. However, the variation of the paper, ink and glue spray thickness may very well be in the order of 10 - 50 μm. The spray glue, for example, was applied manually, introducing the human factor. The exact

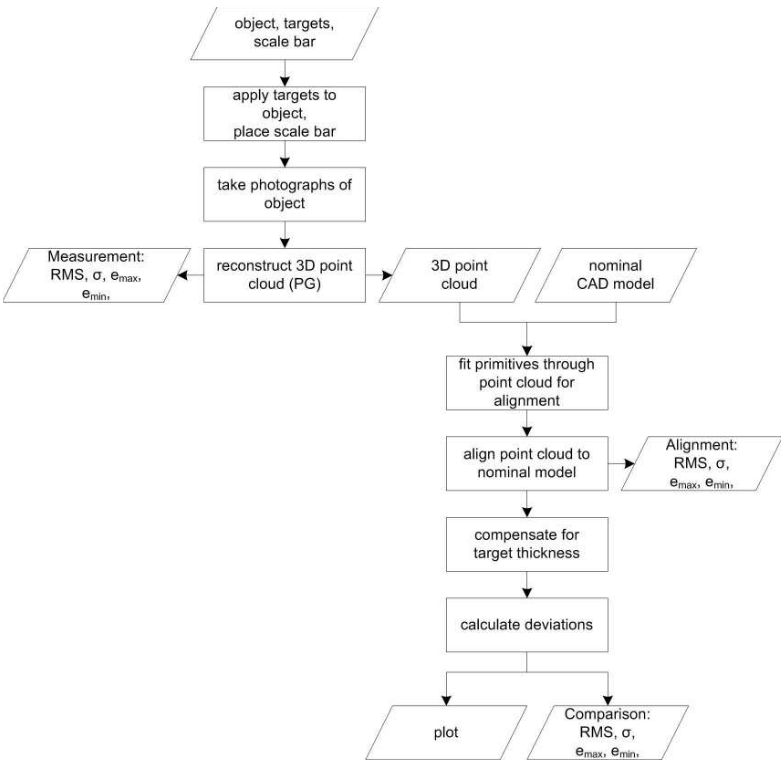
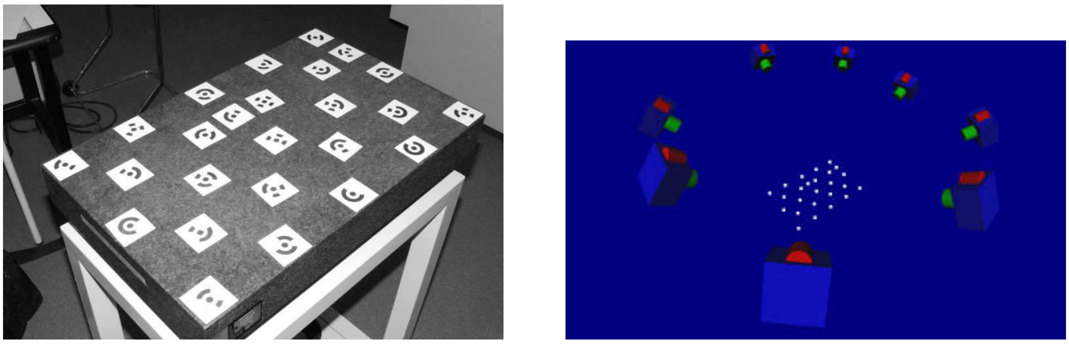


Figure G.2: Flow chart of accuracy analysis without details.



(a) Single photograph from a set of 8

(b) Calculated 3D point coordinates and camera stations

Figure G.3: Photogrammetry data and results

cause of the deviation is hard to define, based on this information alone. More experiments are needed in order to draw valid conclusions. However, the tolerance field does not deviate by unexplainable amounts, which builds some trust in PG.

Block measurement data	
	units: μm
reference block tolerance (year of check: 1994)	8.3
3D point precisions	
overall RMS vector length	10.8
maximum vector length	11.4
minimum vector length	10.3
Plane fitting results	
mean	0
RMS error	8.9
maximum error e_{max}	11.7
minimum error e_{min}	-23.0
measured tolerance e_{max} - e_{min}	34.7
points within +/- 1 σ	80%
points within +/- 2 σ	92%
points within +/- 3 σ	100%

Table G.3: Measurement and plane-fitting data from granite table

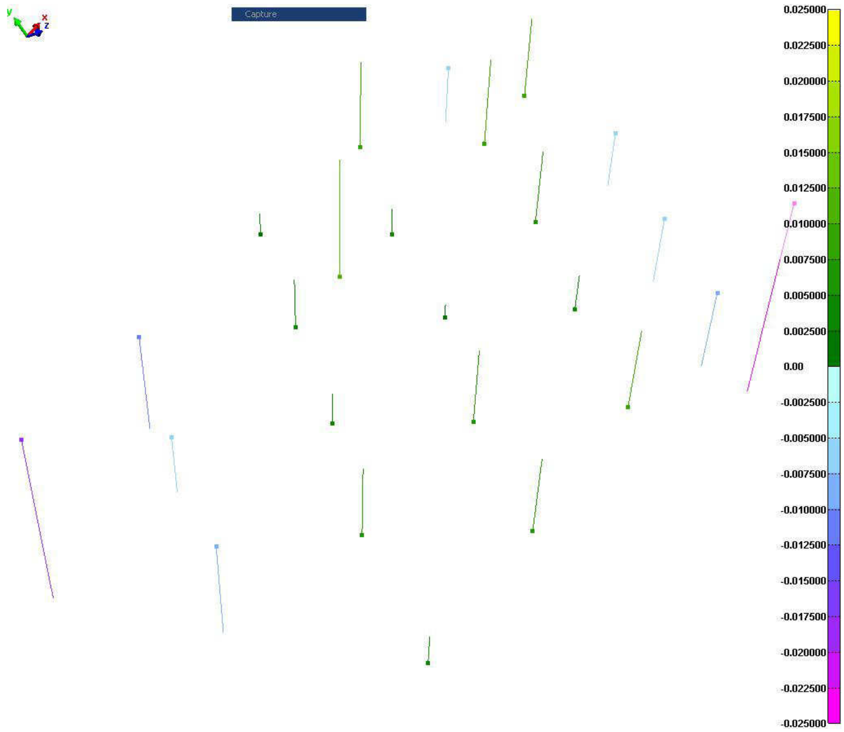


Figure G.4: Deviation of measured points from fitted plane. Perspective is about the same as in figure G.3(a)

G.4 Single layer accuracy analysis

In order to obtain an indication of fabrication accuracy, a simple 2D part of 1000 x 500 x 200 mm was cut. It is depicted in figure G.5(a). The part only contains planar faces except for a top face resembling an airfoil. A target tape is fixed all round the product and used for accuracy inspection. The resulting 3D point cloud is shown in G.5(b).

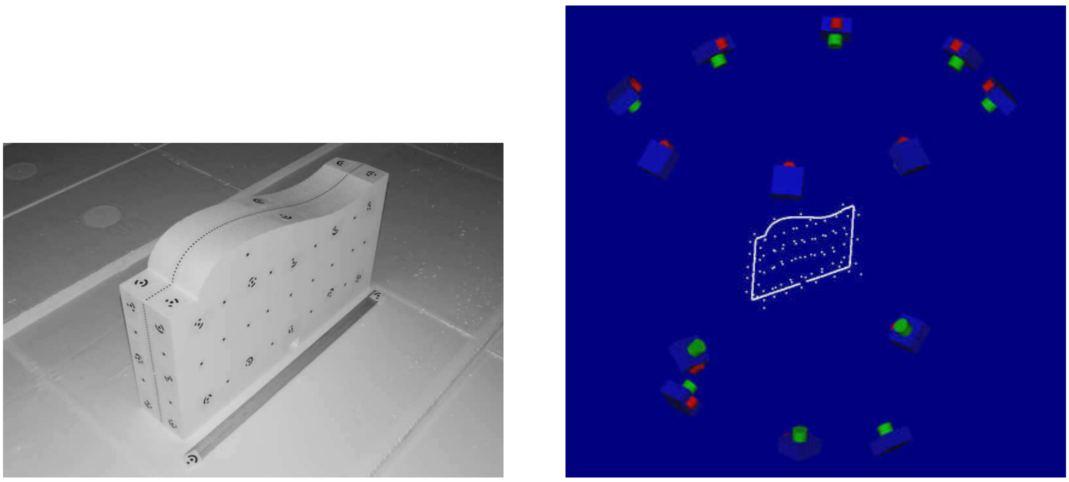
As can be seen in in figure G.5(a), each face of the foam model was marked with multiple targets. Through each of those sets of targets, a plane was fit in order to properly align the point cloud to the nominal model. After alignment, compensation for target thickness t_{tar} at each measured point $\vec{x}_i$ was approximated by

$$\vec{x}_{i,new} = \vec{x}_{i,old} - t_{tar}\vec{n}_{closest}, \tag{G.3}$$

where $\vec{n}_{closest}$ is the normal of the closest nominal surface point near $\vec{x}_{i,old}$.

The 'Shortest distance' is used as measure of accuracy. For each point in the point cloud, the distance to the nominal shape is calculated. After comparison, the result is as graphically shown in figure G.6 and numerically in table G.4. The fitted planes have RMS values around 0.05 mm, although the base plane (approximating to the ground contact face of the object in figure G.5(a)) had a RMS of 0.19 mm, which is relatively large. This could be the cause for a rotational misalignment which might show in figure G.6.

based on these observations, the manufacturing accuracy can be defined as the RMS value of the closest distances from point cloud to nominal model, which has a value in the order of of 0.3 mm. It might be tempting to state that this is an accuracy *per meter*, but this has not yet been shown. However, it could be used as a rule of thumb as it can be regarded as conservative. Repeating or extending this kind of experiment can make statements about accuracy more reliable.



(a) Single photograph from a set of 13

(b) Calculated 3D point coordinates and camera stations

Figure G.5: Photogrammetry data and results

3D point precisions	
	units: μm
overall RMS vector length	24.2
maximum vector length	35.6
minimum vector length	20.1
Comparison results for the target tape	
	units: mm
mean	-0.07
RMS error	0.29
σ	0.28
maximum error e_{max}	0.42
minimum error e_{min}	-0.64
points within +/- 1 σ	66.3%
points within +/- 2 σ	99.7%
points within +/- 3 σ	100%

Table G.4: Measurement and inspection data for single foam part

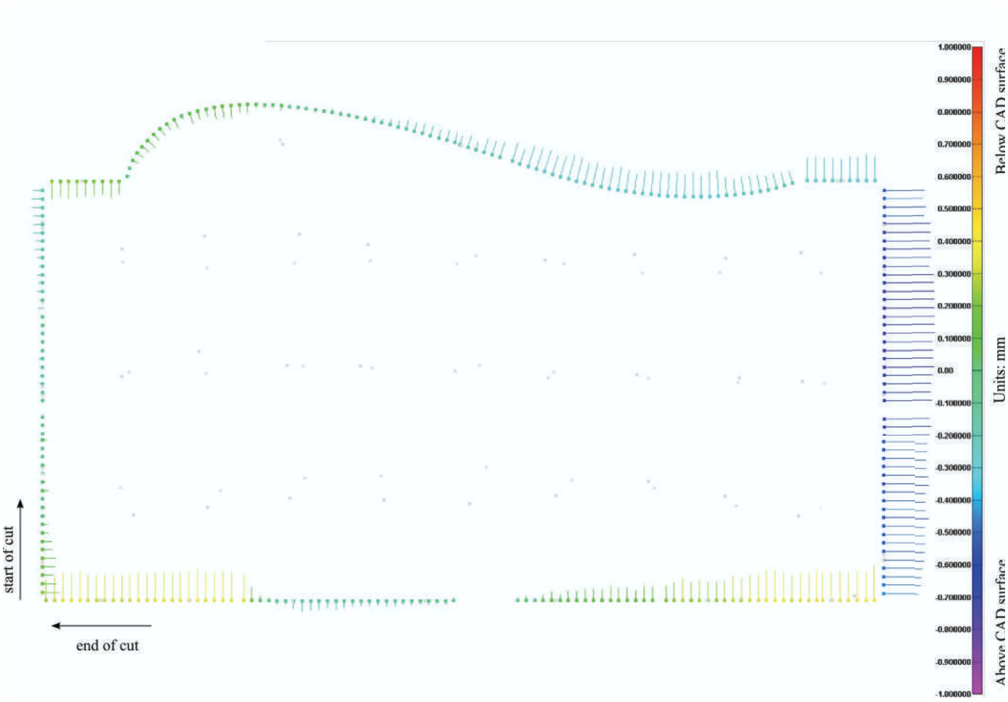
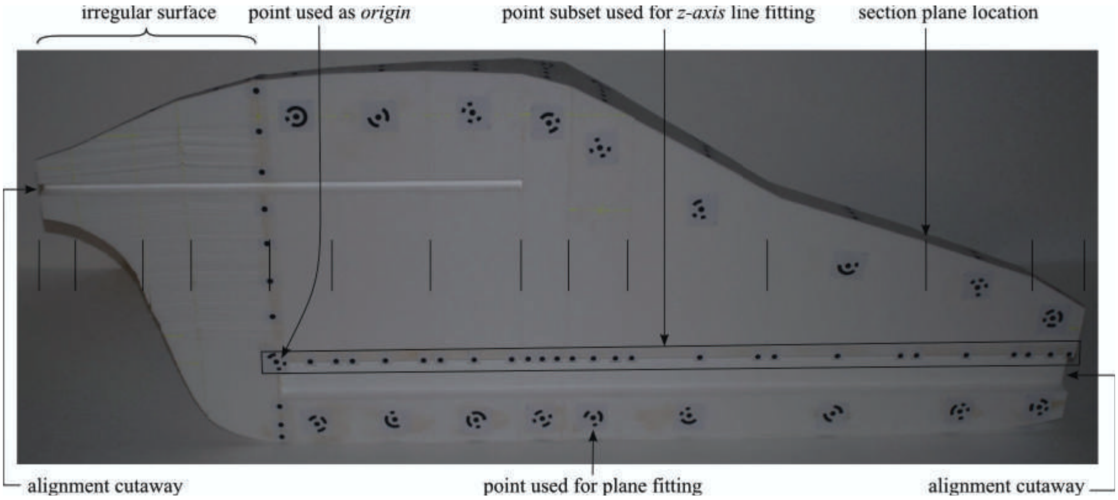


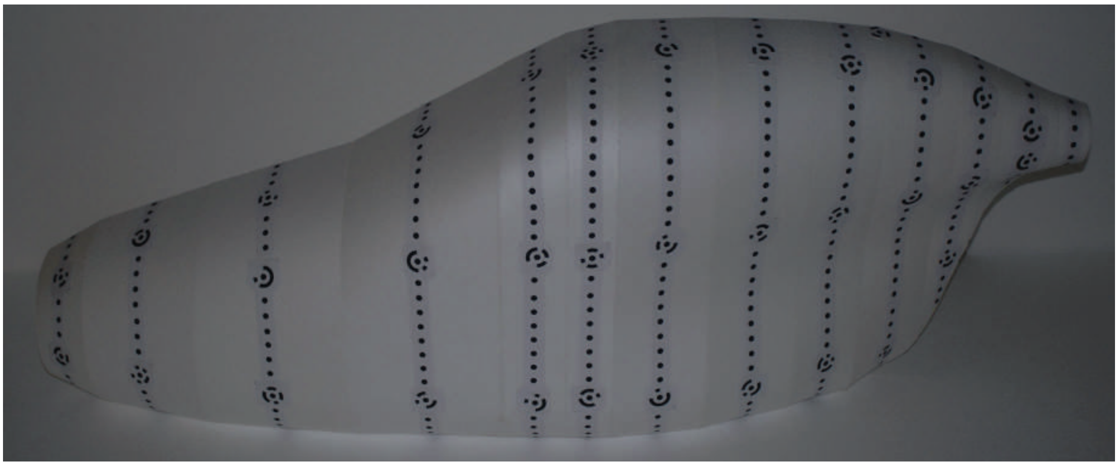
Figure G.6: Exaggerated plot of shortest distance from measurement points to nominal model

G.5 Layered model accuracy analysis

The accuracy for a layered assembly is discussed next. The demonstrator of this work is used for the measurement. The location of the targets on the port side of the body (figure G.7(b)) was chosen to be half way between each section plane because that is the location where the actual cusp height deviation is likely to be highest, as became clear in figure 5.7. At the starboard side of the body, one target was used to define the origin of the point cloud. The starboard-sided coded targets, except for the origin target, are used to later fit a plane. The uncoded targets, boxed in the figure, are used to fit a line. Note that some targets remained unused but were applied anyhow to be on the safe side. In the next phase, these primitives are used to align the point cloud to its ruled surface CAD equivalent.



(a) Starboard side

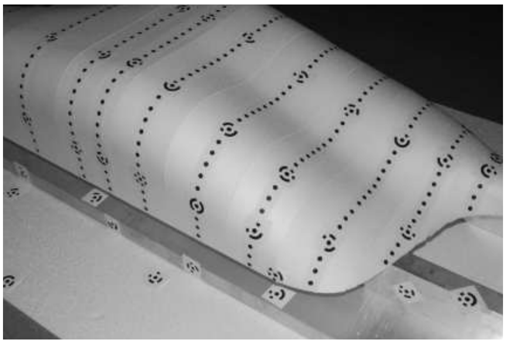


(b) Port side

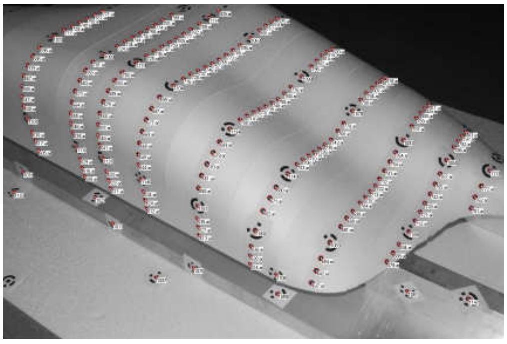
Figure G.7: Targets used for point cloud generation

After generation of the point cloud with photogrammetry, the cloud is aligned to the exact ruled surface CAD model from figure 5.3(b). The alignment procedure is the following. First, an origin, plane and vector are fitted through the intended points of figure G.7(a). Note that the tail region has an irregular surface quality. It was caused by the wire cutting *through* regions of excessively applied glue during the chop-off of the tail extension. It was therefore not used for targeting. The result is shown in figure G.10(b). The redundancy used to create the primitives also offers information about the precision of the fit, as shown in table G.5

The resulting primitives are coupled to the point cloud when moving. The



(a) Unmarked photo



(b) Marked photo

Figure G.8: Example photo of the body with both coded and uncoded targets

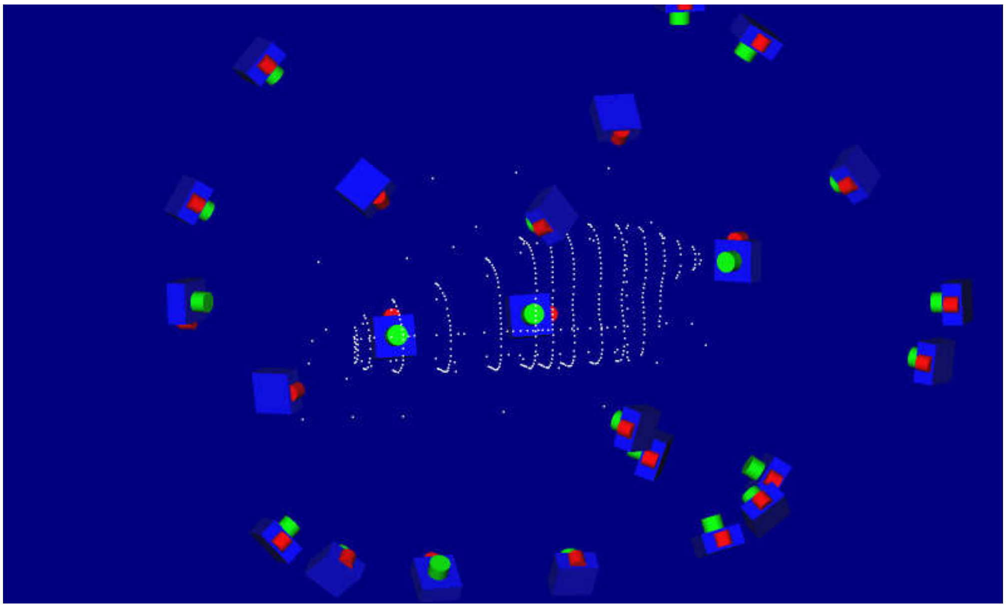
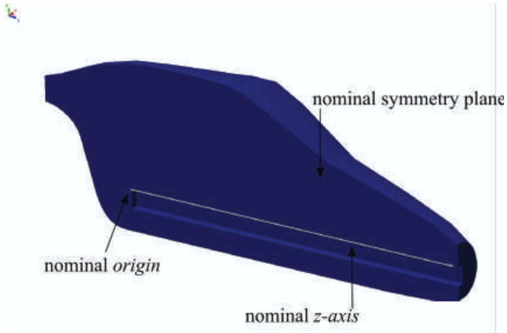


Figure G.9: 3D Point cloud generated by photogrammetry software

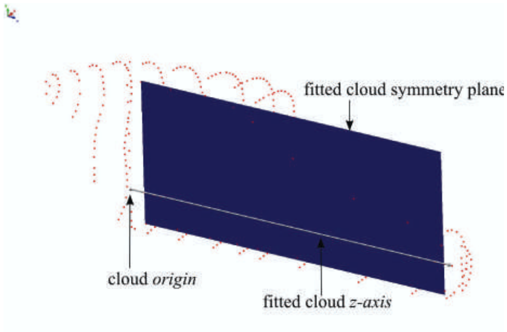
Fit and deviation results units: mm					
	RMS	**mean**	σ	**max. error**	**min. error**
plane fit	0.49	0.00	0.51	0.77	-0.83
vector fit	0.76	0.69	0.34	1.29	0.19
origin alignment	1 (est.)				

Table G.5: Measurement and inspection data for layered object

nominal model (figure G.10(a)) is fixed in space. Next, the fitted plane is made coincident with the reference symmetry plane. Using this alignment, one rotational and two translational degrees of freedom are left to be constrained. Next, the point cloud is rotated so the fitted vector becomes coplanar to the nominal z-axis, constraining the rotational and one translational d.o.f. Finally, the cloud is



(a) Primitives extracted from the nominal ruled surface CAD model



(b) Primitives fitted to the point cloud

Figure G.10: Example photo of the body with both coded and uncoded targets

translated to the point where the cloud origin and the nominal origin are closest to each other, constraining the last translational d.o.f. Now, all d.o.f.'s have been constrained. Target thickness compensation was applied using the same approximation from the previous section.

Next, the closest point deviation is calculated. The result is shown in figure G.11. In this figure, the 'as manufactured' part is compared to the 'as designed' part, showing the normal deviations of the port side targets from the ruled surface. Numerical results are shown in table G.6. The maximum error is more than 5 mm and is clearly visible in figure G.11. The cause of this error turned out to be a local gross human error in establishing the proper correspondence between tho contours after DXF export. In fact, a set of corresponding points was forgotten. Based on the RMS value, the accuracy for a layered object could be roughly set at 1 mm in the layer plane. When the gross error is left out, the maximum error changes from 5.18 mm to 2.22 mm. This results in a measured tolerance of 5.80 mm.

3D point precisions	
	units: μm
overall RMS vector length	26.4
maximum vector length	57.9
minimum vector length	17.4
Closest point comparison results for the target tapes	
	units: mm
mean	-0.36
RMS error	1.17
σ	1.12
maximum error e_{max}	5.18
minimum error e_{min}	-3.58
points within +/- 1 σ	72%
points within +/- 2 σ	95%
points within +/- 3 σ	99%

Table G.6: Measurement and inspection data for layered object

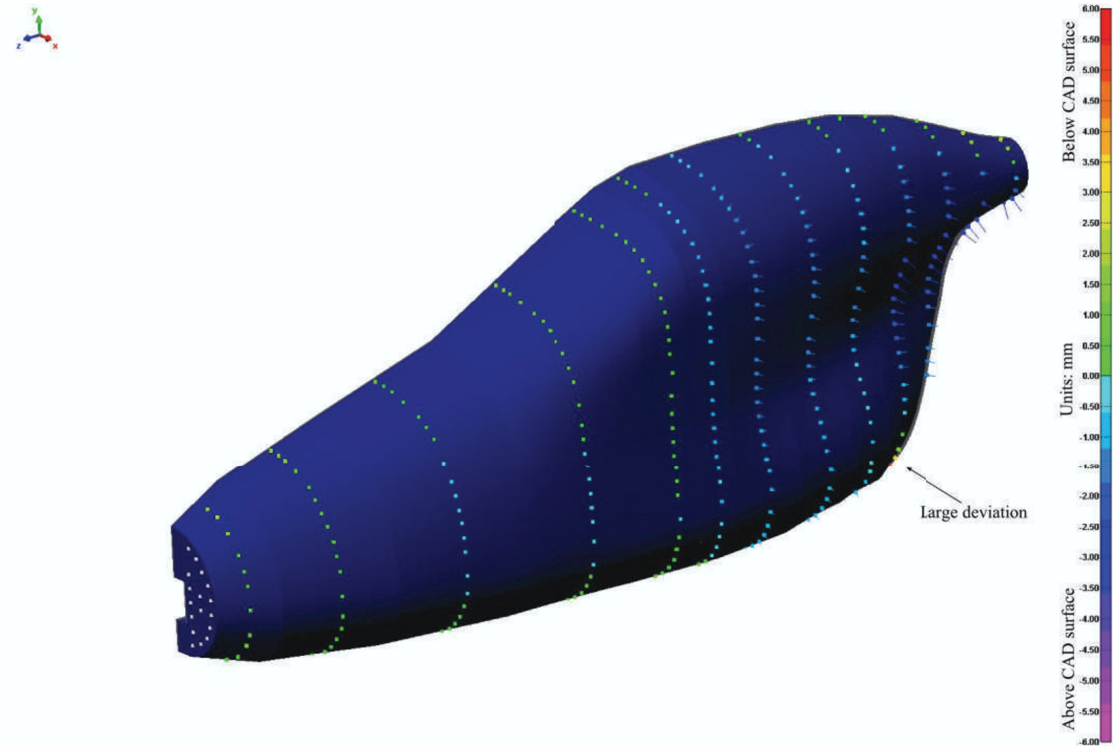
Due to effect like slight bending and warping of the layered object, proper alignment of the point cloud with the nominal object is difficult. In order to estimate the effect of misalignment, a misalignment sensitivity analysis was performed using 1 mm of alignment translation of the point cloud in positive and negative direction of roll, pitch and top axis. The magnitude of this value is based on the magnitudes of the fit precisions. The results are presented in table G.7. Variations in rotational sense and combinations of variations have been left out.

Alignment change results for 1 mm translation				
incr. along axis:	direction of change	Δ(**RMS**)	Δ(**mean**)	$\Delta\sigma$
				units: mm
roll	-	-0.10	0.63	-0.08
roll	+	0.46	-0.63	0.18
pitch	-	0.19	-0.06	0.18
pitch	+	-0.14	0.06	-0.13
top	-	0.26	-0.01	0.27
top	+	0.05	0.00	0.05

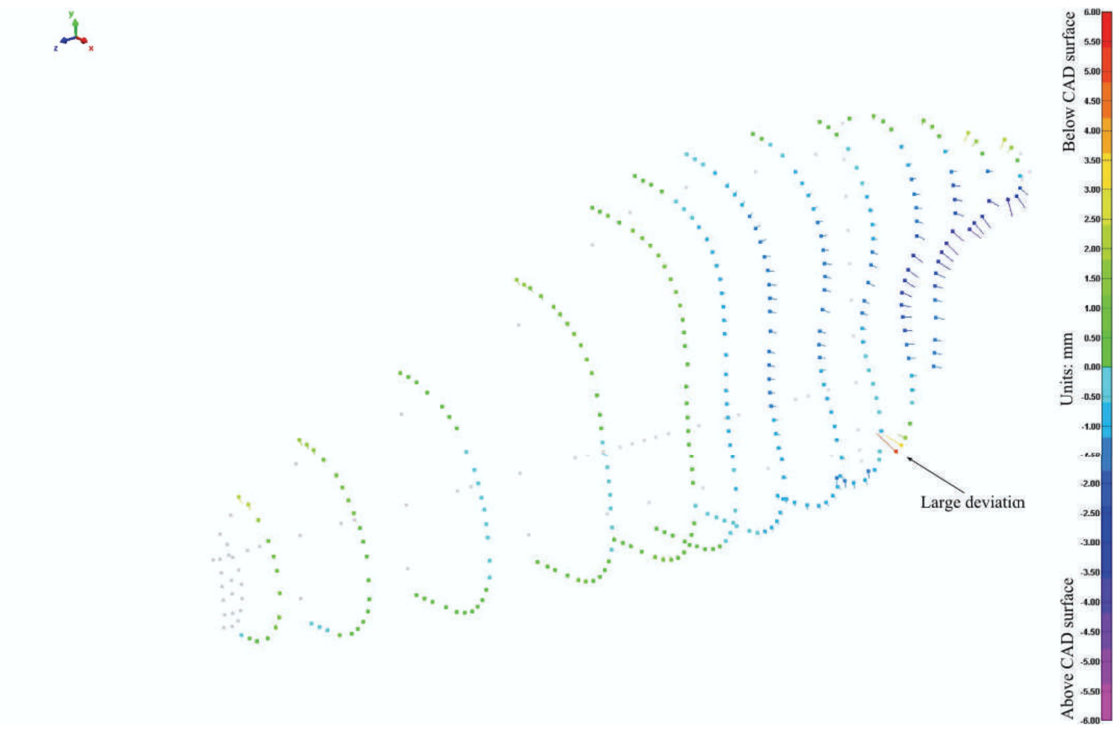
Table G.7: Alignment sensitivity analysis results

Because misalignment in the order of 1 mm is possible, The change in RMS is of particular interest. It could be conceived as a measure of alignment precision which, based on the results, has a value in the order of 0.5 mm.

The deviation in thickness direction was measured as well. Results are shown in G.8. These values strongly depend on the amount, distribution and type of adhesive used, as well as the amount and distribution of compression applied during curing. Also, cutting accuracy plays a role. In this case, a foaming glue was used. Non-homogeneous compression could lead to non-homogeneous expansion



(a) Normal deviation of the point cloud w.r.t the nominal ruled surface CAD model



(b) Normal deviation of the point cloud (without CAD model)

Figure G.11: Example photo of the body with both coded and uncoded targets

of the glue which could then tilt the layers with respect to each other. Based on these results, an accuracy in the order of 0.1 mm per glue film should be taken into account.

Thickness precision	units: mm
nominal length	1750
measured length	1748.5
difference	1.5
deviation per adhesive film (13 planes)	0.13

Table G.8: Thickness deviation from the layered object

Bibliography

[1] Ahn, D., Lee, S., and Yang, D. Cutting path generation of linear hotwire cutter for vlm-st. *The International Journal of Advanced Manufacturing Technology*, 30(5-6):401–415, 2006. ISSN 02683768.

[2] Banerjee, A.G., Kumar, A., Tejavath, S., and Choudhury, A.R. Adaptive slicing with curvature considerations. *International Journal of CAD/CAM*, 3(1):31–40, 2003.

[3] Berzin, D. and Hagiwara, I. Minimal area for surface reconstruction from cross sections. *Visual Comput*, 18(7):437–444, 2002. ISSN 01782789.

[4] Brink, J., Lee, A., Anderson, D., and Ramani, K. Cad model decomposition for wirepath. *Rapid Prototyping J*, 10(5):288–296, 2004. ISSN 13552546.

[5] Broek, J., Horváth, I., de Smit, B., Lennings, A., Rusák, Z., and Vergeest, J. Free-form thick layer object manufacturing technology for large-sized physical models. *Autom Constr*, 11(3):335–347, 2002. ISSN 09265805.

[6] Broek, J., Horváth, I., de Smit, B., Lennings, A.F., and Vergeest, J.S. A survey of the state of art in thick layered manufacturing of large objects and the presentation of a newly developed system. In *First International Seminar and Workshop on Engineering Design in Integrated ProductDevelopment - EDIProD'98*, pages IV–1–IV–12. Delft University of Technology Jaffalaan 9, NL-2628 BX DELFT, The Netherlands, 1998.

[7] Cohen, S., Elber, G., and Bar-Yehuda, R. Matching of freeform curves. *Computer Aided Design*, 29(5):369–378, 1997. ISSN 00104485.

[8] de Jager, P., Broek, J., , and Vergeest, J. A comparison between zero and first order approximation algorithms for layered manufacturing. *Rapid Prototyping J*, 3(4):144–149, 1997. ISSN 13552546.

[9] Faux, I.D. and Pratt, M.J. *Computational geometry for design and manufacture*. Chichester : Ellis Horwood [etc.], 1979.

[10] Fuchs, H., Kedem, Z.M., and Uselton, S.P. Optimal surface reconstruction from planar contours. *Commun. ACM*, 20(10):693–702, 1977. ISSN 0001-0782. doi:http://doi.acm.org/10.1145/359842.359846.

[11] Gabrielides, N., Ginnis, A., and Kaklis, P. Constructing smooth branching surfaces from cross sections. In *SPM 2006 - ACM Symposium on Solid*

and Physical Modeling, volume 2006, pages 161–170. Wales, 2006. ISSN 1595933581 (ISBN); 9781595933584 (ISBN).

[12] Giertsen, C., Halvorsen, A., and Flood, P.R. Graph-directed modelling from serial sections. *The Visual Computer*, V6(5):284–290, 1990.

[13] Hope, R., Jacobs, P., and Roth, R. Rapid prototyping with sloping surfaces. *Rapid Prototyping J*, 3(1):12–19, 1997. ISSN 13552546.

[14] Hope, R., Riek, A., and Roth, R. Layer building with sloping edges for rapid prototyping of large objects. In *Proceedings of the 5th European conference on rapid prototyping and manufacturing*, pages 47–57. Finland, 1996. Trusurf.

[15] Hope, R., Roth, R., and Jacobs, P. Adaptive slicing with sloping layer surfaces. *Rapid Prototyping J*, 3(3):89–98, 1997. ISSN 13552546.

[16] Kai, C., Jacob, G., and Mei, T. Interface between cad and rapid prototyping systems. part 2: Lmi - an improved interface. *The International Journal of Advanced Manufacturing Technology*, 13(8):571–576, 1997. ISSN 02683768.

[17] Kals, H., van Lutterveld, C., and Moulijn, K. *Industrile productie*. De Vey Mestdagh BV, 1996.

[18] Keppel, E. Approximating complex surfaces by triangulation of contour lines. *IBM Journal of Research and Development*, 19(1):2–11, 1975. ISSN 00188646.

[19] Kim, H.C. and Lee, S.H. Reduction of post-processing for stereolithography systems by fabrication-direction optimization. *Computer Aided Design*, 37(7):711–725, 2005. ISSN 00104485.

[20] Koc, B. Adaptive layer approximation of free-form models using marching point surface error calculation for rapid prototyping. *Rapid Prototyping J*, 10(5):270–280, 2004. ISSN 13552546.

[21] Kulkarni, P. and Dutta, D. An accurate slicing procedure for layered manufacturing. *Computer Aided Design*, 28(9):683–697, 1996. ISSN 00104485.

[22] Kumar, M. and Choudhury, A. Adaptive slicing with cubic patch approximation. *Rapid Prototyping J*, 8(4):224–232, 2002. ISSN 13552546.

[23] Lan, P.T., Chou, S.Y., Chen, L.L., and Gemmill, D. Determining fabrication orientations for rapid prototyping with stereolithography apparatus. *Computer Aided Design*, 29(1):53–62, 1997. ISSN 00104485.

[24] Lee, S., Ahn, D., and Yang, D. Calculation and verification of rotation angle of a four-axis hotwire cutter for transfer-type variable lamination manufacturing using expandable polystyrene foam. *The International Journal of Advanced Manufacturing Technology*, V22(3):175–183, 2003.

[25] Li, W., Xu, S., and Zhao, G. Manufacturable surface reconstruction from complex contours. *Rapid Prototyping J*, 10(2):114–122, 2004. ISSN 13552546.

[26] Lin, F., Sun, W., and Yan, Y. Optimization with minimum process error for layered manufacturing fabrication. *Rapid Prototyping J*, 7(2):73–81, 2001. ISSN 13552546.

[27] Luhman, T., Robson, S., Kyle, S., and Harley, I. *Close Range Photogrammetry*. Whittles Publishing, 2006.

[28] Mani, K., Kulkarni, P., and Dutta, D. Region-based adaptive slicing. *Computer Aided Design*, 31(5):317–333, 1999. ISSN 00104485.

[29] Pandey, P., Ready, N., and Dhande, S. Slicing procedures in layered manufacturing: A review. *Rapid Prototyping J*, 9(5):274–288, 2003. ISSN 13552546.

[30] Rock, S. and Wozny, M. A flexible file format for solid freeform fabrication. In H. Marcus, J.J.B. ad J.W. Barlow, D. Bourell, and R. Crawford, editors, *Solid Freeform Fabrication Symposium Proceedings*. The University of Texas at Austin, 1991.

[31] Rock, S. Generating topological information from a "bucket of facets". In e.a. H. L. Marcus, editor, *Solid Freeform Fabrication Symposium Proceedings*, pages 251–259. Rensselaer Design Research Center, The University of Texas, Rensselaer Polytechnic Institute Troy, New York 12180, 1992.

[32] Sederberg, T.W., Klimaszewski, K.S., and Hong, M. Triangulation of branching contours using area minimization. *Journal of Computational Geometry & Applications*, unknown:unknomn, 1996.

[33] Smith, T., Farouki, R., Al-Kandari, M., and Pottmann, H. Optimal slicing of free-form surfaces. *Comput Aided Geom Des*, 19(1):43–64, 2002. ISSN 01678396.

[34] Suh, Y. and Wozny, M. Adaptive slicing for solid freeform fabrication processes. In *Proceedings of the Solid Freeform Fabrication Symposium*. 1994. [Ref. 17].

[35] Thomas, C.L., Gaffney, T.M., Kaza, S., and Lee, C.H. Rapid prototyping of large scale aerospace structures. In *Proceedings of the 1996 IEEE Aerospace Applications Conference. Part 1 (of 4)*, volume 4, pages 219–229. IEEE, Snowmass, CO, USA, 1996.

[36] Unkown. www.step-four.at, 2007.

[37] Wiora, G. *Optische 3D-Messtechnik: Przise Gestaltvermessung mit einem erweiterten Streifenprojectionsverfahren*. Ph.D. thesis, Ruprechts-Karls universitt Heidelberg, 2001.

[38] Wu, T. and Cheung, E.H.M. Enhanced stl. *The International Journal of Advanced Manufacturing Technology*, V29(11):1143–1150, 2006.

[39] Yau, H., Kuo, C., and Yeh, C. Extension of surface reconstruction algorithm to the global stitching and repairing of stl models. *Computer Aided Design*, 35(5):477–486, 2003. ISSN 00104485.

[40] Zheng, Y. *ENABLING COMPUTATIONAL TECHNIQUES FOR TANGENTIAL-BUILDING SOLID FREEFORM FABRICATION*. Ph.D. thesis, CASE WESTERN RESERVE UNIVERSITY, 1997.