UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING AND THE ENVIRONMENT

Computational Engineering and Design Group

The use of automated process planning to minimise unit cost whilst retaining flexibility of manufacturing method

by

David Cooper MEng

Thesis for the degree of Engineering Doctorate

August 2016

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND THE ENVIRONMENT

Computational Engineering and Design Group

Thesis for the degree of Engineering Doctorate

THE USE OF AUTOMATED PROCESS PLANNING TO MINIMISE UNIT COST WHILST RETAINING FLEXIBILITY OF MANUFACTURING METHOD

by David Stephen Cooper

This research focussed on the automatic generation of the optimal method of manufacture, and the cost thereof, at an early stage in design. It used geometric and tolerance data, combined with a database of centrally stored manufacturing knowledge. This allowed the construction of potential manufacturing routes, and the evaluation of their cost.

In order to find the least costly method of manufacture for a given component, with defined geometry and tolerance, it is necessary to take a holistic view of the manufacturing process. This thesis shows that the typical sequential approach of choosing the manufacturing process, followed by sequencing, will not necessarily find the global optimum. Dynamic process seletion is required to consider both the choice of manufacturing processes and the sequence together to avoid sub-optimisation.

Backward and forward propagation from a final manufacturing process, using a random mutation hill climber, was used to construct potential manufacturing process sets. For each process set constructed, the inner optimisation loop was used to find the optimal sequence of manufacturing operations, with the aid of a repair operator to ensure feasibility.

Prior to finding the optimal manufacturing sequence, the forming shape for all potential forming processes was required; this research encompasses a methodology to automatically deduce this shape.

This methodology demonstrated convergence to the global optimum on a contrived test case, and has demonstrated accurate results on a real commercial test case. Major areas for improvement could be a combination of the developed methodology with feature recognition, to enable maximum usefulness as a fully automated process planning and cost evaluation tool.

In summary this research demonstrates that the lack of a holistic view can cause sub-optimal results, and proposes a methodology which provides a holistic perspective. The methodology has been proved to be accurate, generic and expandable.

Contents

List of Figures

List of Tables

Academic Thesis: Declaration Of Authorship

I, *DAVID STEPHEN COOPER*

declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

Title of thesis: ***The use of automated process planning to minimise unit cost whilst retaining flexibility of manufacturing method***

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;

2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;

3. Where I have consulted the published work of others, this is always clearly attributed;

4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;

5. I have acknowledged all main sources of help;

6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

7. Either none of this work has been published before submission, or parts of this work have been published as: [please list references below]:

The methodology FoSCo has been submitted for patent in the USA and the EU by Rolls-Royce plc.

EU patent number 16180366.3 - 1954

US patent submission reference 5895 LMT

Signed:
. . . .

Date:
. . . .

Acknowledgements

The author is grateful for the support of his academic supervisors and internal examiner, Prof James Scanlan, Dr. Robert Marsh and Dr David Toal, which was invaluable in aiding in the formulation of ideas, and giving critical assessment.

The support from his industrial supervisor, Steve Wiseall, and other Rolls-Royce contacts: Craig Johnson, Ralph Boyce and Dave Reuss and many other Rolls-Royce employees, who supplied the data required for the modelling and construction of the database on which the analysis relies, is also greatly appreciated.

Without their assistance the author would have been unable to conduct the necessary investigations and so complete this report.

Author's Declaration

This report is submitted to the University of Southampton in support of my work towards an Engineering Doctorate. All non-original work has been appropriately referenced.

Nomenclature

CAPP Computer Aided Process Planning

CE Concurrent Engineering

CIS Component Interface Spreadsheet

ET Elite Tracking

FPG Finished Part Geometry

G&T Geometric and Tolerance

GA Genetic Algorithm

RMHC Random Mutation Hill Climber

SA Simulated Annealing

TAD Tool Access Direction

Nomenclature specific to sections 4 and 5 are contained at the beginning of these sections.

1 Introduction

This section introduces the rationale behind the research conducted, it proposes a hypothesis to be proved or disproved, and an objective to be fulfilled. It then goes on to introduce the proposed use of the research in industry, the test cases on which it will be used for validation and verification, and finally it will summarise the structure of this thesis.

The investigation will centre on the deduction of unit costs for components for which the method of manufacture is not fixed, it will focus therefore on creating and cost modelling a set of manufacturing methods. This requires appropriate objective functions for cost modelling purposes, and an effective optimiser to perfect the method of manufacture.

1.1 Research Introduction

It is well accepted that the reduction of unit cost (the cost of manufacturing one item), of a component or assembly, is a desirable objective, provided it is achieved without a corresponding loss in capability.

It is also widely believed that the earlier knowledge (such as performance, weight and cost expectations) can be obtained during the design process, the more likely that designers can make good informed decisions, as more options remain open to them.

For a typical design process as analysed by Tam (2004), a number of stages are present. The goal of this research is to identify the method of manufacture that will lead to minimum unit costs, whilst meeting the performance requirements, as early as possible within this process. Early identification of the minimum cost method of manufacture can also allow designers to consider different options for reaching the desired performance, which are likely to have differing costs, and effects on the method of manufacture.

A specific example of this might be load bearing aerofoil sections in a gas turbine, if passages are required in these aerofoils (for anti-icing, oil transmission, etc.) then the choice of manufacturing might be to forge, create through additive layer manufacture, cast etc. However, if the aerofoils have a twisted cross section, then forging is no longer possible. This results in a reduction in structural strength, and results in larger than necessary aerofoils or secondary load bearing structures. The end result may be that the improved aerodynamic capabilities of a twisted aerofoil result in additional mass and cost. These effects should be quantified such that an informed decision can be made on the optimal design.

The research thus aims to:

1. Demonstrate the possibility of embedding unit costing capabilities throughout the design process, whilst retaining the ability to change the manufacturing process as

necessary, and reusing detailed knowledge from previous designs where necessary.

2. Provide high levels of cost modelling automation, by linking with manufacturing process knowledge and a CAD geometry modelling package.

3. Take into account uncertainty and scrap/rework rates, whilst also including the possibility to explore alternative competing and complementary manufacturing processes.

4. Allow the inclusion of newly emerging manufacturing technologies to be considered as manufacturing processes, even where no historical information is available for them.

It is difficult to calculate costs without a detailed product definition, or the assumption of similarity with a previous known value, as one of the key factors in manufacturing cost is the choice of manufacturing process, the driver of which is the geometric and tolerance (G&T) data. It is intended that when G&T information is undefined, estimates will be used from past designs or expert estimation, to be replaced with confirmed values when known. Since the manufacturing route is to be flexible, tolerances must be defined for manufacturing process selection.

As shown in Section 2.3, it is to be noted that one of the key limitations in current research is the inability to consider a flexible one-to-many relationship, between (manufacturing) features and manufacturing processes. This is a key shortcoming in current literature, which this thesis will address.

1.2 Hypothesis

The hypothesis, therefore is as follows:

> *Conducting manufacturing process selection prior to manufactuing sequencing in CAPP can lead to suboptimisation. Dynamic process selection, conducted within an optimisation will improve this, and thus allow superior decision making by component designers and manufacturing experts.*

1.3 Objective

> *The objective of this research is, given design geometry including tolerance data, to define a methodology to calculate the minimum possible manufacturing cost of a component, to within acceptable geometry and quality. It must not be limited to specified manufacturing routes, and should also be capable of use as early as possible in the design process.*

This methodology is intended to be usable in the following scenarios:

- For a manual design decision, potentially resulting in performance and/or cost changes must be made. A designer may be aware of the performance implications but, without manufacturing expertise or a suitable cost model, not the costs. Knowledge of some of the implications of a decision does not allow a rational trade off to be made.
  - A blisk is a solid, single component comprised of compressor blades and the disc, typically manufactured from solid metal, or with the blades welded directly to the disc instead of being retained by a slot/blade root combination.
  - As an example, if a parameterised model of an intermediate compressor blisk is coupled with an aerodynamic model, for specific fuel consumption (SFC), and this methodology to predict the manufacturing cost, then it would be expected that a change in blisk specification (e.g. blade count, size, shape and tolerance) would drive an SFC change and cost change, but that also at some specification the most cost effective manufacturing route will change, from machined from stock/forging to friction welding. The question to be asked by the designer is whether the higher cost of blisk manufacture is worth the mass reduction. This cannot be evaluated without knowing the change in cost.
- Within a design optimisation loop, either as an objective function in multi-objective optimisation, or as a constraint setting the maximum cost. The design variables could then be anything the designer wishes to vary, and existing techniques could be used to evaluate mass, structural strength, aerodynamics and life.

The developed methodology will be demonstrated to deduce the minimum unit cost for a selection of components obtained from the industrial sponsor. The methodology developed here could (and should) be used in a wider sense as an input to life-cycle value engineering (Cheung et al., 2008) as it provides a superior estimation of unit cost, when outside the boundary of existing models, due to its derivation of the manufacturing route.

1.3.1 Sub-objectives

Given the objective:

> *The objective of this research is, given design geometry including tolerance data, to define a methodology to calculate the minimum possible manufacturing cost of a component to within acceptable geometry and quality, it must not be limited to specified manufacturing routes.*

An analysis tool/methodology to meet this objective has a list of tasks it must be able to perform, and this thesis must prove and justify that each of these tasks is possible and practical:

Task 1 To deduce the shape created using additive or forming processes, and the cost thereof of these processes.

Task 2 To determine what possible manufacturing processes *can* be used to make the remaining features (i.e. any not already created by the additive or forming process).

Task 3 To derive *which* of the possible manufacturing processes *should* be used to make the remaining features.

Task 4 To deduce the optimal order in which these processes should be conducted.

Task 5 To make any remaining decisions such as tooling used, access direction.

1.4 Vision

Activity Based Costing (Qian/Ben-Arieh, 2008) combined with process optimisation is an incomplete field of study, and this is the area this research aims to improve. It goes beyond the assumption that manufacturing routes are fixed (or manually constructed), to allow the manufacturing route to change in response to design geometry and tolerance data. Furthermore, by taking into account compound scrap rates, the ideal manufacturing process will depend on the cost incurred up to that point (as proved in Section 5.4).

To demonstrate the need to go beyond the assumption of fixed manufacturing routes for similar components, a typical industrial example will be illustrated. We have two components with the same purpose a and b, of a similar geometric design, but differing scales and manufactured years apart. It could be that the least costly manufacturing route for both components is identical in content, although likely to be differing in operation time. If we denote the least costly manufacturing route for component a as o, and the least costly route for b by p. Then p will be identical (in content) to o if:

- p was optimal at the time of its derivation, (i.e. before commencement of production for a).
- The two components are sufficiently similar in terms of G&T (Geometric and Tolerance) data.
- o does not contain any manufacturing processes which were unavailable when p was derived.

If b is sufficiently different from a, manufacturing advances have been made which should be used in production of this component, or if p was suboptimal, then all new manufacturing routes (or at least all those which might be cost effective) should be explored. This can be done manually or computationally, using manufacturing knowledge and rules. Thus this research aims to construct and demonstrate a systematic and automated framework, which uses formally structured manufacturing knowledge.

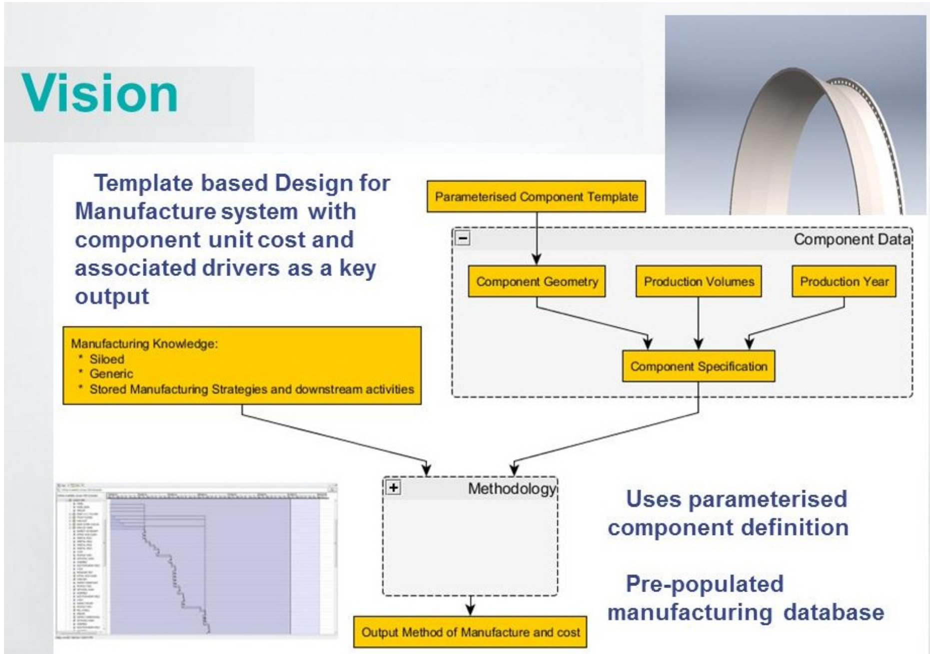


Figure 1.1: Vision of the research

The overall vision of this research (Figure 1.1) is to devise a methodology which is capable of creating, in a systematic and automated way, the optimal method of manufacture from a cost perspective. It is beneficial to separate the manufacturing knowledge from the component specific knowledge, to enable reusability and maintenance of manufacturing knowledge. Manufacturing knowledge can be pulled in as necessary only to create, and assess the feasibility and cost of, manufacturing routes (or permutations of the manufacturing strategies). This methodology can give real time feedback on the cost implications of change, whilst a designer is experimenting with possible solutions to satisfy their specification requirements. This will allow for more informed decision making.

Dynamic process selection is an area not covered well in research literature. By this it is meant that an optimiser has the task of building the set of manufacturing processes, from first principles. There are many research papers (Section 2.3, Section 2.2) which optimise a manufacturing route, once the processes are selected. A few have a one to one relationship (features with processes), only one has a one-many relationship but it is insufficiently flexible (as shown in Section 5.4) because the probabilities are pre-populated by expert opinion.

1.5 Overall Architecture

The work can be cleanly divided into sections:

- Prework which builds the data required across all component types (Figure 1.2).
- Prework which is specific to a style of component style (Figure 1.2).

- Current design work on a specific instance of that component (Figure 1.3).

As shown in Figure 1.2:

- Design features require a mapping to one or more usable elemental features.
- Elemental features require one or more associated manufacturing features to be usable.
- Manufacturing processes require machines which can create them.

In order to create a component template, comprised of an NX model and Component Integration Spreadsheet (CIS) (Section 1.6.3), it is necessary for design features to exist to cover all geometry.

It should be noted that in section 1.7.2, it is shown that Rolls-Royce is planning to introduce a design by feature system. This, and its benefits in so far as this research is concerned, will be explained in Section 1.7.2.

This component template can then be used to design a component of that type for a specific application, its applicability will depend on the quality and flexibility of the parameterisation.

The component template can then be used (as in Figure 1.3) as the input to the design process for a specific application. After changing the geometry and/or tolerance values within the bounds of the parameterisation, the methodology can be run to deduce the optimal method of manufacture, and associated cost. This method of manufacture can be stored to enable very rapid re-analysis of cost (i.e. assuming a fixed method of manufacture). Of course the penalty of this is that this method of manufacture may no longer be optimal. It cannot be known whether a point has been reached where the method of manufacture should be changed, without rerunning the optimisation. A more detailed diagram of the interactions between FoSCo and MCSOT is shown in Section 1.6.

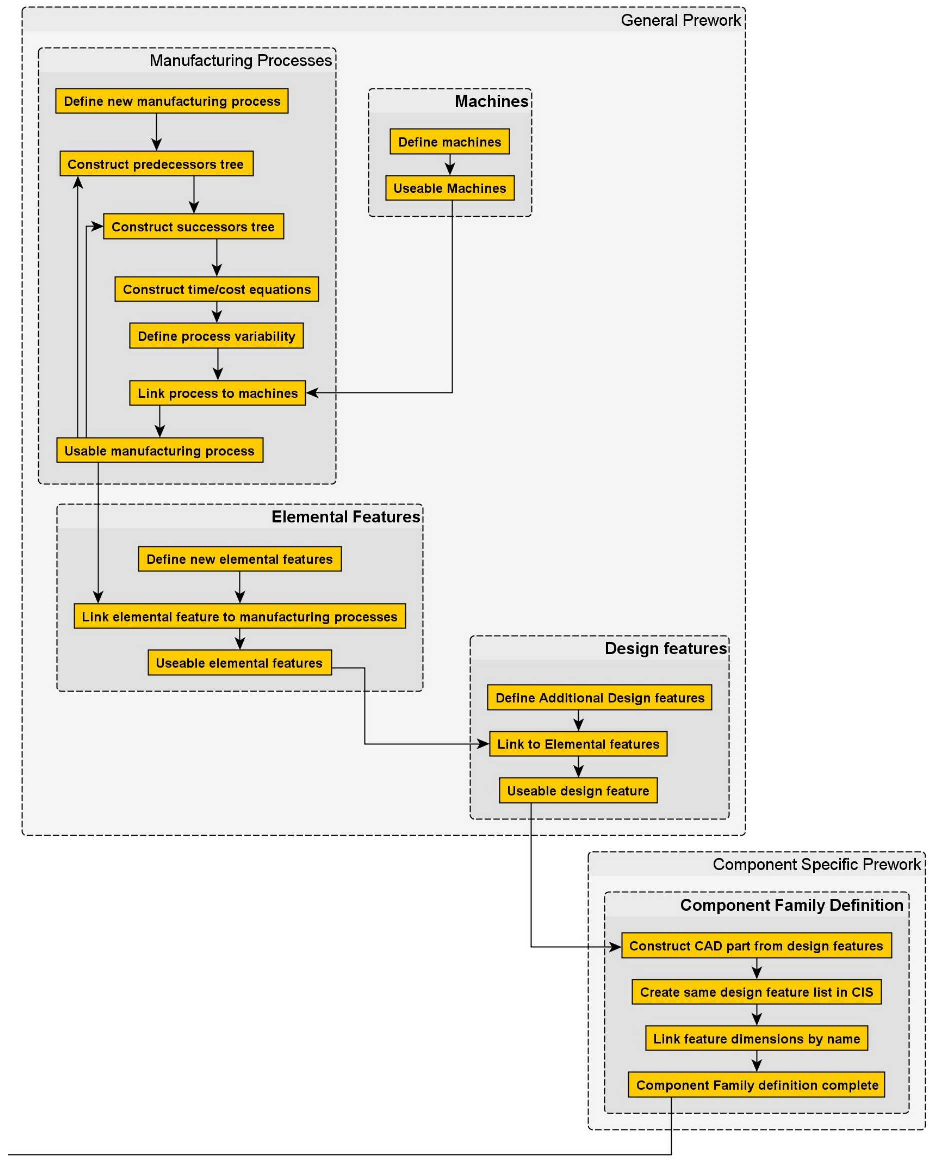


Figure 1.2: Overall architecture - Prework

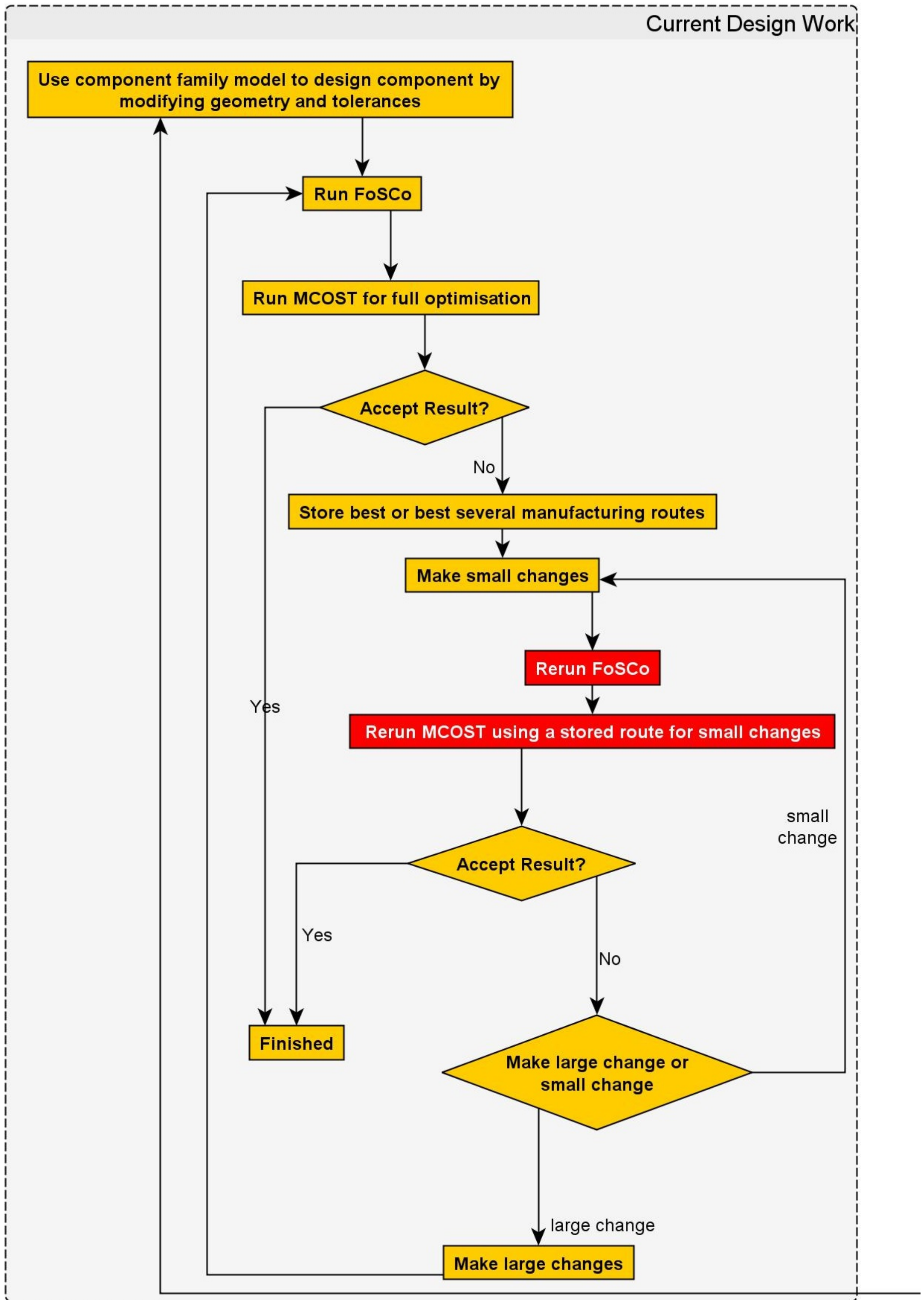


Figure 1.3: Overall architecture - Current Design work

1.6 Methodology structure and interfaces

The primary purpose of this research is to develop a methodology which can predict costs of manufacture, and allow designers to be informed about the cost implications of their decisions. Thus a convenient method of interaction for the designers is required.

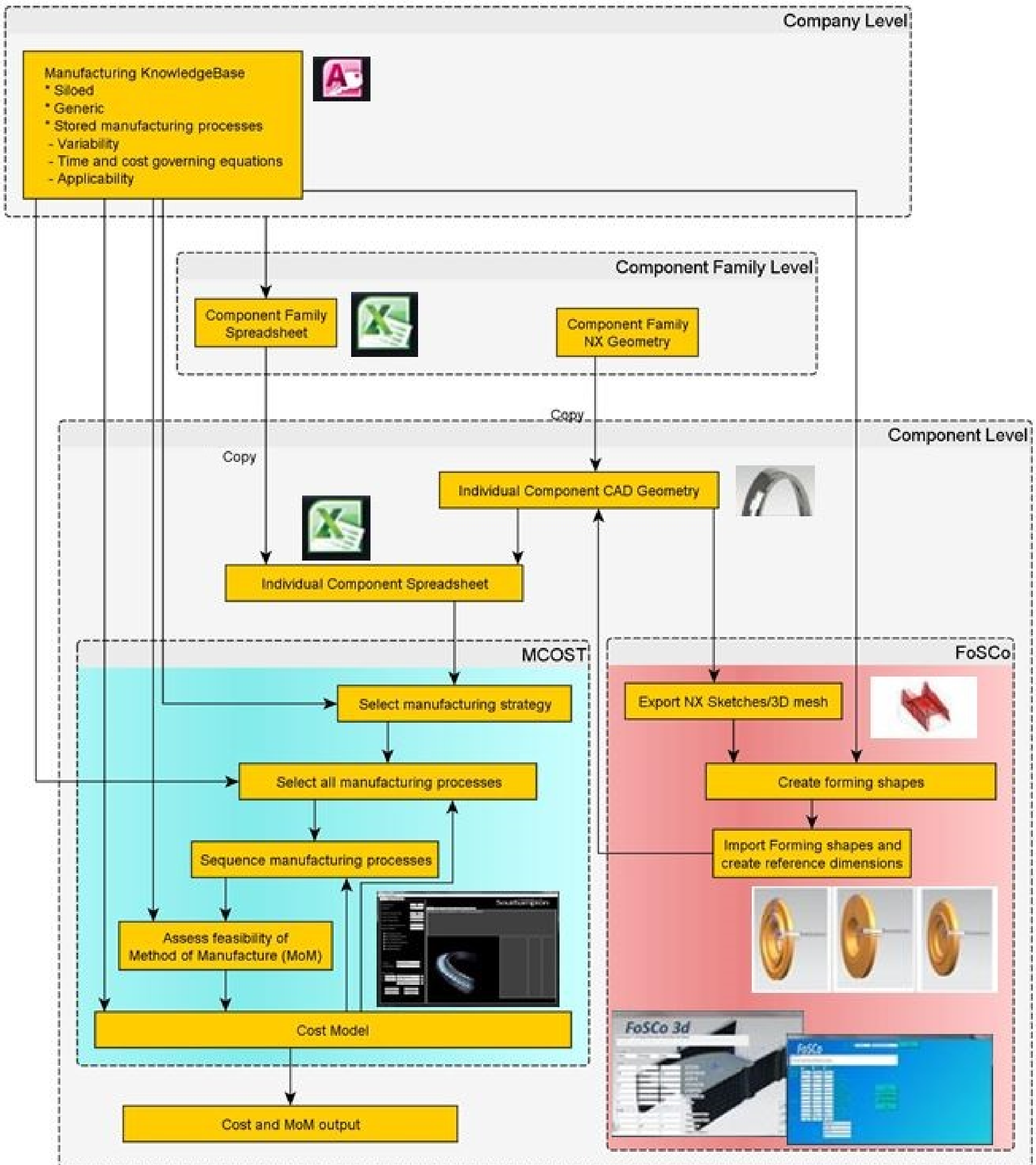


Figure 1.4: The overall architecture of MCOST, FoSCo, geometry, manufacturing and interface files

This research relies on two complementary methodologies, which will be explored over the next two sections, together these tools complete all the tasks required in Section 1.3.1:

FoSCo The **Fo**rming **S**hape **C**reator estimates the forming shape(s) which would be created, and calculates the distances between the FMP (final machined part) shape

and the forming shape for later use by MCOST.
Satisfies:

Task 1a To deduce the shape created using forming processes.

MCOST The **M**anufacturing **C**hoice **O**ptimisation and **S**equencing **T**ool is the central piece of software, choosing how to manufacture the component, but relying on component specifications, manufacturing knowledge and FoSCo.
Satisfies:

Task 1b Calculate the cost thereof of these (forming) processes.

Task 2 To determine which possible manufacturing processes *can* be used to make the remaining features (i.e. any not already created by the additive or forming process).

Task 3 To derive which of the possible manufacturing processes *should* be used to make the remaining features.

Task 4 To deduce the optimal order in which these processes should be conducted.

Task 5 To make any remaining decisions such as tooling used, access direction, etc.

It was decided that the best method of interaction was an interface to Siemens NX, as this is the industrial sponsor's CAD geometry creation tool. As shown in Figure 1.4, MCOST can be launched from within NX when a cost analysis is desired. FoSCo must also be run (prior to MCOST) whenever the geometry is changed, if the change may result in a change in precursor shape.

It would be possible to code both FoSCo and (with more difficulty) MCOST to run directly in NX Open (the NX Application Programming Interface), but from a research point of view was deemed unnecessary given that it is arbitrary exactly where the component data is stored. Hence pragmatism has been used to determine the location of storage of relevant data, and this research has elected to create a C# interface through excel and a csv file, rather than attempting to read directly from inside NX.

A fully automated method able to run multiple full optimisations, and store the results, was also coded, requiring simply a file detailing the required parameters for each optimisation. This allows multiple components to be optimised overnight (for example) or on a high performance cluster.

1.6.1 Software

The software used in this research is:

Siemens NX 8 Used for CAD modelling, this is simply because it is the industrial sponsor's corporate tool, and thus they have relevant experience and models which can potentially be reused. NX 8 was the version the University of Southampton had at the time of this research.

C# General purpose programming language used for the majority of the analysis, C# is used due to its built-in GUI construction capability, speed advantage over MATLAB, and the ease with which customised data structures can be constructed.

Microsoft Excel Used mainly for data transfer and storage of component data. Used because of the ease of interface with C#.

Microsoft Access Used as a prototyping tool to construct and edit the database which stores manufacturing knowledge.

MatLab Used as a prototyping language for FoSCo due to its inbuilt graph plotting capability. Used due to familiarity.

1.6.2 Created Tools

There have been multiple tools created during the course of this research, to prove the value of the developed methodology, these are outlined below:

FoSCo 2d The **Fo**rming **S**hape **C**reator takes a 2d sketch, estimates the forming shape(s) which would be created and calculates the distances between the FMP (final machined part) shape and the forming shape for later use by MCOST.

FoSCo 3d The **Fo**rming **S**hape **C**reator takes a 3d model, estimates the forming shape(s) which would be created and calculates the distances between the FMP (final machined part) shape and the forming shape for later use by MCOST.

2d/3d NX I/O 2d and 3d codes for NX interface to allow export to and import from FoSCo and MCOST.

MCOST The **M**anufacturing **C**hoice **O**ptimisation and **S**equencing **T**ool is the central piece of software, it creates a cost optimal manufacturing process by creating and sequencing the process, then choosing the machine and TAD for each operation.

Component Interface Spreasheet This is a spreadsheet which provides an interface between the CAD geometry and MCOST (Section 1.6.3, Section 7.3).

1.6.3 The component model and spreadsheet

In Section 7.3 sections of the Component Interface Spreadsheet (CIS) are shown. These show the method of linkage between the CAD geometry package (NX), and the representation of design features stored in the CIS. However in summary the essence is to name the dimensions in NX, export them as a .txt file, and pull them into the CIS using a VBA macro. They can then be referred to by name in the CIS.

MCOST will then read the design features in the CIS line by line and convert them into elemental features, this is explained in Section 5.7.3.

A new component template will sometimes be required, because the current template for this type of component does not have the correct design features, for example, or because there is no existing template for this type of component.

The method of creation is to concurrently construct a CAD model for the component and the CIS, the CIS will contain (design) features, as will the CAD model. Dimensions in the CAD model must be named, and the CIS must refer to the appropriate name. Each row in the design features sheet in the CIS (Section 7.3) represents a single design feature, there are various geometrical and tolerance parameters which must be defined for the design feature, dependant on the design feature definition in the knowledgebase (Section 1.6).

1.7 Industrial sponsor context

1.7.1 The Front Bearing Housing (FBH)

The author's research was sponsored by Rolls-Royce and therefore the primary test cases used (Section 1.8) and the manufacturing processes incorporated in the knowledgebase reflect the preferences of them.

Test case one and two were both drawn from the front bearing housing assembly. This assembly includes both aerodynamic and structural components, and as such renders it a suitable assembly from which to draw test cases. The FBH is located directly behind the low pressure compressor (Figure 1.5), and removes most of the swirl from core air in addition to locating the front bearings, for the low pressure and intermediate pressure system shafts. CAD images of the assembly are shown in Figure 1.6, all of these four diagrams were obtained from the industrial sponsor. In addition as a validation exercise a blisk is used to prove the applicability of the forming shape creator (Section 4).

Figure 1.5: The location of the FBH in a typical engine (Rolls-Royce, 2013)

1.7.2 Feature Classification and standard features

The following figures and description are intended to show the possibility of automatically building cost models, during the definition of components within the CAD geometry. This could wholly or partially remove the need to use feature recognition prior to process planning, and would remove some of the overhead shown in Section 1.5, specifically the CIS part of the component specific prework.

Rolls-Royce intend to design using standard features, by which they mean parameterised CAD definitions of commonly used features, which can be superimposed on appropriate surfaces/edges etc. of a CAD model. The hierarchy of these features is shown in Figure 1.7. However, as can be seen some of the features are both design and manufacturing, some are simple, and some are compound features. This definition is thus incomplete and

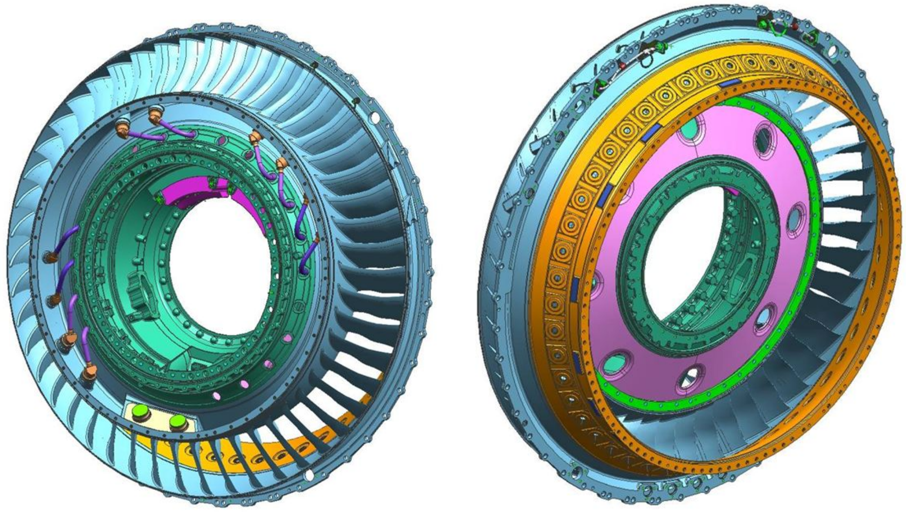


Figure 1.6: A front (Trent 1000) and rear (Trent 800) view of the Trent FBH (C/O Johnson, 2012)

insufficiently formal, however, it is a starting point for a feature based design method such as outlined in Section 2.1.2.

If completed and exclusively used, it could allow the automated building of cost models, including as an input to MCOST (Section 5). That is to say that the incorporation of a design feature into a CAD model, could also update the CIS to include a new line for the design feature, and automatically link up the feature dimensions to named NX dimensions.

Rolls-Royce have been working on identifying a hierarchical model of feature classification for the various components of the FBH assembly as shown in Figure 1.8.

Following this feature classification Rolls-Royce has begun to identify the design and manufacturing constraints in documents. As shown in Figures 1.9; 1.10.

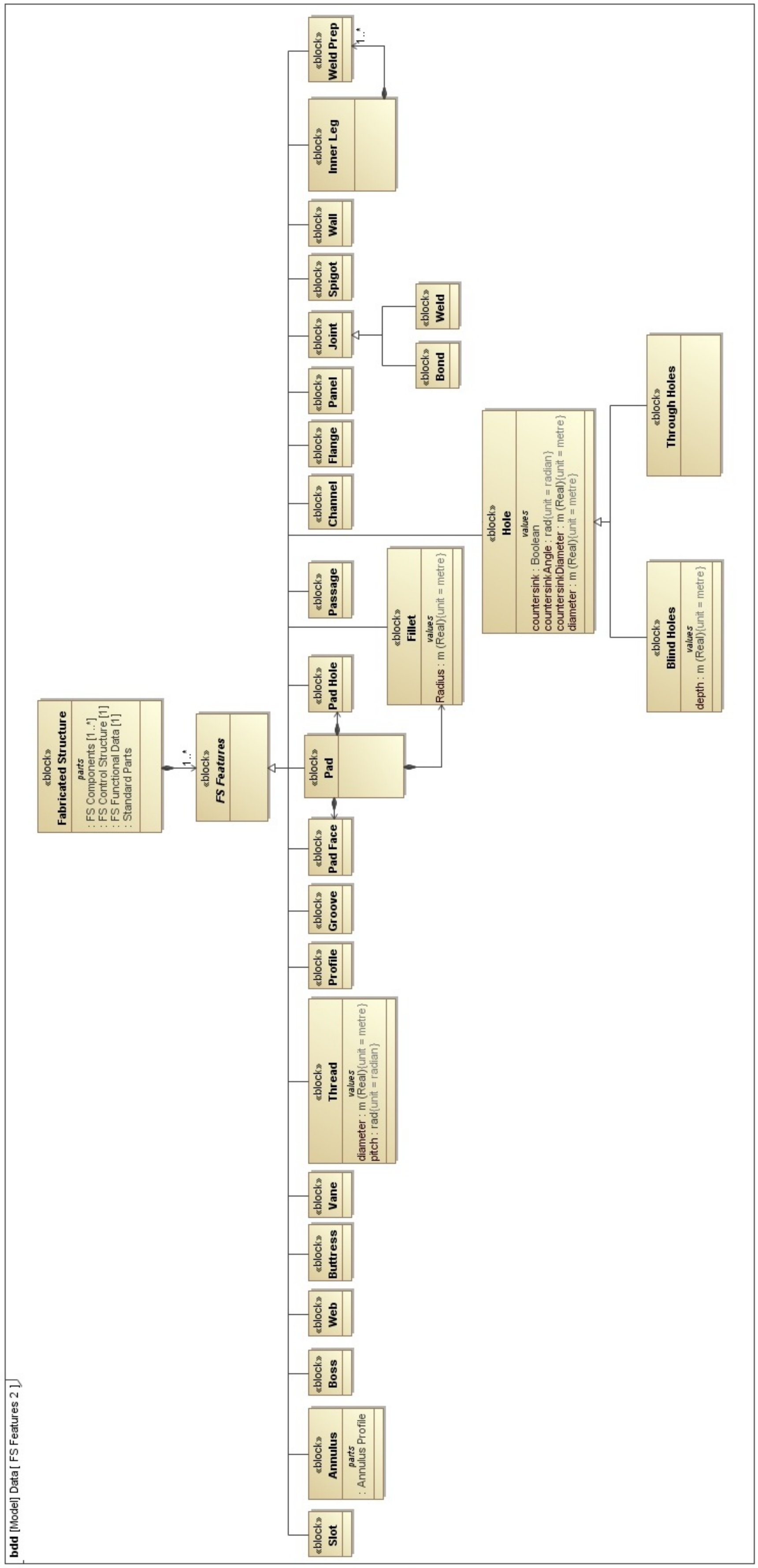


Figure 1.7: Standard Features Hierarchy (C/O BOYCE, 2011)

Entity	Function	Ref. N	Sub-Entity	Sub-Sub-Entity	Function
Rear Outer Casing	VIGV Outer location Interfaces with IPC front case	35	ROC Flange		Assembly to IPC front casing
		36	Bolt Holes		
		37	VIGV Bosses - Counter Bored Hole		Mount VIGVs
		38	VIGV Bushes		Mount VIGVs
		39	Rubbing Pads		
		40	Weight Relief Grooves		Attachment to compressor stages
		41	Rigging Slots		
		42	Dowel Holes		Locating Feature
		43	Non-constant thickness annulus profile		Strength/Weight Considerations
VIGV Flange (Rear Inner Flange)	Provide welding surface for ROV, ROC attachment Interfaces with VIGV Inner Mount	44	Interface Flange		Interface to rear outer casing?
		45	Bolt Holes (Clamping Feature)		Attachment Feature
		46	Dowel Holes (Locating Feature)		Locating Feature
Front Inner Flange			Interface Flange		Interface to fan rear seal
			Bolt Holes (Clamping Feature)		Attachment Feature
			Dowel Holes (Locating Feature)		Locating Feature
Front Panel	Transfer loads from hub to outboard Allow Transfer of services to hub	47	Swaged Service Holes		Pass through service pipes
		48	Panel thickness		Strength/Weight Considerations
			Panel Angle		Strength/Weight Considerations
			Hub Weld Prep		Connect to hub
			ROV Weld Prep		Connect to ROV
Rear Panel	Transfer loads from hub to outboard Allow Transfer of services to hub	49	Swaged Service Holes		Pass through service pipes, inspection access (welding)
		50	Panel thickness		Strength/Weight Considerations
			Panel Angle		Strength/Weight Considerations
			Hub Weld Prep		Connect to hub
			ROV Weld Prep		Connect to ROV

Figure 1.8: FBH Feature Classification (C/O Johnson, 2011b)

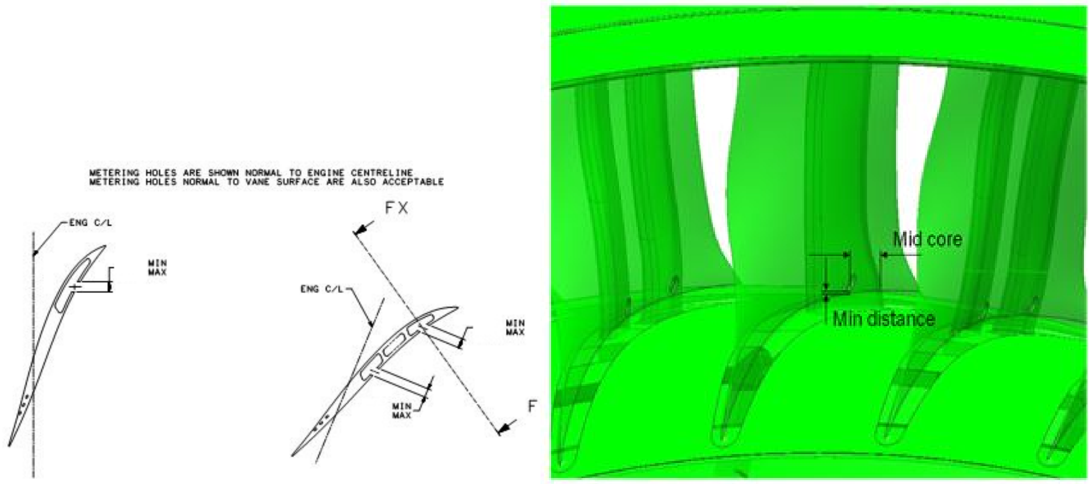


Figure 1.9: Airfoil De-icing Holes Model and Drawings (C/O Johnson, 2011a)

Relationship Description:	
Parent Design Feature ID:	ESS_Aerofoil_Anti-IcingPassage
Child Design Feature ID	*Not Applicable*
Analysis Feature ID:	*ESS_Aerofoil_AirMeteringHole*
Manufacturing Feature ID	

Design Rule:			
ID	*Design Rule:*	*Description (Activity):*	*Reference*
DR1	Does not encroach on fillet at min/max tolerance stack		
DR2:	Minimise distance from fillet		
DR3			

Design Constraint:			
ID	*Constraint:*	*Description(Parameter):*	*Reference*
DC1	Withdraw pin from wax casting		

Figure 1.10: Airfoil De-icing Holes Constraints (C/O Johnson, 2011a)

1.7.3 Model Construction Data

It is assumed that all industrial companies attempting to predict and control costs will have, at the very least, estimates of their future production volumes. In addition, it is assumed that they will have, or be able to construct, 3d geometry models using parameterised design features.

For this research, the CAD package in which the components will be constructed will be NX8[1], simply because the expertise to extract data exists within the industrial sponsor.

The cost of production of a component can be divided into two categories:

Bought in cost This is the cost that a company pays to obtain raw materials or part finished components.

Part finished components are, in extremis, only composed of raw material and manufacturing costs.

Manufacturing cost This is the cost of the manufacturing operations necessary to transform the inputs of raw materials, or partly machined components into the FPG (Finished Part Geometry). These costs contain amortisation of indirect costs, as part of the accounting process.

This research will thus consider only raw material and a detailed breakdown of manufacturing costs, although clearly for bought in parts a profit margin will be implicit within all cost rates. Material cost estimates will be obtained directly from the industrial sponsor's cost modelling department. In other words, this research will ignore the distinction between the company's internal and external supply chain, thus concentrating only on the production cost. Commercial and political considerations will be ignored.

1.7.4 Manufacturing knowledge

Standard material removal processes have standard speeds predefined according to the type of operation and the material number (a machinability index). An example of these tables for drilling is shown in Table 1.1. In addition, there exists similar data for other machining operations such as turning and milling, which can be inverted to give MRRs (Material Removal Rates) for various chip forming manufacturing processes. This data is used in Rolls-Royce cost models, and an extract is shown in Figure 1.11. These data will be used on the cost model integrated within the methodology developed by this research.

[1] NX8 was the latest version the University of Southampton had available at the time.

DRILLING SPEED (m/min)				REAMING SPEED (m/min)				TAPPING SPEED (m/min)	
Table	High Speed Steel	Table	Carbide	Table	High Speed Steel	Table	Carbide	Table	Speed
1	60.85	1	121.70	1	22.85	1	45.70	1	17.10
2	57.85	2	115.78	2	22.85	2	45.70	2	12.80
3	53.45	3	106.90	3	22.85	3	45.70	3	12.80
4	25.95	4	51.90	4	10.65	4	21.30	4	8.55
5	22.95	5	45.90	5	10.65	5	21.30	5	6.40
6	18.35	6	36.70	6	6.85	6	13.70	6	5.35
7	18.35	7	36.70	7	6.85	7	13.70	7	5.35
8	15.15	8	30.30	8	6.85	8	13.70	8	5.35
9	13.75	9	27.50	9	6.10	9	12.20	9	4.15
10	13.75	10	27.50	10	6.10	10	12.20	10	3.85
11	12.15	11	24.30	11	6.10	11	12.20	11	3.85
12	12.15	12	24.30	12	6.10	12	12.20	12	3.65
13	10.55	13	21.10	13	5.15	13	10.30	13	3.45
14	9.15	14	18.30	14	5.15	14	10.30	14	3.45
15	9.15	15	18.30	15	5.00	15	10.00	15	3.00
16	6.20	16	12.40	16	5.00	16	10.00	16	2.80
17	6.20	17	12.40	17	5.00	17	10.00	17	2.55
18	7.60	18	15.20	18	3.65	18	7.30	18	2.50

Table 1.1: Drilling speed against manufacturability index for several types of drilling operation (C/O WISEALL, 2010)

	Machinability number	1	2	3	4	5	6	7	8	9	10	25
Header row	Process			0	0		0	0	0	0	0	Units
Process default	Simple load/unload	3.65	3.65	3.65	3.65	3.65	3.65	3.65	3.65	3.65	3.65	60 sec
Process 1	Complex load/unload	12.28	12.28	12.28	12.28	12.28	12.28	12.28	12.28	12.28	12.28	60 sec
Process 2	Turning, Rough	531915	426257	263158	113366	86858	74388	74388	61847	54654	45631	1.6667e-11 m^3/sec
Process 3	Turning, Finish	46760	37027	22695	19270	14925	12635	12635	10572	8966	8966	1.6667e-11 m^3/sec
Process 9	Broach per slot	3.11	3.11	3.11	3.11	3.11	3.11	3.11	3.11	3.11	3.11	60 sec
Process 10	Grind, External Plunge form, Rough	0	0	0	2421	2421	2421	2421	2151	2150	2151	1.6667e-11 m^3/sec
Process 11	Grind, External Plunge form, Finish	0	0	0	1159	1159	1159	1159	1073	1073	1073	1.6667e-11 m^3/sec
Process 12	Grind, Internal Plunge form, Rough	0	0	0	1183	1183	1183	1183	1183	1183	1183	1.6667e-11 m^3/sec
Process 13	Grind, Internal Plunge form, Finish	0	0	0	591	591	591	591	591	591	591	1.6667e-11 m^3/sec
Process 14	Grind, External Face, Rough	0	0	0	501	501	501	501	471	471	471	1.6667e-11 m^3/sec
Process 15	Grind, External Face, Finish	0	0	0	501	501	501	501	471	471	471	1.6667e-11 m^3/sec
Process 16	Grind, Internal Face, Rough	0	0	0	612	613	613	613	613	613	612	1.6667e-11 m^3/sec
Process 17	Grind, Internal Face, Finish	0	0	0	377	377	377	377	377	377	377	1.6667e-11 m^3/sec
Process 18	Auto argon arc weld 2,00mm	50	50	50	40	40	40	40	38.46	38.46	38.46	0.00002 m/sec
Process 19	Resistance seam weld 0,90mm	66.67	66.67	66.67	100	100	100	100	100	100	100	0.00002 m/sec
Process 20	Resistance seam weld 2,00mm	55.56	55.56	55.56	76.92	76.92	76.92	83.33	83.33	83.33	83.33	0.00002 m/sec
Process 21	Electron beam weld 6,00mm	22.22	22.22	22.22	20	20	20	20	20	20	20	0.00002 m/sec
Process 22	Ring rolled forging	1	1	1	1	1	1	1	1	1	1	1
Process 23	Band Saw	237	188	115	89	69	58	58	54	47	47	1e-6 m^2/sec

Figure 1.11: An extract of lookup table containing process rates for some of Rolls-Royce's commonly used manufacturing technologies (C/O Reuss, 2010)

1.7.5 Verification/Validation Data

To verify convergence of the algorithm test case zero (Section 1.8) will be used, this is a simplified version of test case one, contrived to test the optimisation algorithm. To show convergence toward the global optimum for other test cases, convergence graphs will be shown.

To validate the methods of manufacture output by MCOST, SAP ERP data will be used. This gives manufacturing routes, times and the cost, for each manufacturing operation in the industrial sponsor's internal supply chain.

The standard location for storage of Method(s) of Manufacture (MoM) in Rolls-Royce, once production has begun, is SAP ERP software (usually referred to as SAP). An example of an exported MoM from SAP (in CK-86 format) is shown in Figure 1.12. These data were automatically extracted to produce manufacturing process plans of current components, and can be used for the validation of test case one. Cost data for test case two was also used.

16.08.2010 Dynamic List Display 1

Material TRN101737 DUCT,INTER COMP SUPP ASSY
Plant SCAM STATIC COMPRESSORS B'WICK
Costing Variant ZCUR Current Standard
Costing Version 1
Costing Date from 15.08.2010
Costing Date to 01.01.2011
Lot Size 1,000 NPTItem
Cost Base 1 NPTItem

Cost of goods manufactured
Explosion Level *

Lev	Material	Plnt	Item Category	Resource	Resource (Text)	Cost Eleme	Total Value	Fixed Value	Curr	Quantity	Un	POrg	Info rec.	Item Category (Text)
1	ESC30-P20BC	SCAM	M	SCAM ESC30-P20BC	CONTACT IS ESC30P20BC	580326								Material
1	NQF005214	SCAM	M	SCAM NQF005214	BLOCK	580327								Material
2		SCAM	E	3005 3005FITD LABOUR	INSPECT ON RECEIPT	800005								Internal Activity
2	FW15897	SCAM	M	SCAM FW15897	BOX,ASSY, SPEED PROBE	580326								Material
2	FK90710	SCAM	M	SCAM FK90710	CABLE, SUB-ASSY	580327								Material
3		SCAM	E	3005 3005FM07 LABOUR	INSPECT ON RECEIPT	800005								Internal Activity
3	ESW1200-010-006	SCAM	M	SCAM ESW1200-010-006	CABLE	580326								Material
3		SCAM	E	3005 3005FITD LABOUR	CUT TO LENGTH	800005								Internal Activity
3		SCAM	E	3005 3005FITD LABOUR	ASSEMBLE	800005								Internal Activity
3		SCAM	E	3005 3005FITD LABOUR	ASSEMBLE	800005								Internal Activity
3		SCAM	E	3005 3005FITD LABOUR	PRESSURE TEST	800005								Internal Activity
3		SCAM	E	3005 3005FM07 LABOUR	FINAL INSPECTION	800005								Internal Activity
3		SCAM	E	3005 3005FM07 LABOUR	FINAL CLEARANCE	800005								Internal Activity
3		SCAM	V		Bulk Material Consumption (2 PtNos)	800040								Variable Item
3		SCAM	G	ZB06 802002	All Parts Burden	802002								Overhead
2		SCAM	E	3005 3005FITD LABOUR	ASSEMBLE	800005								Internal Activity
2		SCAM	E	3005 3005FITD LABOUR	ASSEMBLE	800005								Internal Activity
2		SCAM	E	3005 3005FM07 LABOUR	FINAL INSPECTION	800005								Internal Activity
2		SCAM	E	3005 3005FM07 LABOUR	FINAL CLEARANCE	800005								Internal Activity
2		SCAM	V		Bulk Material Consumption (5 PtNos)	800040								Variable Item
2		SCAM	G	ZB06 802002	All Parts Burden	802002								Overhead
1	SPS0700V	SCAM	M	SCAM SPS0700V	CONNECTOR,MALE	580326								Material
1	TRN12925	SCAM	M	SCAM TRN12925	PIPE,SERVICE, VENT 1	580326								Material
1	TRN12926	SCAM	M	SCAM TRN12926	PIPE,SERVICE, VENT 2	580326								Material
1	TRN12927	SCAM	M	SCAM TRN12927	PIPE,SERVICE, ELECTRICS 1	580326								Material
1	TRN12928	SCAM	M	SCAM TRN12928	PIPE,SERVICE, ELECTRICS 2	580326								Material
1	TRN12929	SCAM	M	SCAM TRN12929	PIPE,SERVICE, OIL FEED	580326								Material
1	TRN12930	SCAM	M	SCAM TRN12930	PIPE,SERVICE, OIL SCAVENGE	580326								Material
1	TRN101738	SCAM	M	SCAM TRN101738	DUCT,FABRICATED STRUCTURE	580327								Material
2		SCAM	E	3005 3005FM07 LABOUR	INSPECT ON RECEIPT	800005								Internal Activity
2	NQF003593	SCAM	M	SCAM NQF003593	E.S.STATOR,RING OF VANES	580327								Material
3	NQF004161	SCAM	M	SCAM NQF004161	SHIM,ANNULUS AND PANEL LEG	580326								Material
3		SCAM	E	3005 3005FM07 LABOUR	INSPECT ON RECEIPT	800005								Internal Activity
3	FK91506	SCAM	M	SCAM FK91506	SHIM-OUTER,	580326								Material
3	NQF006349	SCAM	M	SCAM NQF006349	DUCT,I.C.D. ASSY	580326								Material
3		SCAM	E	3005 3005FITD LABOUR	WITHDRAW FROM STORES 3 OFF AGS 3732	800005								Internal Activity
3		SCAM	E	3005 3005FM11 LABOUR	PRESSURE TEST	800005								Internal Activity

Figure 1.12: An example of a CK-86 SAP data export with numerical levels C/O Johnson (2010)

1.8 Test Cases

In order to demonstrate the applicability of the methodology whilst satisfying the customer demands, multiple test cases will be used.

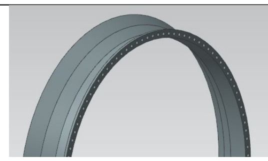
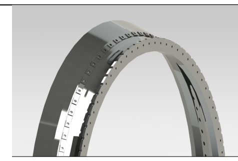
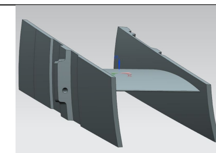
			
Test Case	Zero	One	Two
Component	Simplified rear outer casing	Rear outer casing	Engine Section Stator (ESS)
Forming Method 1	Roll Forging	Roll Forging	Forging
Forming Method 2	Centrifugal Casting	Centrifugal Casting	Casting
Forming Method 4	Machined from tube	Machined from tube	Machined from billet

Table 1.2: Test cases to be analysed

Three test cases are to be used, with multiple manufacturing strategies for each (Table 1.2). Test cases zero, one and two are shown in Table 1.3. Test case zero is intended to show that the methodology can find the global optimum, since it has been contrived to give a known optimum (for the created manufacturing knowledge database). Test cases one and two are intended to show accurate costs on different types of component, and are actual Rolls-Royce components (though with non cost effecting features such as minor fillets removed). Test cases zero and one are mainly axisymmetric, and the manufacturing will thus tend towards technologies such as ring roll forging, centrifugal casting, and turning type operations. Test case two has no axisymmetry, and will thus rely more on die forging, milling, and other similar technologies not dependant on continuous rotation of the component.

In addition to these test cases which were used for both MCOST and FoSCo, an additional test case (Figure 1.13) was incorporated purely for the testing of FoSCo. As it is an IP sensitive component Rolls-Royce did not want its manufacturing process included within this thesis, but an approximation of the outline geometry was acceptable. Figure 1.13 shows the finished part geometry and forging shapes, as estimated by the Industrial Sponsor.

Because this research does not include feature recognition, but instead uses a mapping from design to elemental features, a design feature list is also needed for each test case. This is shown in Table 1.3. Note that because the methodology will deduce the optimal machining order, the order in which the features are defined has no effect.

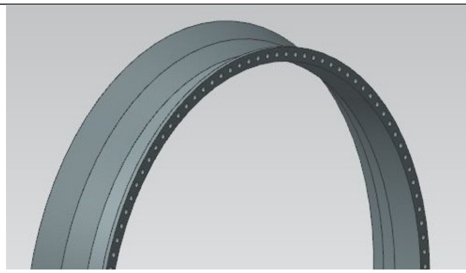
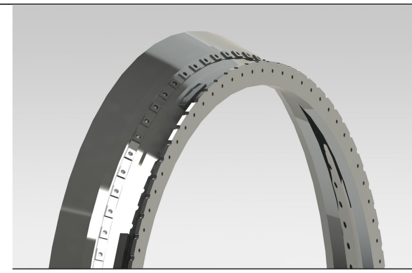
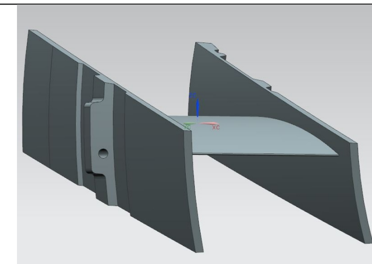
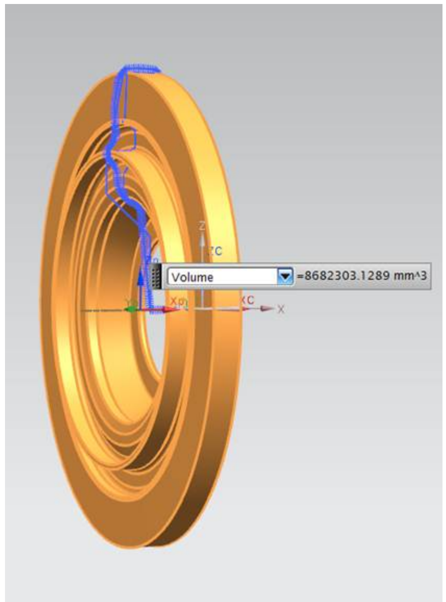
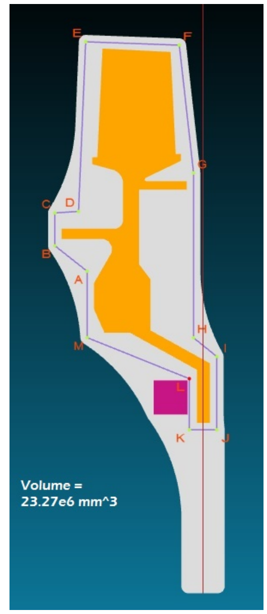
					
Test case zero		Test case one		**Test case two**	
Design feature name	**Design feature type**	**Design feature name**	**Design feature type**	**Design feature name**	**Design feature type**
Rear Outer Casing	Condition of Supply - Cylindrical	Rear Outer Casing	Condition of Supply - Cylindrical	Engine Section Stator	Condition of Supply - Billet
Outer Slots	Slot Group Set Rotated	Pads	Pad	Lower Vane Surface	Concave Slot
Inner Slots	Slot Group Set Rotated	Weight Relief Slots	Slot Group Set	Upper Vane Surface	Convex Slot
Flange	Flange	Bolt and Dowel Holes	Through Holes Counterbored	Slots Outer	Slot Group
		Variable Stator Vane Actuator (VSVA) Holes	Through Holes Double Counterbored	Slots Inner	Slot Group
		VSVA Bosses	Bushes	Slots Top	Slot Group
		Mounting Flange	Flange	Slots Bottom	Slot Group
		Slots Inner	Multi-stage Groove	Services Access	Slot
		Slots Outer	Multi-stage Groove	Mounting Bracket	Bracket
				Faces Inner	Faces
				Faces Outer	Faces

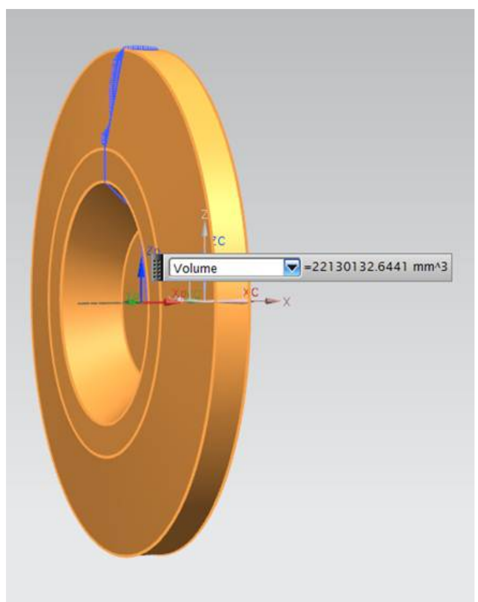
Table 1.3: Test cases: Design feature list



(a) Axis-symmetric finished part geometry



(b) Forging Shape - estimated by industrial sponsor using ACE (rules based tool)



(c) Forging Shape - estimated by industrial sponsor (ignoring fillets)

Figure 1.13: Test case three - Blisk

1.8.1 Minor Test Cases

In addition to the three primary test cases, 20 other cases were tested, to ensure that the methodology was able to cope with them without modification. These were provided by the industrial sponsor, and were the same as used by the sponsor to investigate the quality of the commercial tool aPriori, evaluated in Section 3.2.

The only modification required for accurate results, was enabling the methodology to take account of batch production[2]. Enabling the machine setup time to be amortised over all components in the batch is necessary to ensure accurate costing estimation.

The minor test cases have been split according to whether they contain mainly axis-symmetric features or not. This will determine how the formed shape is modelled by FoSCo (**Fo**rming **S**hape **C**reat**o**r), as there are two different versions. There is no distinction in MCOST. In addition the user must choose a shape to create as the initial shape, as different manufacturing processes and cost rules apply to rectangular billets, bars and tubes, be they forged, cast or supplied from supplier stock (the last choice can be made by MCOST).

The test cases are shown in Table 1.4 Table 1.5:

[2] This is part of future work.

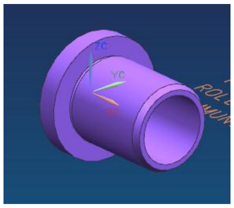
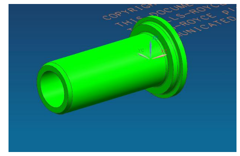
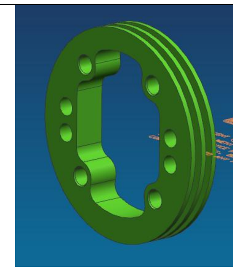
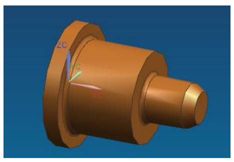
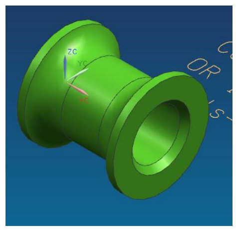
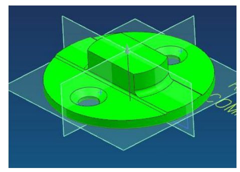
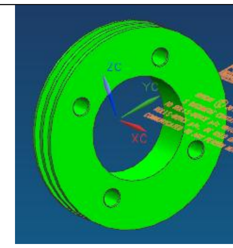
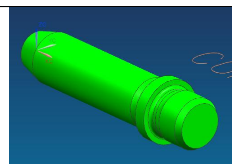
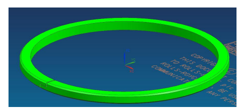
			
			
			

Table 1.4: The 10 minor mainly axisymmetric test cases

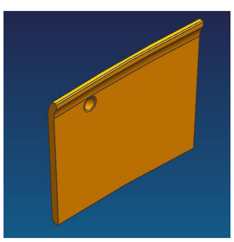
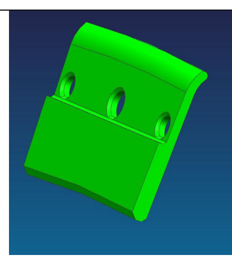
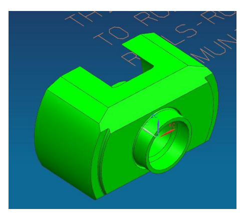
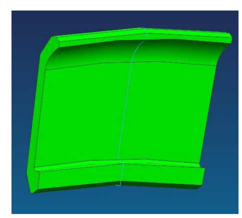
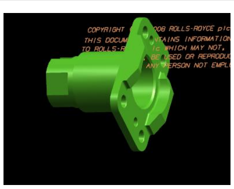
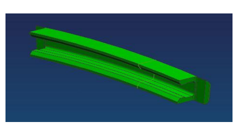
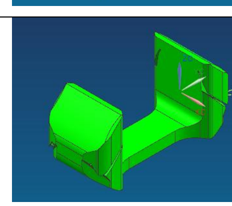
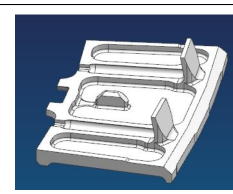
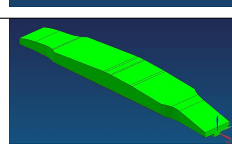
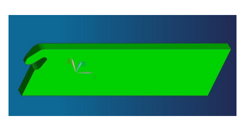
			
			
			

Table 1.5: The 10 minor non-axisymmetric test cases

1.9 Papers and patents

A patent application related to FoSCo (see chapter 4) has been filed, in 2015 (in the UK), on the industrial sponsor's instructions.

This was withdrawn and an application was made for a US and EU patent by the industrial sponsor. The intention is to use this methodology in preliminary design.

EU patent number 16180366.3 - 1954

US patent submission reference 5895 LMT

1.10 Thesis synopsis

Section 2: Literature Survey Section 2 details the academic research currently available, mainly exploring process planning, cost modelling and precursor shape prediction. It also explores the complementary technology of feature recognition, the design process, and manufacturing processes.

Section 3: Commercial Costing Systems Section 3 details and critiques commercial costing systems currently available. Commercial systems invariably have some form of feature recognition, presumably because it is much easier to sell to customers a product which does not require any significant change in the design method. However at present they are limited in not considering the manufacturing sequence, thus reducing their effectiveness when the manufacturing process has a non trivial chance of causing rework, scrap or concessions due to the tolerances being close to or tighter than the variability of the manufacturing process.

Section 4: FoSCo (Forming Shape Creator) Section 4 details the methodology used to predict the forming shape, i.e. the shape generated by the forming or additive process. It is subdivided into the axis-symmetric version and the 3d version, although as will be seen the axis-symmetric operates in the same way where z is always zero. This methodology is necessary for the optimisation below to operate, as without an accurate estimate of material removal volumes, it is impossible to accurately estimate operation times. Having explained the methodology it continues to present results, and opportunities for further work.

Section 5: MCOST (Manufacturing Choice Optimisation and Sequencing Tool) Section 5 details the methodology used to construct and optimise the manufacturing route, including the operation of the objective function (cost model). It includes a description of the data storage necessary for the methodology to operate. Optimisation of the process route takes account of the choice of manufacturing processes, sequencing, tool access direction and the optimiser attempts to minimise unit cost. As above it presents results and further research and improvement opportunities.

Section 6: Conclusions Section 6 details the conclusions arrived at during this research, and some suggestions for the future direction (of FoSCo, MCOST, and methods to reduce the overhead of using these methods) which could be taken.

Section 7: Appendix MCOST and FoSCo additional information Section 7 has additional digrams and description related to the operation of MCOST and FoSCo, it is referred to from Sections 4 and 5.

Section 8: Appendix FoSCo patent submission Section 8 contains the full patent application for FoSCo.

2 Literature Survey

A requirement has been established to embed cost knowledge into the design engineering and process planning team. It can be seen, that cost modelling once the production process is decided, is a relatively well developed area. The main area of improvement, is therefore thought to be analysis and optimisation of possible manufacturing routes, from design geometry, whilst cost modelling each iteration. This section of the thesis details the academic literature related to this subject. The logical extension of this is then to combine with life cycle modelling and conduct value analysis (discussed in Section 2.1.6), combine with factory modelling to see the dynamic effects on the supply chain, or both. However both of these will be defined as out of scope, with the research limited to attempting to automatically deduce the minimal cost method of manufacture, assuming a supply chain with unlimited capacity and load independent costs.

To begin with, the process by which components are designed and prepared for manufacture, from a top-level perspective, will be reviewed. This section will also review alternative methods of design, and geometry creation. A review of the methods used for process planning, from the point of view of supply chain management, will be conducted. The section will continue on to high fidelity process planning methods, thus exploring the computer aided process planning (CAPP) area of the literature. This continues into methods of representing and exploring the process planning search space, and analysis will be conducted on the methods of linking CAPP to the geometry creation stage of design.

Following on from this, the author will explore the Computer Aided Manufacture (CAM) literature and the manufacturing methods which can be used, each manufacturing method has limitations on its use, and properties associated with it, which any process planner must know, if they wish to make use of that particular manufacturing process.

It is then necessary to explore methods of cost modelling which can be used, where they are useful and what information they require to produce accurate data. Particular attention is focussed on generative cost modelling, because of its ability to link directly into the outputs of CAPP.

Finally, although out of scope, it is noted that there could be linkages between simulation, cost modelling and process planning. This would give the potential for a more comprehensive tool, to further improve the design of components. This is because the cost of machinery is in part dependant on the machine loading.

2.1 The Design Process

It is widely held to be the case, that design changes made later in the design process, are more costly in both time and money, as stated by Xu et al. (2006). Later in design more

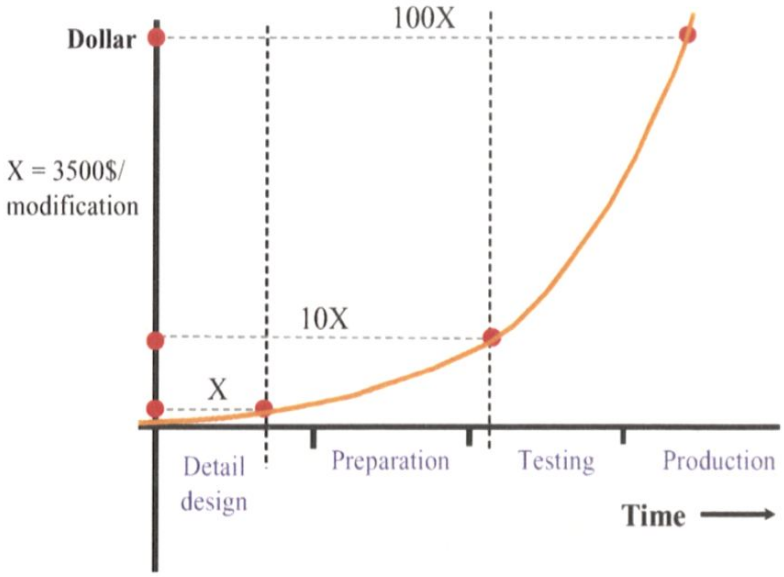


Figure 2.1: An illustration of the magnitude of the cost of design changes (for an unidentified component) through the early product life-cycle Xu et al. (2006)

time has been spent on the previously chosen option. In addition, other design decisions are likely to require re-evaluation, due to the changed design choice. An approximation of the expected costs due to a design change is shown in Figure 2.1.

The reasons for this increase in costs are due to more extensive recalculations, and in addition a need to increase the rapidity of the design process, as delivery dates are normally fixed.

The proportion of the cost split between design and lifetime manufacture will depend on, among other factors, the complexity of the product and the size of the production run. Therefore, if it is a small batch of production so design costs cannot be amortised over a large batch, it may be beneficial to under-engineer the product and accept higher materials/manufacturing costs in the interests of saving design time. This trade-off is studied in Thompson/Paredis (2010) where the cost of a series of design decisions and their outcomes are known.

It is also interesting to note, that Wasim et al. (2013) assert that whilst there are many design systems to assist with choices, they are often limited to a single objective function, such as production cost, structural strength or performance. This is the area that Value Driven Design (VDD) is intended to address given its ability to reduce multiple objective functions to a single value (usually currency to reflect profit). Profit is often used, as it is in principle attributable back to the profits of the manufacturer, and is thus directly useful from a financial point of view. Other forms of multi-objective optimisation such as pareto fronts can be used, but are not designed to give a single value for design quality.

2.1.1 Concurrent vs Iterative Loop Engineering

Concurrent Engineering (CE) is a system of engineering, where engineering activities which were traditionally carried out in a sequential way are carried out simultaneously. As explained by Gao et al. (2008):

> "CE is a philosophy that suggests the need to consider design issues simultaneously where they were considered sequentially in the past".

The main purpose of CE is the shortening of lead times, thus enabling the adoption of technology earlier than otherwise to gain competitive advantage. There are several methods to analyse the benefits (positive or negative) of concurrent engineering, some of which are detailed by Gao et al. (2008), in addition to their suggestions on improving the design process. This analysis is beyond the scope of this work, but this section is simply an illustration of alternative design methods.

In contrast to concurrent engineering iterative loop engineering requires the conducting of a series of tasks iteratively, and using the output of one task to guide another, for example repeating design and CAPP iteratively until a design which fulfils all design requirements is created, then progressing to manufacturing.

A paper by Khoshnevis et al. (1994) presents a Real Time Computer Aided Process Planning (RTCAPP) system. The system indicates to the designer the cost implications of their decision in real time, as they are designing a part, by producing a process plan using dynamic programming and a knowledge based process planner. This paper is significant, in that it is the only paper found, which states the intention to be used by designers to consider the implications of their design decisions, but also uses generative process modelling. It reuses as much of the existing process plan as possible, thus reducing optimisation time, although the benefit of this will vary according to the magnitude and type of the changes. The system designed by Khoshnevis et al. (1994) can be used for prismatic parts, and it is assumed that the cost rates for machinery are pre-defined since no mention is made in the paper to the contrary.

Tammineni et al. (2009) use a form of concurrent engineering in their knowledge based system (Figure 2.2). Input to the distributed knowledge base, is performed by experts in the relevant area: designers, cost modellers and manufacturing engineers, who interact through the knowledge base library and manufacturing databases. This type of knowledge storage is particularly of interest to large organisations, as they have experts in the various fields. The relevant expert can therefore be given permission to alter only the section of the knowledgebase relevant to their expertise.

Set-Based Concurrent Engineering is a concept developed by Toyota and explained by Morgan/Liker (c2006.). One of the concepts they analyse is the possibility of multiple

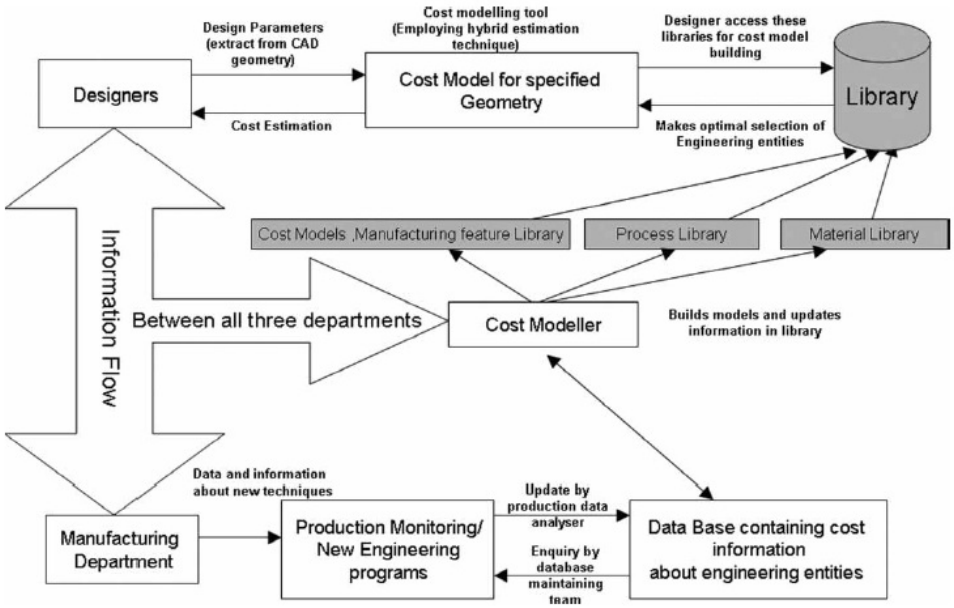


Figure 2.2: A knowledge-based system for cost modeling Tammineni et al. (2009)

streams of engineering working continuously, which together can deliver the necessary goal. The end result of the design which is produced is an area of the design space which all necessary parties agree they can achieve. The main advantages of this process over typical design iterations in series are:

1. A reduction in lead time since component design and manufacturing design are conducted in parallel rather than in a serial loop format typical of most design systems.

2. A right first time approach reducing wasted resources due to uncertainty of practicalities.

It should be noted that set-based concurrent engineering as described by Morgan/Liker (c2006.) (Figure 2.3) is different to design loops (the iterative model) although they are often taken to be synonymous. The iterative approach sequentially passes information between different designers where each team will accept or reject the modifications to the design requested by the previous teams until a time limit is reached (Figure 2.3). It should be noted that this is illustrative, and not an exhaustive set of the design activities. In set-based concurrent engineering different teams will generate and analyse many designs for their particular function (for example body, chassis and suspension) simultaneously; system engineers will then decide which set of designs will be used for the product, which must be located in the region of the design space which all relevant departments agree is feasible. To illustrate this there will be aesthetically pleasing and aerodynamically efficient bodies to a car which are perfectly feasible to make, but a practical chassis cannot be designed for them at an acceptable cost, designs of this type would be rejected by the systems

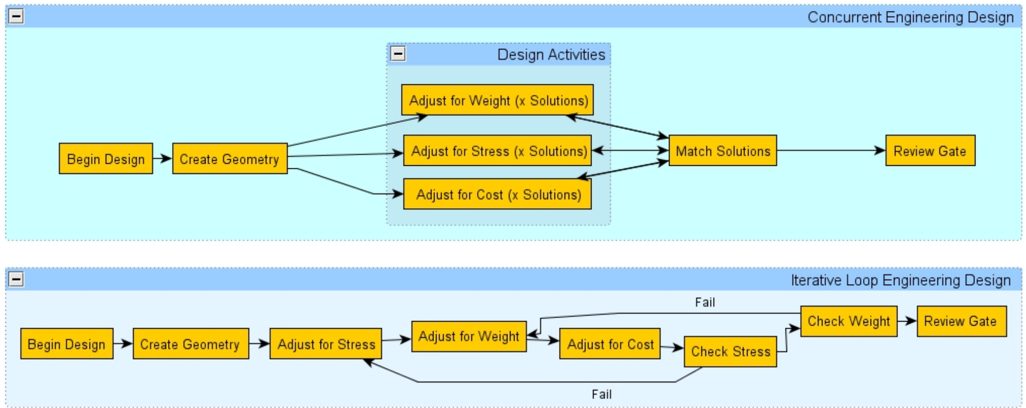


Figure 2.3: The iterative and concurrent design process models for (a subset of activities in) a generic aerospace design processes

engineers. If no set of feasible solutions can be found, the closest solutions are returned to the design teams to narrow in on a solution (thus resulting in some iteration). This approach is thought to reduce lead time, prevent the inefficiencies of loops and deliver an optimal solution rather than just a possible solution which is optimal only to the last team to alter the design before the cut-off time is reached.

It can be inferred that although concurrent engineering minimises lead time, it is inherently wasteful, as multiple designs which will not be taken forward are produced. It can, however, be argued, that it produces superior designs, as more are considered, and thus more flexibility is present. Typically iterative loop design only continues until a workable design is reached, which may or may not be optimal.

In the author's opinion, concurrent loop engineering is most useful for semi-independent systems. The greater the inter-dependence between systems, the more design effort must be expended. Concurrent engineering by its nature requires either: pre-defined interfaces and specifications affecting other components, or, teams must anticipate other team's decisions and prepare for them in their design.

2.1.2 Potential Design Methods

The purpose of a design process is to produce a design which satisfies all design requirements, including manufacturability at an acceptable cost. It must therefore be composed of features which can be manufactured, and should best satisfy the cost-value trade-off. As stated by Grabowik et al. (2005) a typical design process involves a design phase followed by a process planning phase after the design is complete. However, the effect of considering the manufacturing process after completing detailed design, is an increased difficulty and cost of design changes if changes are required.

Manufacturability can either be checked separately or considered as a part of CAPP.

Manufacturability checking without CAPP however typically assumes a known manufacturing route, which this research goes beyond. Two possible automated methods of checking manufacturability are stated by Di Stefano et al. (2004) are shown as 1 and 2(a) in the list, 2(a) and 3 are alternative options for the integration of CAD with manufacturing optimisation:

1. Produce the design as a solid model and use automated feature recognition to decompose the design into manufacturable features.
2. Design by feature using features which have at least one potential manufacturing route associated with them.
   (a) Either the designer can explicitly select the manufacturing method using a hierarchy.
   (b) Or the computer can choose a manufacturing method for each manufacturing feature.
3. CAD is conducted using design features, which are then converted into manufacturing features
   (a) The designer specifies a feature, geometry, tolerances and surface finish and the computer will use manufacturing rules to determine the optimal way of doing it, or present a series of options.
   (b) The designer specifies an intention and the computer will specify a feature, geometry, tolerances and surface finish and then use manufacturing rules to determine the optimal way of doing it, or present a series of options (Brunetti/Golob (2000)). This is discussed in Section 2.1.5.

There are advantages and disadvantages associated with each of these possibilities which are outlined in Table 2.1:

2.1.3 Features

The study of features is a wide ranging and extensive topic and necessary for both design for manufacture systems, and some design systems as well.

There are several definitions to be discussed:

Design Features These are features utilised by designers to achieve certain aims, for example:

- To provide structural rigidity

	(1) Sketch based (solid model) CAD and feature recognition	2(a:b) Explicit Manufacturing Method	3(a:b) Computer Selection of Method
Advantages	• Extremely flexible when configured • Allows designer complete freedom	• Relatively easy to implement • 2(b) includes optimisation	• Computer Selects manufacturing Process • Knowledge of method selection can be input by an expert • Does not rely on the designer having specialist knowledge • Can give results quickly • Increased productivity of designer
Limitations	• Possible issues with elliptical, complex components • Not the method of design Rolls-Royce wish to pursue • What to do when multiple feature combinations are possible • Possible computational issues (real time feedback to designers) • Feature recognition not fully solved	• Design Team selects manufacturing method (they may not have the necessary expertise) • New models necessary to take account of new manufacturing processes	• On what criteria does the computer select method (cost/quality/lead time) • Complex to write the manufacturing rules • Feature interactions • Tolerance interactions • High level of optimisation needed

Table 2.1: Advantages and Disadvantages with feature based design and feature recognition

- To provide aerodynamic/hydrodynamic surfaces to guide fluids or increase/reduce pressure
- To alleviate stresses
- To locate other components
- To enhance ergonomics/aesthetics

Manufacturing Features These are features created by specific stages in the manufacturing process, they fall into multiple categories:

Intermediate Steps, Serving no fundamental purpose apart from the fact that the operation stops there before another takes over. For example where a roughing operation ceases and a finishing operation starts. That is to say they are a consequence of the manufacturing rather than a driving factor.

Finished geometry features Features that are created by a single manufacturing operation that exist on the finished part geometry.

Fixturing Features Features required for fixturing purposes, for clamping/locating the component on capital equipment.

Reference Features Features required for referencing other features during production, for example a particular feature may have a tolerance defined against another feature which may/may not still exist when the part is finished.

The set of manufacturing features will of course determine the possibilities of manufacturing processes which can be used to make the component and also the potential sequences which are feasible.

2.1.4 Feature Recognition

Feature Recognition in this context is the process of automated deduction of manufacturing features from the designer's CAD geometry creation tool. However, it is worth noting that it can have several starting points:

1. In a feature based design environment translation from design features to manufacturing features is required. Translators have been developed including tools such as ontologies, spreadsheets and the AP-224 translator used by Xu et al. (2005) to translate between design features and manufacturing features.

2. Where features are created through the use of sketches rather than built in features of the CAD package graphical identification of the component is required, which may convert the geometry directly to manufacturing features or may have design features as an intermediate step.

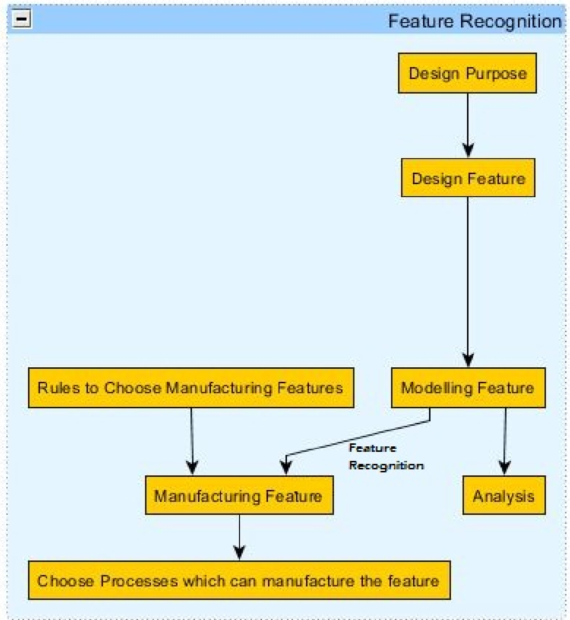


Figure 2.4: A diagram representing a feature recognition based design method

To convert design geometry into a component it is manufactured in general by a series of operations which create features using fundamentally different manufacturing methods. The reasons for these differences are partially due to the method of force application, for example a hole will probably be drilled or punched whereas a groove with rotational symmetry will be turned and a groove without rotational symmetry will be milled. It is clear that milling will have a lower material removal rate than drilling (assuming a constant size of tool) because the force exerted is a torque on the side of the tool rather than a compressive force. In turning however, the shape of the tool can allow shear loading and thus turning is faster in general than milling for a given removal volume. The access direction relative to both the part and the surface on which it is being machined is also a major contributing factor due to the loading on both the machine and the part.

The manufacturing features must be created from the difference between the final machined design geometry and the geometry created by the forming method of manufacture. There are several methods to do this: human conversion, Automated Feature Recognition (AFR) or the designer specifying the manufacturing feature during design (explored in the next section). Humans are naturally good at this operation for simple shapes as it can be done by intuition and experience, however, having people in the loop prohibits the vast number of evaluations necessary for optimisation and requires training and experience. According to Han et al. (2000) feature recognition takes CAD geometry with modelling features and converts to manufacturing features as shown in Figure 2.4. The three main approaches to feature recognition according to Han et al. (2000); Babic et al. (2008) take the finished modelled geometry from a CAD program and involve different techniques of extracting data from geometric models:

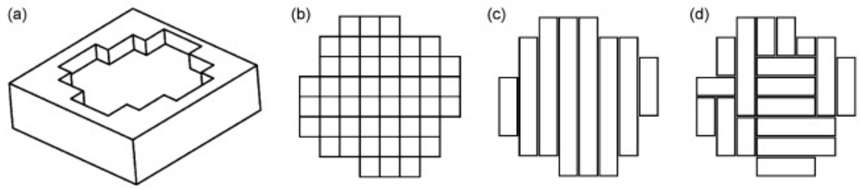


Figure 2.5: Volumetric Decomposition: Cell Decomposition Babic et al. (2008)

1. Volumetric decomposition
   (a) Convex hull decomposition
   (b) Cell decomposition
2. Graphical algorithms
3. Hint based geometric reasoning

The ways that these approaches work are shown below:

Volumetric Decomposition Works by applying volume identification algorithms to finished component solid model data.

Cell decomposition Works by dividing the space into many small cubic volumes, calculating if material is present in the small volume and then recombining empty volumes to create features, as in Figure 2.5. There are issues with complexity, particularly when the component involves curved surfaces as in Dong/Vijayan (1997a).

Convex Hull decomposition Works in a similar way to cell decomposition but with variable geometry polyhedral shapes instead of the cubic volumes characteristic of cell decomposition as shown by Dong/Vijayan (1997b); Kim (1992).

The fundamental problem with both of these types of volumetric decomposition rests in their inability to work with non-polyhedral type shapes (in other words any shape with finite curvature) unless a high density sample is generated, in which case vast data manipulation and some degree of approximation is required. To rectify this shortcoming Graphical Algorithms were developed.

Graphical Algorithms An early example of this approach was published by Joshi/Chang (1988), they define a face adjacency graph which maps the adjacent faces to determine if they are concave or convex. From this graph feature subgraphs can be extracted and compared against a set of predefined feature graphs to identify the feature. The algorithm recognises some feature interactions through splitting and re-joining graphs as necessary but there are still some issues; if a feature is not recognised then the algorithm can return a not recognised message in preference to an incorrect match.

A more advanced algorithm allows a wider range of relationships between faces (Di Stefano et al., 2004). This predefined geometry can take forms including arcs such as circular holes and slots with rounded ends. One of the methods reviewed worked by identifying faces and then testing all faces with all other faces to check whether the faces were:

1. Parallel
2. Perpendicular
3. Coaxial
4. Coincident

As shown in Figure 2.6, this approach identifies joins between two faces as secondary features and categorises them according to the angle between the faces. Once the faces and their relationships are identified, additional data is attached to each node (representing each face). The additional data is:

Number of loops indicates the total number of loops of the face

Number of handles indicates the number equivalent handles of the face

Concavity indicates the concavity of the faces. A face of the model can be convex or non-convex. A face of the model is considered non-convex if one or both of the principal curvatures in at least a point of the face are negative. Otherwise it is considered convex.

Geometry indicates the geometry of the face, in compliance with the taxonomy depicted in Table 1 of Di Stefano et al. (2004).

Geometrical parameters indicate geometric characteristic of a particular shape (e.g. the radius and centre of a sphere, the axis the base point and the radius of a cylinder), as stated in Table 1 of Di Stefano et al. (2004).

Face visibility indicates if a face is completely visible from, at least, one direction. A visible face has a nonempty visibility map. More details about accessibility and visibility of faces are provided in Section 4 of Di Stefano et al. (2004).

- **Total visibility** When all points on the face are visible along the same vector
- **Not visible** If there are points on the face which are not visible

Face accessibility Indicates if a face is completely accessible from at least one direction.

Once all of the faces have been identified there is more processing to identify what type of surface the plane is according to categorisation, then further processing to

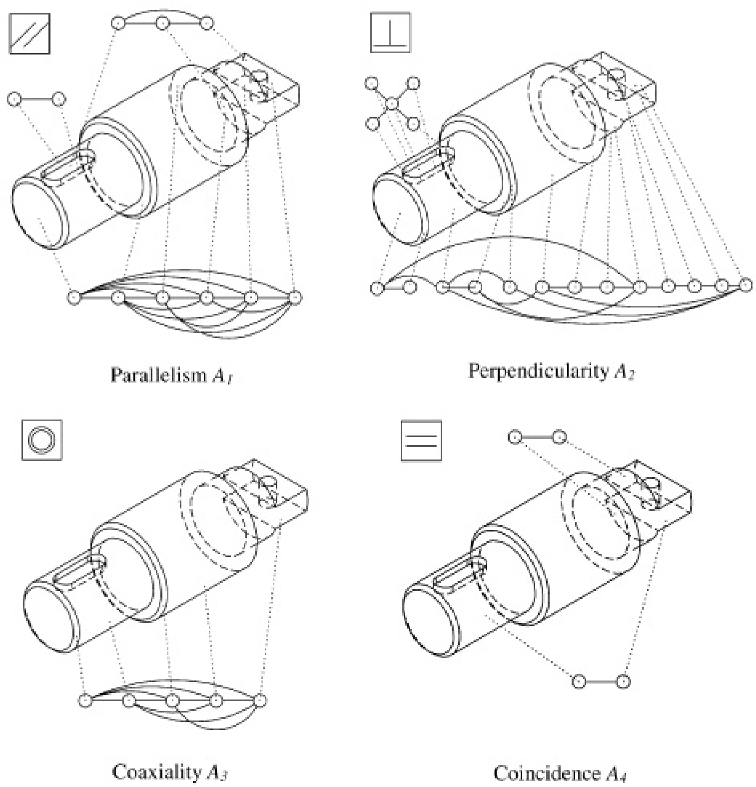


Figure 2.6: Graphical Algorithms: Face based analysis Di Stefano et al. (2004)

identity the relationship between groups of planes, labelled as features. Di Stefano et al. (2004) use a hierarchy of features, sub-categorising features until the end of the hierarchy is reached.

These methods have issues with intersecting features which change the original feature to become unrecognisable unless high tolerances are used, in which incorrect feature classification may occur, although Gao/Shah (1998) attempt to solve this problem by combining the graph based approach with hint based reasoning, discussed below. Essentially they search the design for isolated features whose graphs match predefined feature graphs, then any features which do not match standard graphs are matched to features by checking for: merged faces, lost concave edges and split faces. It should be noted that this approach appears to work only for right angle intersections. No non 90 degree intersections are demonstrated in the paper.

Hint Based Geometric Reasoning A similar approach to Graphical Algorithms but searches for traces of features which may be partially altered by other intersecting features. It searches for traces of features rather than attempting to match features directly. The difficulty with this approach is possible ambiguity. According to Di Stefano et al. (2004),

"The feature-recognition process begins from a solid model plus optional

> information as tolerance, attributes and functional form features. Features hints, which may result from particular combinations of part faces or by tolerance or attribute specification, are investigated and processed in order to generate the largest possible feature volume consistent with the available data."

We can see from authors such as Babic et al. (2008) that hint based reasoning is used to select the largest geometric feature which could be used to create an identifiable trace in the design. Several techniques have been used to provide an input to the algorithms including trimetric or orthographic drawings and the "viewer-centred approach" proposed by Sommerville et al. (2001) which takes several views of the component by taking an external point, drawing vectors out of the point and analysing where the vectors first intercept the component.

Similarly Vandenbrande/Requicha (1993) use an approach based on searching for partial geometry on features to detect features which have faces or edges missing due to other features. They restrict the possible features using criteria including accessibility, non-intrusion and presence. That is to say the feature candidate must not intrude upon the part, it must be fully accessible and it must leave a trace on the component otherwise there is no functionality from it. They also consider the use of material removal volume intersections through optional and required portions: required portions are those where only one feature affects that volume; optional portions are when more than one feature could remove the necessary volume.

Feature candidates are selected from pre-defined and hard-coded primitives (features) or composite features (sets of primitives). The system essentially generates features which may lead to the hints which it has detected from the geometry and then tests to see if the feature is valid and expands the feature to ensure the largest possible non-intrusive feature is tested; then it validates the resulting geometry against the real geometry, e.g. by checking that the required portions are indeed removed. This generate and test cycle continues, until the geometry which results from the machining is the same as the CAD geometry of the part.

An example feature recognition algorithm (Zhou et al. 2007)

A detailed algorithm for conversion of modelling features to manufacturing features is presented by Zhou et al. (2007) (see Figure 2.7):

1. Get the modelling features (F_0) one by one from the CAD model object using APIs provided by the CAD platform.
2. Extract the geometry entities of F_0, including its 2D sketch or profile (if it exists), and get the surface set of the created feature.

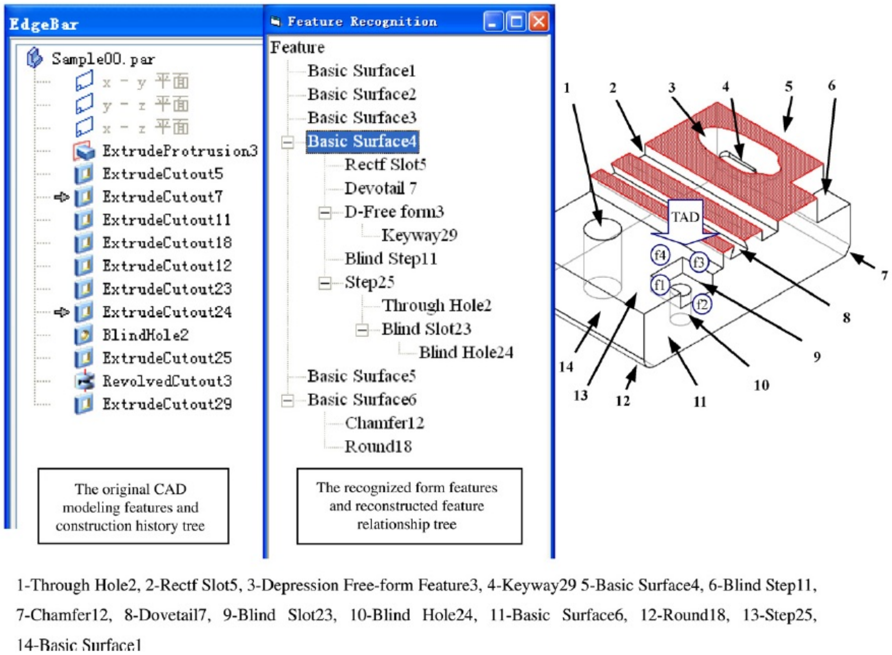


Figure 2.7: Feature recognition and feature tree reconstruction (Zhou et al., 2007)

(a) Identify surface adjacent relationships among the surface set by crossing the outside normal vectors of every two adjacent surfaces and dotting with the directional vector of the common edge to get the angle (constraint) between adjacent surfaces. If the angle is less than 180, two surfaces are adjacent through a concave edge. Otherwise, two surfaces are adjacent through a convex edge. In the case of a freeform surface, the normal vector at the neighbourhood near the common edge can be used to determine the concave or convex attribute, and the neighbourhood can be obtained along two intersection curves that are created by intersecting a plane normal to the common edge with two adjacent surfaces.

(b) Detect constraints between surfaces that do not join together directly by vector operation on outside normal vectors of such two surfaces; if their normal vectors are opposite, it means there exists a parallel constraint between these two surfaces.

(c) Analyse the adjacent character of the surface set, detect all subsets of surfaces that are joined together, and determine available kinds of form features (FF) created by the F_0 by comparing each subset of surfaces and the related constraints with the predefined form feature templates. If more than one feature is created, record their twin relationship for application feature mapping.

(d) Calculate feature parameters and extract the form feature instances, including calculating the distance between two parallel surfaces, inheriting parameters from F_0 and the parent feature to which the form feature is attached, and deriving the feasible tool access directions from the axis direction, modelling the operation direction of F_0, and/or outside normal directions of the corresponding surfaces in the surface set which are not sheltered from other faces of the geometry model.

There may be difficulties with this process, although it is possible to adjust the logic, due to the fact that, for an elliptical surface it is impossible to define a normal vector. Since there are an infinite number of normal vectors in the plane perpendicular to the surface.

It is interesting to note that according to Zhou et al. (2007) selecting the optimal machining method for each feature will not necessarily lead to the lowest total cost but only by optimising all machining choices together will the overall process be optimised.

According to (Di Stefano et al., 2004) typical issues with feature recognition include non-uniqueness of the model creation, that is to say that the geometric shape created by the designer in a CAD tool can be created by a designer in multiple different ways. It is for this reason that feature recognition algorithms use image data rather than model construction data.

It is thought that feature recognition is generally favoured by industry, not because it is necessarily the best method, but simply because it is a low risk approach. It has lower potential benefits but a higher chance of success than a completely new approach such as in Section 2.1.5.

A combined algorithm which conducts feature recognition and process planning was presented by Liu et al. (2014) but it is highly limited in the features it can use from the CAD package Siemens NX.

2.1.5 Design for Manufacture Systems

Design For Manufacture (DFM) and Design For Assembly (DFA) are twin methodologies whose aim is to guide the designer to produce a component definition which is cost effective to manufacture and assemble. According to Boothroyd (1982) methodologies such as these can result in substantial cost savings, reductions in time to market and lower probability of manufacturing defects. DFM requires one of two situations:

1. The designers have knowledge of the manufacturing capability available and design components within the capability of the supply chain to be used.
2. The designers are constrained in what they design such that any design they can produce is manufacturable.

Satisfying the former possibility through direct manufacturing knowledge is becoming difficult given the increased specialisation necessary by designers to achieve ever higher performance requirements. However, another way is to give the designer knowledge of the cost implications of their designs without requiring them to acquire manufacturing expertise, through the use of fully automated process planning and cost modelling system. This would enable the designer to test the cost implications of several possible designs to fulfil a design requirement without requiring manufacturing engineer analysis or for the designer to have expertise in manufacturing.

The latter can be too restrictive, prohibiting good but unusual solutions to issues such as weight reduction and raw material usage. Some methods of allowing restricted designs are elaborated on below.

Designing by Feature Library According to Xu et al. (2011) there are two ways of designing by a feature library. The first step is to determine the input shape and material; which may be a bespoke shape or may be readily available industry standard stock. This is likely to depend on the purpose of the product, production size and the specifications of the finished part.

Destruction by machining features Starts with the model of the raw stock from which a part is to be machined. The design model is then generated by subtracting depression features corresponding to the material to be removed by machining operations from the stock.

Synthesis by design features Synthesis by the design features method is built by both adding and subtracting features.

According to Hoque/Szecsi (2008) designing from a feature library is already implemented within software such as Pro/Engineer. This is becoming normal practice in other CAD packages such as Solidworks and NX as well. Hoque/Szecsi (2008) supplemented this library with manufacturing rules which will produce error messages if there is an issue with manufacturing. However, the features demonstrated are relatively simple, with limited interaction, such as single holes or slots. The features are arranged in hierarchies and the designer navigates down the hierarchy until they reach a final stage, and they supply the appropriate set of data, a flowchart of this method of design process is shown in Figure 2.8. In Hoque/Szecsi (2008) the designer is also required to select the method of manufacture before choosing the precise feature. An example of a feature library and a parameterised drilled hole are shown in Figure 2.9.

Clearly this method relies upon some form of concurrent engineering type practice (Section 2.1.1), since different engineers are required to write the rules for the required manufacturing methods in a database or similar knowledge repository. A similar approach has been taken by Hu et al. (2009) using DM (Design and Manufacture) features. They explore the link between taking feature drawn CAD geometry and CAM whereas Hoque/Szecsi (2008) restrict the scope to what is possible within manufacturing rules.

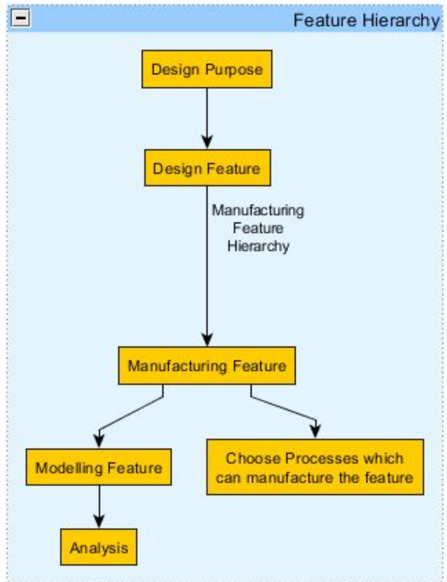


Figure 2.8: Manufacturing Feature Design Process Design Flow chart

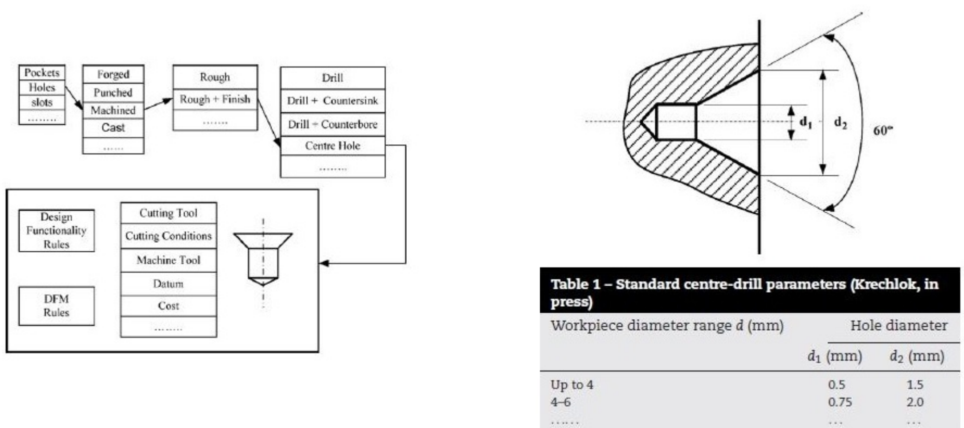


Table 1 – Standard centre-drill parameters (Krechlok, in press)

Workpiece diameter range d (mm)	Hole diameter	
	d_1 (mm)	d_2 (mm)
Up to 4	0.5	1.5
4–6	0.75	2.0
......	...	...

Figure 2.9: Designing Using Hierarchical Feature Libraries Hoque/Szecsi (2008)

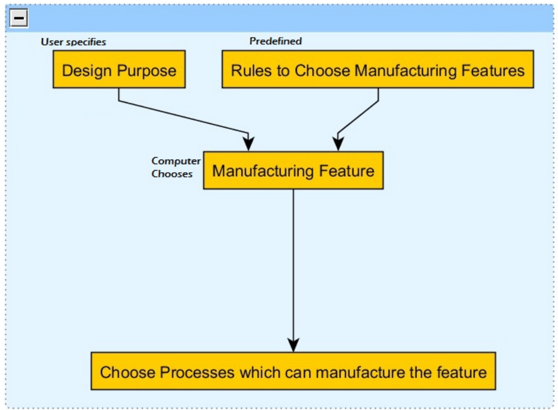


Figure 2.10: A design process where the computer selects the necessary design and manufacturing feature according to designer's declared intent

The primary issue with this approach is that the designer is expected to know which is the optimal method to produce a manufacturing feature, which may not be the case for anything other than simple components.

The authors Brunetti/Golob (2000), of a system of feature library design, envisage the data as being structured into four levels, the purpose and type of data within these levels is explained below:

1. Assembly Model
   (a) Contains the relationships between parts, and the reasoning behind it.
2. Part Model
   (a) This level of model contains the data linking the component parts together, such as welds or the data specific to the part such as material.
3. Feature Model
   (a) Feature models contain the generic data necessary to reproduce the feature, but in addition the functional description of the feature, such as its connection to other features.
4. Generic Model
   (a) Stores application data, this is data used to construct the model such as would be stored in current CAD tools such as Solidworks or NX.

The system presented by Brunetti/Golob (2000) (Figure 2.10) allows the user to present a problem to the computer, that is to say they will inform the computer what they want to do, and the computer will then present a list of options, i.e. how this function can be achieved. For example if two parts are to be joined together the computer will know they can be: welded, riveted, bolted, glued etc., and will suggest the different methods according to the functional requirements such as heat conductivity, strength of join etc. This is in effect the opposite to that presented by Hoque/Szecsi (2008) because whilst Hoque/Szecsi (2008) gives the user the opportunity to choose manufacturing methods based on experience and what they wish to achieve, Brunetti/Golob (2000) simply request that the user specifies what they wish to achieve and the computer can analyse the different methods of achieving it. This is advantageous because it means that the experts in manufacturing methods need not be in the design team but can simply write the rules that the computer will use when presenting a list of viable options to the designer.

Ontological Conversion of Features Even if it is the case that the design features within a part are known through manual construction by feature there is still the requirement to convert between design and manufacturing features. Design features typically have specific purposes and will still be present when the part is finished, manufacturing features however are simply a result of a manufacturing operation, which may or may not still be present in the finished part geometry.

Most feature recognition algorithms ignore this issue by recognising manufacturing features, although of course any manufacturing features not present in the finished geometry (Section 2.1.3) are challenging.

Thus if feature recognition is not used there exist some methods of converting design features to manufacturing features. One way is a simple mapping, the pre-specification of a design feature as a combination of one or more manufacturing features. However, even if the mapping from one to many is simple the mapping of dimensions is not. One reason for this is that due to the way that design and manufacturing are carried out. Designers will typically think of features which achieve a purpose, whereas manufacturers think of how to achieve the feature, either using subtractive or additive/fabrication processes. These two purposes require different dimensions for the same feature, either requiring some computational conversion rather than direct mapping of dimensions or a vastly over-dimensioned design with both required design and manufacturing dimensions present.

A more advanced alternative to this simple mapping is the use of an ontology as proposed by Anjum et al. (2012), they use ontologies for the purpose of manufacturability verification but the approach can be adopted for cost knowledge purposes. The central premise behind their hypothesis is that design to manufacturing feature mappings can be: one to one, one to many, many to one and fractions in either direction, in addition combinations of design dimensions may be necessary to obtain a single manufacturing dimension. Once they have obtained the manufacturing features they use a rules/logic based approach to determine the manufacturability of the features. It should be noted that their approach is component specific and at this stage needs to be hard coded for each component type with consideration for the tooling available in a specific factory. It would in theory be possible to make the knowledge more generic but this would come with a cost of either more computation time expended, less accuracy or both.

Agyapong-Kodua System Agyapong-Kodua et al. (2012) have focussed on the need for design for manufacture systems to take into account detailed information about the capabilities for the company or supply chain they are being used for, in addition to dynamic aspects such as resource availability. They also make the same point as has previously been made; this type of costing activity must be possible whilst the design decisions are being made, if the costing activity is conducted later it is either too late or it lengthens unnecessarily the design time causing a loss of competitive advantage. They also consider more business case decisions such as the make/buy choice but as with most of the literature are unclear how exactly they generate the process sequence for manufacturing the component.

DFM is part of a wider family of DFx methods reviewed by Agyapong-Kodua et al. (2013). They maintain that a design system should consider:

Design for:

1. Manufacture
2. Assembly
3. Maintainability
4. Environment
5. Quality
6. Decommissioning
7. Life-cycle

2.1.6 Value Driven Design (VDD)

Life-cycle methods including VDD (such as researched by Cheung et al. (2010); Zhao et al. (2014)) encompasses all of the above DFx methods, in addition to the profit generated for the operator (which is a function of performance) or customer of the product (in this case a turbofan). This type of holistic approach is a requirement of VDD, where the value of a product to a potential customer is the revenue the product can generate subtracting the cost. If a sufficient profit exists to compensate for the risk the potential customer would be expected to purchase the product. For an evaluation of the value of a product however all costs and benefits must be known hence this research into a small piece of this puzzle, the manufacturing cost. Once it is accepted that manufacturing cost is desirable information at the same stage it must be accepted that it is worse than pointless to know the manufacturing cost for a sub-optimal method of manufacture, this is where process planning (Section 2.2) can be of use.

2.1.7 Summary of the design process

In summary the design process is continuously evolving, with more information available earlier in the process enabling trade offs to be rationally made. This is allowing an ever accelerating design process and continual improvement in product performance, weight and cost. In order to remain competitive it is necessary to keep up with these trends and to continually conduct more analysis, earlier and with a greater breadth or exploration, lest a superior solution be found by a competitor. Although not addressed in this research it is necessary now to not only consider the cost of production of components, but to also consider their maintenance, lifespan and decommissioning in a range of industries.

This requires detailed modelling of production costs, MTTF, decommissioning and recycling costs in order to ensure a future loss is not made on today's production. VDD attempts to quantify this.

As an addendum however it should be ensured that any change of activity (such as new production) can have a beneficial/deleterious effect on all existing production of a business. Thus if practical this should be analysed as well. A good example of this could be the introduction of a new component into an existing factory, taking the factory over capacity and removing its resilience to unforeseen circumstances such as unplanned downtime. Equally the introduction into a factory operating below capacity can increase utilisation and thus reduce the costs of components already in production. These considerations are defined as out of scope but are briefly commented on in Section 2.2.16.

2.2 Process Planning

Process planning at its most generic is the creation of a manufacturing sequence, to convert materials from their naturally occurring form to the desired geometry and composition. Certain parts of process planning may be implicit, for example a car manufacturer may buy steel of a specified grade and shape, not iron ore. There are many levels of process planning for the manufacture of components, particularly complex components in today's globalised world. This is due in part to specialisation and increased competitiveness, the levels are:

Supply chain planning Analysing which company should complete which parts of the manufacturing process. Considerations here extend to the ability of the supplier to discharge their supply obligations, however, considerations such as responsiveness to unplanned deviations from planned volumes or design changes, ability to grow as required, respect for intellectual property, future strategic considerations, vulnerability to disruption will also play an important role in these decisions.

Process Planning The determination of the method used to produce the relevant features, for example, a choice on milling or turning, drilling or EDM (Electro-Discharge Machining). The middle level of fidelity shown in Figure 2.11. This level of detail is the one required for factory planning, as it shows how much time a given part spends on a given machine or type of machine. This is the technology used to convert from a design to a manufacturing route and is usually referred to as CAPP (Computer Aided Process Planning). This level of process planning must decide both the process to be used, and the order of these processes. This level of planning is generally conducted by a human-computer combination and is normally thought of as the process planning stage.

Tool Path Analysis Once the manufacturing route has been decided on a top level, it is necessary to obtain either CNC (Computer Numerically Controlled) code or a detailed instruction set, similar to the milling process shown in Figure 2.11, this is usually referred to as CAM (Computer Aided Manufacture), and is regarded as separate from CAPP. In the case of Rolls-Royce the software used for CAM is the same as used for CAD (Computer Aided Design), *UG-NX*. There may also be checking of the manufacturing tool-path, to ensure errors such as tooling collisions are discovered, for Rolls-Royce *Vericut* is used for this checking. It is very easy in principle to optimise and verify the CAM information, since the lower level of process planning is simply conducted by knowing the machining time for each feature, the path times between all features, and using a travelling salesman type algorithm such as presented by Papadimitriou/Steiglitz (1998) to find the optimal route. This will yield a time subject to smaller uncertainty. The uncertainty will come from manual parts of the process, and the potential requirement to slow the machining process to rectify unanticipated issues, such as vibration, which have not been discovered during the computational analysis. For this level the process type is specified but the order of operations is yet to be decided.

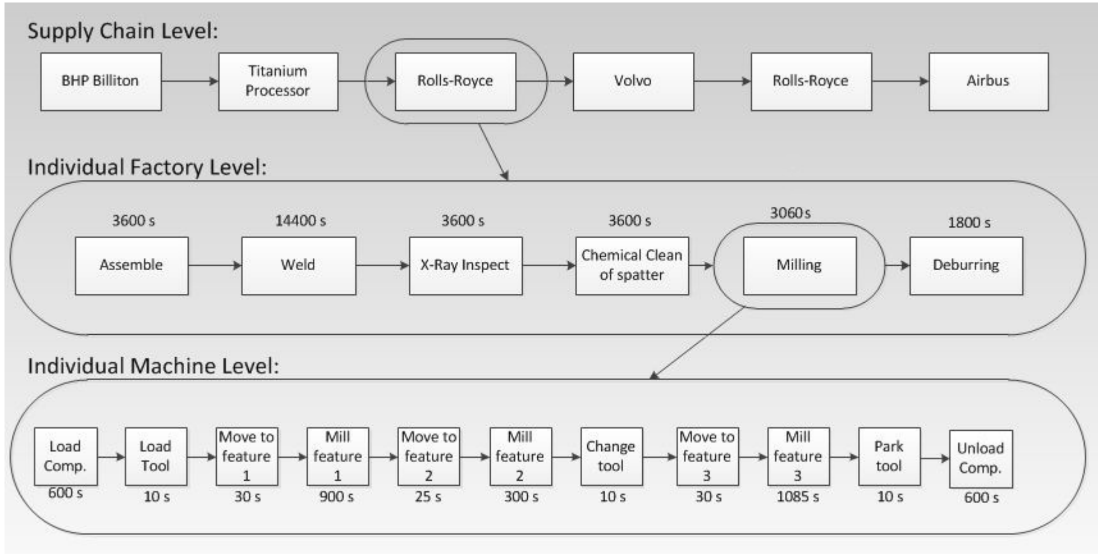


Figure 2.11: A multi-level view of a hypothetical supply chain with times of processes

2.2.1 Supply chain modelling

Much progress can be made by improving the supply chain, although there is much debate on what, exactly, constitutes a "good" supply chain. There has been a shift, from large industrial companies in the developed world designing and manufacturing their own products in their own country, to manufacturing their products in the developing world. Now, however, companies are subcontracting out the manufacturing, in an effort to reduce costs, as smaller, more specialised companies, can manage production more efficiently. There are two major disadvantages with this approach:

1. This lean supply chain can lack responsiveness in the event of unforeseen circumstances and the actions of competitors are more likely to disrupt it as suppliers tend to supply more than one downstream business, this has in fact been one of the driving forces for the process but can be deleterious during periods of high demand.

2. It has also led to a lack of knowledge about the cost drivers by designers. Designers know how to improve performance but they do not necessarily know its effect on cost. A part is designed, and bids are accepted for contract work and so the knowledge of cost drivers is not necessarily returned to the designers. This is to protect the supplier's intellectual property as a detailed cost model could be used to reverse engineer the production process.

One possible method of working around this issue of intellectual property is to solicit bids from a number of different suppliers after design, as done by Marion (2008), where the design was made available on the internet and suppliers were invited to bid for production by offering contract terms. However, this is only really suitable for companies producing simple products, where many suppliers have the capability to manufacture the product. In the aviation industry, due to the certification requirements, and the relatively small production volumes (relative to, for example, the automotive industry) there are a limited number of suppliers. Finding new suppliers is also a costly exercise due to the auditing activities involved. In addition, several design iterations are usually involved, and the supplier is selected prior to the final design stage. Therefore this approach is unsuitable for the high value or critical parts in aerospace industry. Some companies, including Rolls-Royce, are using cost modelling to predict how much supply chain components should cost, thus their negotiators can enter into contract negotiation, with an expectation of the costs faced by their suppliers. This can be done using past experience and statistical models, or by using generative cost models, however, it is just a prediction of what someone (who is not the supplier) company thinks it should cost, and this method may not use data directly from the manufacturer. In addition to the cost aspect of supply chains, one very important factor for the aerospace industry in particular, due to its long lead times and limited selection of suppliers, is risk. The issue of supply chain risk is focussed on in Ghadge et al. (2010) using a SoS (System of Systems) approach.

An article written by Pierreval et al. (2007) also engages in analysis of the supply chain. They use a continuous simulation approach, and logic to simulate the decisions of different sections of the supply chain. They have shown that continuous simulation can be a powerful tool, in anticipating and preventing issues such as supply shortages or increased lead times due to insufficient capacities, and can thus be a useful tool when choosing order volumes, frequencies and stock levels.

2.2.2 An Introduction to CAPP (Computer Aided Process Planning)

Computer Aided Process Planning (CAPP) is a computational interface between Computer Aided Design (CAD), and Computer Aided Manufacture (CAM). CAPP is intended to be the process which makes all of, or a subset of decisions. Petrovi et al. (2015); Isnaini/Shirase (2014) carry out extensive studies on the process planning problem, and generally agree on the necessary tasks a process planning system should perform, and what decisions it should be making. In particular they highlight the need for flexibility of operation choice, sequence, machine, tool and TAD and transportation costs. However, in addition to all of these, there is the consideration of fixturing/tooling, which may be a substantial portion of the cost.

The objective of CAPP is thus to construct a manufacturing process route which can be used to create a specific part within specified tolerances, however, there is discrepancy within literature as to which of these decisions should be made within CAPP. For example research by (Yang et al., 2011; Razali, 2014) assumes that CAPP begins after the choice of manufacturing processes. As will be highlighted throughout this section this is the case with much literature, or the method of process selection is unclear.

There are several categories of CAPP:

Manual CAPP CAPP where a human is used to make decisions, based on information and analysis provided computationally. In the opinion of the author these systems are typically little more than advanced spreadsheets. Whilst they typically provide an accurate answer for cost and production times among other data, the knowledge is typically less generic than in automated CAPP, and the routes generated by different experts are likely to differ. The results will often be sub-optimal but typically good enough.

Singh et al. (2011) published an approach for the selection of a manufacturing strategy or particular process. It relies on expert opinion to populate the matrices since they are filled with subjective opinion from a list of statements. It then evaluates these matrices to obtain a best process, from the point of view of Quality, Cost, Capability and Production.

Automated CAPP CAPP where the human is not involved after prior programming of databases or other knowledge repositories. This system requires computational decision making as well as cost and feasibility modelling. Automated CAPP comes in two major sub-categories (Yusof/Latif, 2014a):

Variant Where the process route is the same as, or similar to, a previously used manufacturing route or route options.

Generative The process route is typically derived from some generic manufacturing knowledge, typically less accurate and more computationally expensive, but more flexible than variant CAPP. The differentiating factor is that generative methods generate a method of manufacture (MoM), whereas variant based methods use a pre-supplied MoM.

According to Xu et al. (2011) CAPP systems require manufacturing features to be defined. However, it is more accurate to state that CAPP systems must either deduce the manufacturing features or they must be pre-defined. This is an important distinction because manufacturing features will often be different from design features, particularly when designers add, for example, an extrusion to a part normally forged. A manufacturer will think of adding extra material to the forging and then removing the extraneous material with machining operations. Thus either manufacturing features must be used during design (Section 2.1.5) or there must be a method of converting from design geometry to manufacturing features such as discussed in Section 2.1.4. In addition to this the design features may well be different from the modelling features used to construct them (Zhou et al., 2007).

Several authors have defined process planning and stated what they believe the output of a CAPP tool should be:

> "Process planning is defined as the activity of deciding which manufacturing processes and machines should be used to perform the various operations necessary to produce a component, and the sequence that the processes should follow." Marri et al. (1998)
>
> "A detailed process plan usually contains the route, machine and tool, processes, and process parameters." Marri et al. (1998)
>
> "Systematic determination of the methods by which a product is to be manufactured economically and competitively." Culler/Burd (2007)
>
> "Process planning deals with the selection of necessary manufacturing processes and determination of their sequences to 'transform' a designer's ideas (namely the designed part) into a physical component economically and competitively." Xu et al. (2011)

This section of the literature review will focus mainly on the choice of manufacturing route, although certain parts of the literature are more broad ranging, such as Xu et al. (2005), which also encompasses feature recognition, CAPP and CNC code generation.

A CAPP tool such as that developed by Sadaiah et al. (2002), must determine which process(es) and tool(s) to use for a given feature, and the order in which to perform the operations. To determine the outcome of these choices, the geometric and tolerance information of the manufacturing features is typically used. Sequencing is usually implemented through a set of constraints between pairs of operations. These constraints can be stored in a number of ways such as matrices from Xu et al. (2005), or a list/table such as that used by Salehi/Tavakkoli-Moghaddam (2009). It is interesting to note that some authors sequence features prior to ascribing processes to those features whereas some simply keep the constraints and allow the optimiser to find a solution.

The manufacturing process is totally dependent on the forming process chosen for the input material, although some parts of the process may be common to more than one forming process. Kumar/Rajotia (2005) take this into account by using standard sizes for axisymmetric components but of course the amount of material which must be removed is dependant on the difference between the forming shape and the finished part geometry. They also consider the power requirement of the tool to identify which machines are capable of each operation.

A creator of a CAPP system must consider which decisions the system should be making and how, what will be predetermined, and what will be left to downstream tools as options, explicitly or implicitly.

2.2.3 Knowledge Representation

A key factor in process planning (and generative costing) systems is the representation of knowledge. In a costing system, the knowledge typically stored will consist of equations used to calculate the times and costs of manufacturing processes as a function of geometry, material choice and tolerances. In a process planning system in addition to this knowledge, data on how to choose and intelligence on how to sequence the operations is also required. Consideration on how exactly to store this knowledge has been published by Grabowik/Knosala (2003). They, like most other researchers, have adopted a database type method because of its native ability to utilise complex query statements. They explore how to deduce the set of manufacturing processes and tools selection necessary to construct a specific feature using expert knowledge encoded in a set of databases, options are suggested to the user who may confirm or change the selection.

Other researchers to recently use database knowledge for CAPP includes Jinks et al. (2010); Tammineni et al. (2009).

2.2.4 A Review of CAPP Methods

According to Marri et al. (1998); Culler/Burd (2007); Xu et al. (2011); Prakash et al. (2012); Yusof/Latif (2014b) there are multiple approaches to CAPP systems:

Manual A manual approach uses human judgement, to make decisions regarding which processes, machines and tooling to use, and the sequence of the aforementioned processes. It typically results in a process plan similar to what would be produced by a variant based method. This is because humans are typically experience led, and thus more likely to choose a proven method over a novel, and possibly untested, methodology, provided the previous route is acceptable. There are many examples of this type of approach in the academic literature and in commercial use. In academia a good methodology was presented by Wasim et al. (2013), which allowed the user to make the necessary choices, but also provided guidance on how well the route met various throughput, cost, strength and scrap rate metrics.

Variant Based A variant based approach is useful when there is a known process for a particular component family, and this process can thus be replicated and modified (in a limited way) for other similar components. It typically involves interrogating a database after selecting the appropriate component family, and if necessary, sub families. For example on a component such as a gear with bolt holes, it is assumed that holes will be drilled, and that this will occur at a certain point during fabrication, the number of holes thus determines the cost of drilling them, there is variation but no wholesale change to hardwired manufacturing routings. This method is limited for novel components or changes in manufacturing process.

Generative An alternative to this method is a generative system which utilises geometry data to plan the manufacturing process. It does not store manufacturing process routes in a database like the variant method, but instead generates the manufacturing process each time it is executed. Thus it is more flexible, not requiring an existing process plan and does not rely on a human to correctly identify the appropriate component family.

These artificial intelligence methods rely on a knowledge base such as proposed by Park (2003) to allow decisions to be made and the structure of the knowledge base is shown below:

Facts Data Objects

Constraints What are manufacturing processes capable of.

Rules Control the way of thinking

Intelligence Implements the rules to create a process within the constraints using the facts

The knowledge base can have multiple levels, Gologlu (2004) presented an algorithm for finding near-optimal manufacturing sequences from all available process plans in a machining set-up using a four level hierarchy:

1. Feature level
2. Machining scheme level
3. Operation level
4. Tool level

There are, however, other forms of knowledge management which should be addressed. A process planner would typically take into account: machines, tools and labour; operation times, set-up times. Denkena et al. (2007) state that data about the supply chain/facilities and the work committed to should also be explicitly taken into account by the process planner. In addition they believe that commercial constraints should be considered, such as order size, payment terms etc. They structure this data in two ontologies: one for facilities, used for many parts and pre-existing; and one for data specific to the part being designed, geometry, tolerances and commercial considerations.

Generative methods can use many different methods of implementing AI (Artificial Intelligence). One method is a KnowledgeBase constructed as suggested by Zhou et al. (2007) and reviewed by Xu et al. (2011). Each of these optimisation techniques have advantages and disadvantages, however it is clear that due to the combinatorial nature of problems and the high number of dimensions, classical methods such as calculus and more recent methods such as surrogate models (such as Kriging) are of limited use.

A brief outline of the optimisation methods to be considered for use is shown below (Yusof/Latif, 2014a):

Neural Networks Approaches which learn either over time or using a training set of examples to give a higher likelihood of selecting highly used outcomes.

Genetic Algorithms Approaches which use quality based selection and propagation of superior solutions to converge to optimal answers through evolution.

Agent Based Methods Methods which apply intelligent agents, often competing with each other in decisions.

STEP Compliant CAPP CAPP which has two way compatibility with STEP, it can use real factory data to modify its decisions and code (relies on suitably equipped factories).

Graphical Algorithms Algorithms which make use of graphs and route planning methodologies, such as travelling salesman problem solutions, to conduct the process planning task.

Simulated Annealing An addition to genetic algorithms which reduces the probability or the selection of inferior individuals as the optimisation progresses.

Initially the author's research will focus on genetic algorithms and agent based methods may need to be used to improve convergence of the algorithm in order to reduce optimisation time.

It is interesting to note that Park (2003) considers the fact that expert humans will quickly find solutions to a process planning problem by eliminating infeasible solutions. If there is a computational way of eliminating swaths of the solution space, regions that are infeasible or obviously suboptimal, then this should help to increase the execution speed of the algorithm. Also they consider how to store the knowledge of choices. They use the example of a multi-diameter hole and use the concept of a final machining operation (which meets surface roughness and tolerance requirements) and pre-requisite operations to construct an operations list. It then constructs an order by using tool access requirements.

In addition it is useful to see that Lee et al. (2007) consider how to automatically generate feature precedence relationships which are necessary due to material deformation, nested features or referencing. They then optimise based on tool access directions, there is therefore the implicit assumption that the task can be conducted on one machine and tool change is less costly than part manipulation.

Another useful point of note is made by Han et al. (2001), who consider it necessary to conduct feature recognition and process planning together. Since unless all possible feature combinations are output by the feature recognition algorithm, the process planner may be over-constrained and either find suboptimal solutions, or fail. It may fail because a feature interpreted in one way, may give an unsolvable problem, due to constraints such as deformation or access directions, whereas a different interpretation may give a feasible solution. Their algorithm constructs and evaluates all possible methods of making the set of features, and makes use of knowledge of the dimensions, tolerance and surface finish, of the feature. It then applies the fact that certain operations such as reaming require other operations as prerequisites. Although all possible machining sequences are addressed in addition to constraints at both the feature and operation level it appears from the article that the individual operation sequences to achieve the feature to the tolerance and surface finish requirements are hardwired based on Manufacturing Knowledge. It is also apparent that emphasis is placed on handling time and machine/tool changes, rather than the actual machining time for the feature. Finally it is unclear how additional/new processes would be added to the database although it is assumed that the complete operation sequence would be added, as an additional possibility. The algorithm attempts to minimise tool changes, and the overall number of tools used. According to Xu et al. (2011) this enabled the problem of operation sequencing to be systematically addressed.

Another approach to the problem of manufacturing process selection was presented by Cheng (2011), who details the use of queries and formulae which can be used to determine process choice, feed rate and tool choice etc. This can replace the use of highly skilled and experienced people for producing process plans for production. It is more detailed, but not as intelligent as other methods involving GAs presented earlier. Many methods now require little or no human input during run-time but this still requires human input during the planning process. This method, of course, requires high levels of maintenance, probably by people skilled in both manufacturing and coding but does have the advantage of reducing computational run time as the design search space is heavily restricted.

A more recent piece of research based on a similar concept which also allows the construction of multiple stages of manufacturing process for a single feature was conducted by Zhang et al. (2014), using Semantic Query-enhanced Web Rule Language (SQWRL). This is a powerful and expandable approach.

Additional research focussed on process planning focusses on integrating it with additional technologies such as by Culler/Burd (2007) for factory simulation or Wang et al. (2013); Bettwy et al. (2014) for factory scheduling.

In some cases, authors of literature such as Zhang et al. (2008) allow the consideration of multiple factories, but there is no fundamental difference apart from the necessity to consider transport costs. Profit margins can be considered as just a percentage change in time/volume based cost rates.

2.2.5 CAPP methods: Variant Method

One paper which incorporated CAPP using an assembly simulation was Culler/Burd (2007). Their platform is shown in Figure 2.12, however the process planning was limited to the ordering of the processes. The choice of processes was predefined and the times thereof was estimated by the Industrial Engineering function of Bombardier Aerospace.

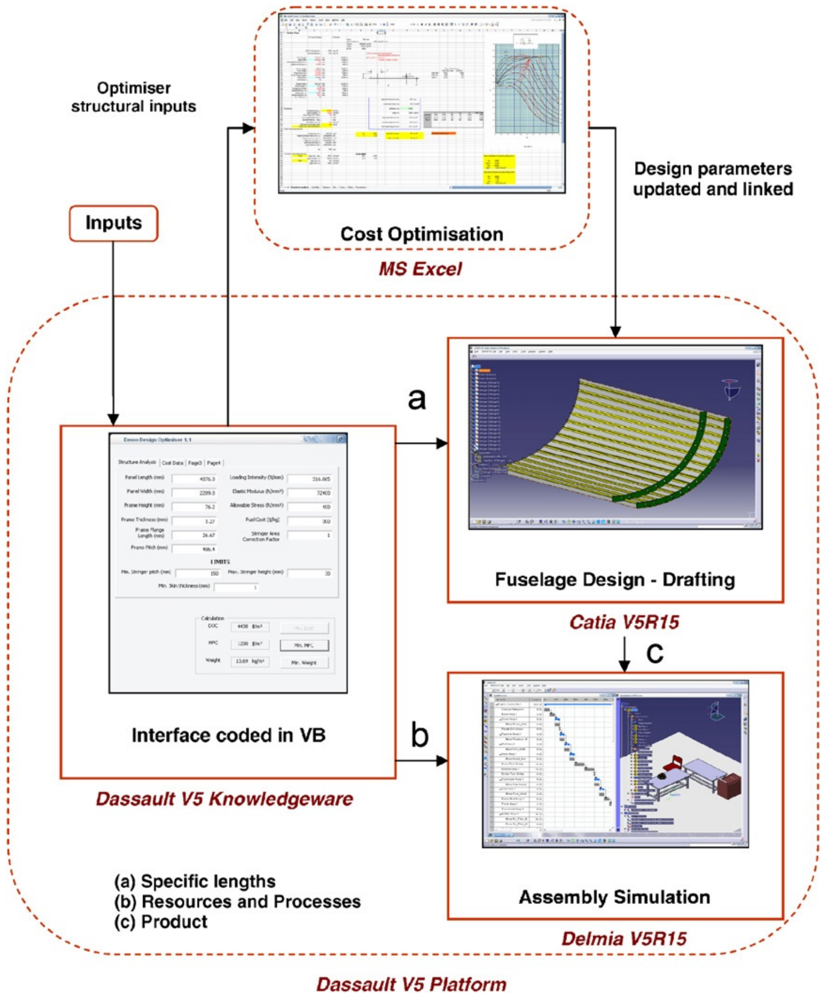


Figure 2.12: The Delmia V5R15 Assembly Simulation in the Dassault V5 Platform (Culler/Burd (2007)) showing an optimiser complete with discrete event simulation.

Particular issues with the variant based approach are:

- The passage of time and progress of technology may have rendered the baseline [process plan] obsolete or suboptimal.
- The new component is sufficiently different from the baseline component to render the original process plan unsuitable (for example different material, geometry or tolerances).
- The baseline process plan may never, in fact, have been optimal and this is still not known by the process planner.

2.2.6 CAPP methods: Infeasible solutions

It is likely to be the case that some features require others to be machined first. It is obvious for a human that finishing a face requires the face to be created first. Firstly there must be a way or storing/deducing the precedence constraints such as the method proposed by Zhou et al. (2007) which decides precedence based on:

Datum Datum features (used to locate others) are machined first

Family Parent features are machined before their children

Process Roughing features are machined before finishing

Concentration Concentrated similar features are machined together (this is not a constraint but it is generally desirable)

Once these constraints are defined a methodology must be created to identify the satisfaction or otherwise in a given process plan. If they are unsatisfied then according to Su et al. (2015) the main methods of solution are: repairing the process to a feasible one after generation and before/during objective function evaluation, or creating the process plan in such a way as to not create infeasible routes. It is a powerful approach but is unclear regarding the possibilities or selection of machines or tool access direction.

A combination of repair and graph based route construction was explored by Huang (2013) where the initial population contained only feasible routes by construction, if infeasible routes were created during recombination and mutation they were then repaired. It was limited to a single operation per feature.

2.2.7 CAPP methods: Genetic Algorithm

Some of the earliest proponents of the use of a Genetic Algorithm (GA) (Goldberg, 1989) to conduct process planning were Rocha et al. (1999), they attempt to minimise manufacturing time whilst producing only feasible routes. As with many of the later papers, it is unclear how the operation list is generated, or whether the GA can add operations. However, the authors do show that the selection of machines and tools, in addition to the sequence of operations is controllable by GA chromosome modification.

Salehi/Tavakkoli-Moghaddam (2009) used a GA to optimise process plans for a part predefined in terms of features.

The methodology they follow is:

1. The feasible sequences are created by GA
   (a) By applying order constraints and clustering constraints
   (b) Optimising for the maximum satisfaction of feasible constraints according to a weighting matrix.
   (c) Each entry of the GA contains a feature number (as in 1, 2, 5, 4, 6, 3)
   (d) Produces n feasible solutions for input into the next optimisation sequence.

   The crossover operator used was developed by Gorges-Schleuter (1989) which prevents the formation of illegal children from legal parents, an issue found with TSP type GA solution methods. Mutation is conducted through an exchange operation.

2. The method then optimises, sequences, and selects which tool to use.
   (a) Starts with the feasible solutions from the prior optimisation.
   (b) This time each entry in the GA vector contains vectors/strings of machine, tool and TAD
   (c) By applying additive constraints to feasible sequences.

   The Crossover operator for this optimisation originated from a paper by Davis (1985) but has been developed for vector individuals.

An approach building on this was presented by Kafashi/Shakeri (2011), however, whilst these algorithms are good at informing how to optimise a list of operations given a set of constraints, the arrival at the list of operations is ill defined. It appears to be based on tolerances and the type of feature, however, it is thought that they have created the operations list, as shown in Figure 2.13, by using human knowledge rather than an automated approach.

Amin-Naseri/Afshari (2012) build on the use of genetic algorithms, within a process planning algorithm, by considering the scheduling problem posed by multiple parts passing through a factory simultaneously. They pre-calculate the operation precedence graph for each job (component) and also pre-calculate the time of the operation for each potential

Feature	Feature type	Operation(Op_id)	Machine candidate	Tool candidate	TAD candidate
F1	Pocket	Milling(Op1)	M2, M3	T5, T6, T7	$-z$
F2	Blind hole	Drilling (Op2)	M1, M2, M3	T2, T3, T4	$-z$
F3	Through hole	Drilling (Op3)	M1, M2, M3	T2, T3, T4	$+z, -z$
		Reaming (Op4)	M1, M2, M3	T9	$+z, -z$
		Boring (Op5)	M3, M4	T8	$+z, -z$
F4	Four Through hole	Drilling (Op6)	M1, M2, M3	T2	$+z, -z$
F5	Slot	Milling (Op7)	M2, M3	T5, T6	$-y, -z$
F6	Slot	Milling (Op8)	M2, M3	T5, T6	$+y, -z$
F7	Two Blind hole	Drilling (Op9)	M1, M2, M3	T1	$+y$
F8	Two Slot	Milling (Op10)	M2, M3	T6, T7, T10	$+y, -y, +z$
F9	Pocket	Milling (Op11)	M2, M3	T5, T6, T7	$-x$
F10	Pocket	Milling (Op12)	M2, M3	T5, T6, T7	$+x$

Figure 2.13: A list of operations used as the input to a GA optimiser routine Kafashi/Shakeri (2011)

machine, they are also attempting to maximise throughput, a different objective function. In addition, they make several simplifying assumptions that are not made by this research: negligible transport and setup times, or at least independent of sequence, also processing time is known and fixed.

They also convey a method of tuning genetic algorithm control variables developed by Franca et al. (2001), which is effectively derived from the Hooke and Jeeves numerical optimisation method for functions. This methodology of holding all but one of the optimisation variables constant and iterating over all variables is a simple and effective local search to find the best parameters, although local optima in the parameter search space could be problematic and multiple problems must be tested to give the best general parameters.

Another paper of particular interest is written by Wang et al. (2011). They also explore the use of genetic algorithms but only for the evaluation of feasibility. They have not gone so far as to actually create the manufacturing route using backward and forward propagation and have not evaluated the sequence for cost, instead using a surrogate of complexity. The chromosome definition appears elegant, although at a cost of flexibility and expandability, and they have used crossover and a population. Furthermore they do not consider scrap/rework/tolerances and it is unclear how they handle crossover between chromosomes of differing length, in fact it is unclear that they allow them at all. Huang/Hu (2011) use a multi-level G.A. to select from a pre-defined set of manufacturing process chains for each manufacturing feature in an outer optimisation ring, then sequences them using graphical methods for satisfaction of constraints. As with many other papers they do not consider costs apart from machining time, nor how to construct process chains in-situ. Presentation of their results is also simple due to the fact that fewer cost parameters are used.

Zahid/Baqai (2013) also explore the use of genetic algorithms to construct and evaluate process plans, their evaluation is limited to the consideration of setup and tool change costs. They do highlight the option of reusing previously generated manufacturing routes to accommodate new features, which could potentially be a useful addition, but it is unclear how they account for the potential to change the optimal manufacturing method for existing features.

Zhang et al. (2015a) highlight the issues of the distributed supply chain with many options, i.e. the new opportunities when manufacturing is not constrained to a facility or even a single company. They highlight the need for a more complex encoding of individuals, mutation operators which can operate on them and an efficient decoding methodology in the objective function.

Xu et al. (2014) consider the creation of multiple stages for a single Feature Machining Element (FME), however, they do not show how these stages are derived, or how the times are calculated, i.e. fitness evaluation.

2.2.8 CAPP Methods: Graphical Algorithms

Graphical algorithms such as that proposed by Huang et al. (2012) work by constructing an operation precedence graph, and combining this approach with genetic algorithm optimisation. The GA ensures optimisation whilst the graph exists to force feasibility. The feasibility algorithm used is iterative checking and switching of operations where a constraint is breached, and they do not consider grouping constraints at all. This paper also uses a repair function after each iteration, to ensure that all manufacturing sequences are feasible. There is a stochastic method of machine and TAD (tool access direction) selection controlled by the central optimiser. Finally, it is clear that this tool is simply a sequencing tool as no thought is given as to how to select the set of manufacturing processes or how to automatically create the process precedence constraints. Huang (2013) do begin to consider the selection of processes utilising their previously published graphical approach (Huang et al., 2012), but they do not use the recursive propagation found in other literature and this thesis (Section 2.2.10).

2.2.9 CAPP Methods: Simulated Annealing

Simulated Annealing (SA), as used by Lian et al. (2009), works by initially allowing fitness of individuals to decrease and thus allows exploration of the design space, but progressively discourages descent to lower fitness (or ascent to higher energy states). It is based on annealing in materials science by gradually reducing the thermodynamic free energy available to promote crystalline reorganisations. The particular strength of SA is that the optimiser can traverse areas of decreasing fitness in order to find the global optimum whereas a pure hill climber requires a particularly fortuitous mutation to achieve this when there is high epistasis in the design search space. A GA (Genetic Algorithm) with SA was proposed by Lian et al. (2009) and was shown to be substantially better than a GA without SA.

Lian et al. (2009) perform both manufacturing process selection and sequencing, they consider the type of operation, machine used, the tool access direction, and the specific tool used. They also perform an exhaustive search for each design feature to find all possible combinations of manufacturing processes, machine, TAD and tool. They have a repair algorithm to find a feasible solution if one exists which operates by checking constraints and moving a successor operation if it breaches a constraint to behind the predecessor.

Their hybrid algorithm essentially works by putting the best result from a GA generation into the start point for SA. Only when the SA process plan is better than all the results from the GA is the SA algorithm used. Effectively the GA populates the start point for the SA algorithm the SA can be seen as a local search which is not always a hill climber, though its propensity to descend to lower fitness values reduces as time progresses.

2.2.10 CAPP Methods: ANN (Artificial Neural Network)

Zhou et al. (2007) proposed a method based on backward propagation (BP), to create a list of processes and constraints, then using a GA to optimise the sequence. The constraints include datum precedence, feature precedence and process precedence although it could be argued that datum precedence is simply a sub-category of feature precedence.

The way that BP works, in detail, is the algorithm selects a manufacturing process for a feature based on a trained ANN (Artificial Neural Network), and then works back through the processes necessary as pre-requisites, according to probabilities which are known by the neural network from training, by manufacturing process experts. This builds on the work by Park (2003), which uses the concept of a final machining operation using tolerance and surface finish requirements.

In addition, the algorithm takes into account machines, machine cost rates, tools, tool cost indices, and tool access directions.

Khalifa et al. (2006) suggest the use of neural networks for the selection of machine tools, it is possible that a neural network approach could be used, either within the optimiser to learn over time, or with an external data input, to improve the convergence rate. Bringing the neural net inside the optimiser and making it learn which manufacturing processes are used most frequently for specific features would also further differentiate the work from Zhou et al. (2007), who used a pre-populated neural network.

2.2.11 CAPP Methods: Logic and Query based

In addition to the aforementioned optimisation based methods above, Grabowik et al. (2005); Zhang et al. (2010) and others have promoted logical methods of process selection. This will optimise a component on a feature by feature basis, which may result in a globally sub-optimal solution. It will, however, be much faster and potentially sufficiently good for industrial applications. This type of method requires far more extensive calibration of the database as rules must be coded rather than simple options. Given the speed of this type of approach Grabowik et al. (2005) incorporated with it a sequencing planner designed to react to disturbances, for example an offline machine. Their system could quickly re-route a component to minimise the loss of output by selecting alternative manufacturing methods or sequences. The methodology used by aPriori (Section 3.2) uses logic and query based methods.

2.2.12 CAPP Methods: Other

Other methods have been developed for optimisation of the manufacturing route.

One of these was developed by Lv/Qiao (2013), who construct an AND/OR graph detailing the manufacturing routes, and machine possibilities, for each manufacturing operation. This graph can be manipulated to give all feasible sequences, but it is this author's belief that with the levels of complexity dealt with in this research the graph would not only be unintelligible, but excessively computationally complex to create. One of the aspects that they do consider however, which is dealt with differently in this research, are the intra-factory transfers. Using a machine list they specify the transport times between all pairs of machines, the times are, however, independent of the current value and mass of the component.

Another method, by Prakash et al. (2012) used a Knowledge-Based Artificial Immune System (KBAIS). They considered the re-sequencing of a series of operations where the set of operations was fixed, but the order was not, and the machine, tool and TAD choice was also made by the optimiser from a pre-defined list. In addition they claim novelty in considering scrap to be a factor of which machine was chosen and calculating the costs on this basis.

Tabu searches have also been explored by Li et al. (2003) for the CAPP optimisation problem as a way of escaping local minima as an alternative to simulated annealing, Tabu search can be considered an extension to simulated annealing optimisation.

2.2.13 Difficulties with CAPP

Issues with CAPP software tend to be incompatibility of data structures; some of this is due to fundamental differences between the way that designers design components and the way they are made. Due to various factors including structural requirements, it is often the case that components start as a forged CoS (Condition of Supply) and are machined down to the correct dimensions. Thus manufacturers tend to think in terms of forming and subtractive mechanisms whilst designers may think of additive processes, for example adding an extra feature. In some cases such as slots or holes a designer may think in the same way as a manufacturer, but this is not always the case.

Despite much progress in understanding how to model the CAPP process, and information flows which are necessary, there has not been as much progress on the actual integration of fully automated Process Planning systems. According to Zhou et al. (2007) this is in part because of the difficulty in converting CAD features to manufacturing features. Zhou et al. (2007) propose feature recognition as a solution to this and have implemented this solution in UGS/SolidEdge.

The reason Zhou et al. (2007), do not favour manufacturing feature based design is the same reason stated by Scanlan et al. (2006), namely it would restrict the designers and prevent interesting and potentially useful experimentation because they cannot (or it is difficult to) create unusual parts in their design tool. The other issue, according to Zhou et al. (2007), with feature based design is that intersecting features which may be separate in a feature based design tool may actually combine as manufacturing features.

Thus they favour feature recognition despite the current limitations, particularly issues resulting from the storage of data on tolerances, surface roughness and hardness, which can affect manufacturing cost substantially, affecting machining time and the process which can be used. One way of extracting this non-geometric data, presented by Prabhu et al. (2001), is using natural language processing intended to read the notes intended for human interpretation from the model file, and combining them with features before the data is output as a part model. Features are then defined together with their GD&T (Geometric Dimension and Tolerance) information. The way the text is matched with the appropriate feature involves the use of recognition of arrows and the location using encoded logic within the programming rather than artificial learning techniques such as neural nets. Another way of storing the information, which also takes into account the “datum story” was explored by Shah et al. (1998) by storing the GD&T information within a feature based design model. It decomposes dimensions and tolerances into three dimensions and deals with each separately. Thus, provided it is aware of the tolerances and what they measure, it can calculate, in principle at least, the ultimate deviation of a feature. This type of approach may also be useful when calculating material removal volumes of intersecting features.

Issues with a GD&T system are difficulties in making it suitable for both CAD and CAPP. This is the essential difficulty in comparison between design and manufacturing features as outlined in Section 2.1.4, since the simplest way to design a feature is rarely the simplest way to manufacture it. The further challenges posed by feature extraction according to Zhou et al. (2007) are:

1. Recognition of intersecting features.
2. Handling multiple interpretations of features that correspond to different ways to machine the part and therefore provide downstream applications with added flexibility.
3. Controlling computational complexity.
4. Feature constraint extraction.
5. Feature precedence tree reconstruction.

2.2.14 ASP (Assembly Sequence Planning)

A subset of problems within process planning is assembly planning. It is related in that joining methods must be chosen and sequenced, and the criteria are similar: to meet the design specifications as cost effectively as possible. The issues are normally access constraints or difficulties, weld deformation, join strength and access for later maintenance.

According to Yu/Wang (2013), ASP is generally conducted by ACO (Ant Colony Optimisation), GA (Genetic Algorithm) or PSO (Particle Swarm Optimisation). Their chosen approach of ACO works by encouraging the use of well-travelled paths, although the algorithm was adapted to maximise computational efficiency. ASP generally requires knowledge of access difficulties and the authors differentiate between functional components and connectors, connectors are purely in existence to locate the functional parts. They maintain that this algorithm does take care of subassembly problems, but it is unclear how. One major issue of process planners for assemblies is when are manufacturing operations carried out, if they could be done before or after assembly operations. This occurs when design features are still accessible after assembly operations, indeed to ensure continuity of the surfaces and structural rigidity during assembly it may be desirable to conduct some operations afterwards. However, this question is incredibly difficult to formulate and adds another layer of complexity to a problem which is already computationally challenging.

The algorithm Yu/Wang (2013) propose essentially attempts to minimise changes in orientation and will try to group similar operations together but taking into account hard constraints such as access but also the softer constraints caused by friction and ease of assembly, a solution worthy of consideration.

Esterman Jr/Kamath (2010) also propose metrics for the improvement of the assembly sequence. Firstly they consider the optimum planning of an assembly line with regards to the number of workstations and operators, they then consider the potential improvement by relaxing precedence constraints, that is to say component redesigns which make alternative assembly sequences feasible. They analyse the assembly sequence with the objective to improve: a) assembly time, or b) load balancing.

A particularly large assembly planning problem in the shipbuilding industry was explored by Iwakowicz (2016), they use precedence constraints and a genetic algorithm to explore the design space in order to minimise the assembly time.

2.2.15 Process selection and sequencing

The processes which can be used to produce a given feature are dependant on the factors discussed in Section 2.3. Thus if this information is in the design system, the design system can identify which processes can be used to produce feature x. Some of the manufacturing operations will be due to features which exist in the drawing, however some are due to other operations. For example a weld operation which will appear in the drawing may result in mandatory full x-ray checking of weld lines. Many machining operations are subject to dimensional inspection, but wouldn't be if the machine had an independent position checking probe, since it could check dimensions at the same time as the machining were occurring and thus correct under machined defects before the component leaves the machine. It is also clear that an inspection operation would necessarily be following the machining operation which it is inspecting so the approach taken by Sormaz et al. (2010) is to keep the machining and inspection as a single part activity which requires multiple machine activities. They also mention the fact that machine cells are normally configured to minimise inter-cellular transfers, however, this is not necessarily always the case. Where the cost of inter-cellular transfer is only slightly more than intra-cellular transfer and machinery can be run by a partially absent operator it may be more sensible to minimise operator movement, thus reducing the employee numbers necessary to operate the factory.

2.2.16 Production planning (Scheduling)

It is worth noting that the process planning task cannot be completely isolated from production planning, and in an ideal world they would be optimised together (Zhang/Wong, 2015; Bettwy et al., 2014). Leus et al. (2014) in particular highlight the benefit of similarity between product lines and create a method whereby manufacturing routes for multiple components undergoing simultaneous production can be optimised together to minimise total cost. The optimal process plan should take into account the availability of the machines and labour required to conduct the manufacturing. Additionally, in an ideal world the process plan of any/all existing components passing through a facility should be re-examined if a new part is introduced so as not to sub-optimise the facility (Morinaga et al., 2015). Given that it is challenging to even optimise a process plan for a new component passing into a factory the next step is aspirational at best for anything other than simple and small test cases.

In most cases process planners produce multiple/flexible process plans and the production planner is left to select from them such as by Tavcar et al. (2013); Li et al. (2014a); Wan et al. (2013). The obvious disadvantage of this type of approach is that the level of flexibility of the generated process plans vastly increases the design search space. Resulting in a trade of between time, flexibility and number of components. Clearly the more complex the manufacturing route for each component or the more components to be considered simultaneously, the worse this trade off becomes.

Wen et al. (2013); Wang et al. (2013) also follow the flexible process plans type of approach, with the addition of using various distributions of times instead of a deterministic calculation in their models. Musharavati/Hamouda (2015) also take a set of process plans for multiple parts and reconstructs them in such a way as to optimise for multiple parts simultaneously, it relies on preconstruction of a set of possible process plans. Musharavati/Hamouda (2015) do highlight a method of resolving the issue of variable length chromosome recombination.

In aviation the cell scheduling problem is explored by Yu et al. (2006).

2.2.17 Summary of process planning

It is clear that most process planning research is conducted on generative process planning. There is variety in exactly which decisions are made computationally, which are made by humans in real time, and which are prepopulated decisions.

There is extensive research on methods to sequence the manufacturing operations, less so when the requirement to choose a manufacturing process is included. However, one area of limited research in particular is the issue of multiple operations for a single feature, most researchers limit themselves to one operation. Of the ones who do not, in most cases they consider a pre-determined set of manufacturing processes rather than allowing the methodology to choose the processes. It will be demonstrated in Section 5.4 why this is inadequate.

There is no consensus on how to resolve the issue of infeasible solutions. Although graph based approaches have proven themselves capable of generating only feasible routes, repair based algorithms are also capable of the conversion of infeasible solutions into feasible ones.

Additionally consensus has not yet been reached regarding the optimal algorithm, or combination of algorithms, required to sequence the operations for minimum time or minimum cost. Although genetic algorithms, frequently combined with other methods, have shown promise.

In terms of the selection of machines during the process planning the most powerful methodology was created by Zhang et al. (2014), their use of Semantic Query-enhanced Web Rule Language allows vast flexibility in the conditions applied to choices, such as machine or tool choices.

The area of compound scrap and its incorporation into process planning has not been explored adequately.

The issues of assembly sequence planning are defined as out of scope in order to concentrate on component manufacturing, equally the dynamic aspects of production planning are disregarded.

2.3 Integrating CAPP with CAD

CAPP generally requires more data than CAD, particularly information about tolerances and surface finishes. In addition, CAPP is generally conducted using manufacturing features whereas CAD generally uses design features (either explicit or implicit within the mind of the designer). Therefore in general it is necessary to convert the data. This can be done either using structured data and a translator or feature recognition, or a combination. Proponents of the latter include Kang et al. (2003) who propose neutral file formats for the interlinking, particularly the STEP AP formats. Proponents of the former such as Leo Kumar et al. (2015) use XML and feature based modelling, to avoid the need for feature recognition, and attempt to combine this data with a knowledge based system, however, their research is limited to prismatic micro parts.

There is also much research modelling the process by which the linking should be done. According to Feng (2003), several process models are available which describe how to produce manufacturing process planning data, there are several pieces of information generically necessary to select a manufacturing process:

1. Product Model Data
   (a) Geometry
   (b) Material (including necessary treatments such as nitriding, carbiding or shot peening)
   (c) Tolerance
   (d) Surface Properties
2. Process Related Data
   (a) Manufacturing Resource Availability
   (b) Process Capability Specification

A significant number of the models reviewed use the IDEF0 modelling language, which describes the necessary information inputs to a task, the control mechanism, the resources necessary to complete the task and the information it provides. It can thus be used in hierarchy form by decomposing activities into sub-activities multiple times, which is an essential requirement in complex process planning for human interpretation. However although it describes the actions and information necessary for a complex activity it does not necessarily detail how it can be done.

The modelling that Feng (2003) conducted does covers both CAPP and CAM as it:

1. Takes CAD data

2. Derives machining features
3. Process plans the manufacturing route (considering feature type and tolerance)
4. Process plans the factory floor route (Given the options available for manufacturing, on which workstation should the operations be completed on.)
5. Creates NC programs
6. Checks NC plans

It is said by Zhou et al. (2007) that most researchers use neutral file-based solutions to address the issue of the transfer of data between CAD and CAPP and CAM. The data is transferred using STEP (simple ASCII coded files) or IGES (Initial Graphics Exchange Specification) files (which are also ASCII encoded) to transfer data, although XML has been used more recently, notably by Sormaz et al. (2010). The difficulty with this is that the communication is usually one-way, thus a separate method is necessary to give feedback to the designer.

Zhou et al. (2007) follow a several step process.

1. Get modelling feature F_0, from the CAD tool
2. Convert to design feature F_D, through a feature recognition algorithm.
3. Convert to manufacturing feature F_M, which could take one of two forms:
   (a) Form Features: Pre-defined features such as holes and slots etc. with analytically calculable material removal volumes and associated manufacturing methods.
   (b) Surface features: Refer to surfaces which are not part of a recognised form feature.
4. Manufacturing Process selection
5. Operation Sequencing

The way that Zhou et al. (2007) choose manufacturing processes when more than one is available to machine a feature is a back propagation Artificial Neural Network (ANN). This relies of course on knowing the final finished form of the component and expert evaluation of data sets for neural network training according to:

- Feature
  - dimension
  - tolerance and

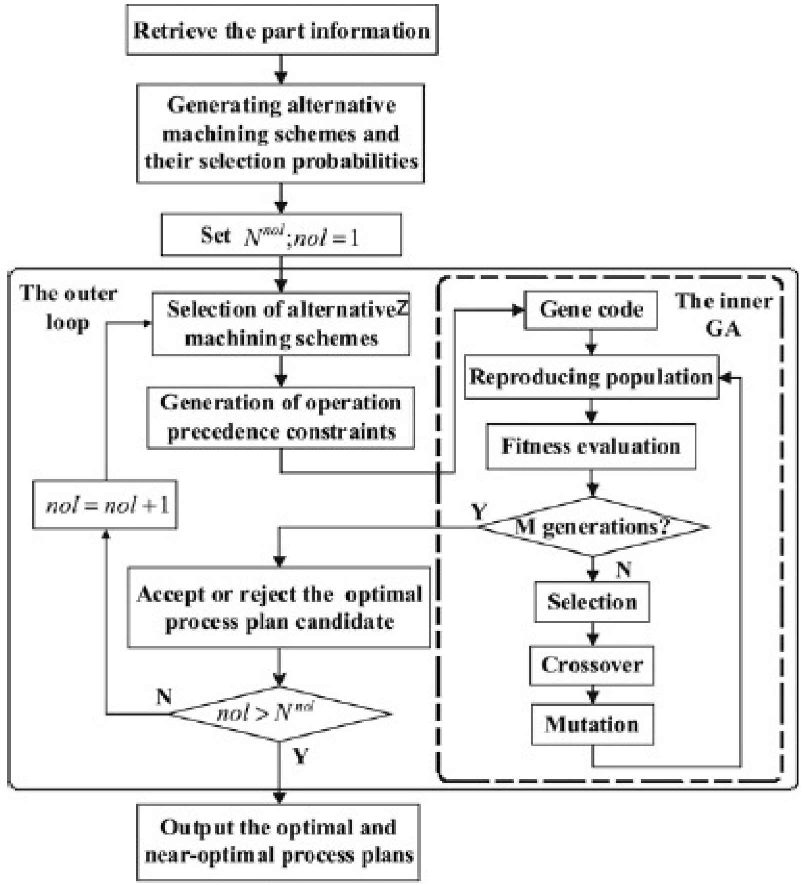


Figure 2.14: Flowchart of the GA-based approach for process planning and optimization Zhou et al. (2007)

  - surface roughness of the feature

- Component
  - batch size
  - material
  - stock type
  - heat treatment
  - structural form

Manufacturing Process Selection and Operation Sequencing are optimised sequentially for cost using a GA (Figure 2.14).

A more recent and alternative method of integrating the CAD and CAPP activities was published by Sormaz et al. (2010) where they use a feature based modelling approach where the features contain sufficient information to allow CAPP to work without the feature recognition stage as shown by Sormaz/Khoshnevis (1997), in addition to the feature model however they believe it is necessary to maintain a solid model for geometric analysis and the geometric model would be required in any case for analysis such as FEA or CFD. Sormaz/Khoshnevis (1997) also have a feature precedence network (FPN) to aid in the sequencing of operations. Some operations can be performed in any order but in some cases, either because of access or other constraints some features must be machined before others. In addition to a feature precedence network they use a manufacturing precedence network to denote which order manufacturing processes can/must be used, for example hole starting drilling is necessary before drilling and then finishing the hole.

2.3.1 Summary of CAD-CAPP integration

Feature recognition or design by feature (in their broadest sense) are clearly the only alternatives which exist for converting design geometry into the features necessary for process planning. They both have issues, feature recognition mainly with intersecting features and design by feature with the potential for reduced design flexibility if insufficient features are defined.

This area will be defined as out of scope and design by feature will be used to provide a geometry driven input to the process planning methodology developed.

It is also worth highlighting that geometry is not the only required input to process planning; material choice, production volumes, date of commencement of production and type of part are also important factors, although this is not an exhaustive list.

2.4 Prediction of precursor shape

It is noteworthy that a fundamental consideration in process planning is the question "what do I start with?". Without an answer to this question, only the times of processes independent of volume can be predicted, and these are few and far between. As stated by Masel et al. (2010), there are several classes of methods to predict the forming shape and these will be outlined below:

Intuitive An expert, potentially assisted by a computer program (Caporalli et al., 1998; Gronostajski et al., 2015), generates the forging shape. Therefore (Masel et al., 2010) these are semi-automated systems such as outlined by BISWAS/KNIGHT (1976), including: Cubic Splines, exponential functions and geometric transforms.

Analogical

- **Expert** An expert, potentially assisted by a computer, generates the forging shape using past designs.
- **ANN** An artificial neural network can be trained on a set of data to recognise patterns, and derive forging shapes for new components. However Masel et al. (2010) highlight the fact that if the ANN is exposed to a class of component not encountered in the training set, it may give an inaccurate or unrealistic reponse.

Parametric Using parametric rules the forging shape is designed, and constrained to the shape of the finished part geometry. Examples could be:

- **Face-line offset** A simple offset which assumes a simple, fixed, offset of n mm to a face (3d) or line (2d) Toal (2012).
- **Smoothed** Similar to above, but ensures a smooth shape without low radius curves Toal (2012).

Analytical Using rules, but at a more detailed (feature based) level, the forging shape is estimated, and responds to the shape of the finished part geometry. Can be more flexible than parametric methods, but requires more time and expertise to build a generic rule set.

Rules Based Generally more accurate than the above (parametric methods), but requires generation of a rule set such as outlined below, in order of ascending complexity and usefulness:

- This can be for a specific component type such as high pressure compressor disk. This has been completed by Meas (2015).
- A more generic rule set which could work on component class such as disks. This has not been found in the literature.

- A truly generic rule set which could work on any component geometry. This has not been found in the literature.

Simulation An analytical type of approach where instead of rules applied to geometry, a flow simulation is created. An example of this was published by Caporalli et al. (1998). They use Ansys (v5.3) to simulate the metal flow, however, as they state, the system relies on experts to modify the forging and process sequence.

There are also two different levels of rules based methods in the literature: simple rules applied consecutively to the finished shape and, more frequently, simulation based time dependant finite element flow models, to predict detailed mechanical properties of the forged component.

A simple rules based approach was presented by Masel et al. (2010). They developed this methodology to provide a rapid estimate of forged shapes, and therefore cost of the forging, since they state existing methods were extensive computational exercises. Existing methods, they claim, were designed for detailed design of die shapes, making them unsuitable for iterative design work. Their method relates only to axisymmetric shapes, and relies upon knowledge of the finished part geometry. The rules are relatively simple and applied sequentially, but need to be coded for each manufacturing strategy. The resultant volumes do correlate strongly to known forging volumes from a sample of parts, but it is thought that a more generic method is desirable than one specifically designed for forging. No mention is made of its accuracy, when applied to other manufacturing strategies.

A detailed thesis into a rules based approach for blisks and discs, which was patented and adopted by Rolls-Royce plc, was written by Meas (2015), in which he highlighted the difficulties in accurately predicting the forging shape. His methodology centered around creating an ultrasonically inspectable Condition of Supply (COS), and from that deriving an appropriate and accurate isothermal black forging shape, from which the COS could be machined. The methodology behind the creation of the black forging shape was to sequentially apply rules (obtained from Semiatin (2005)) to the COS shape until all rules were satisfied.

The rules applied were drafts, corners and fillets, webs and ribs, flash and trim. A diagram of the process used by Meas (2015) is shown in Figure 2.15.

One of the factors of interest is that Semiatin (2005) showed that parameters such as fillet radii are material dependant. This will have an effect on precise design but not necessarily a significant effect in early design, when more inaccuracy is present in other factors.

Papers presenting forging shape prediction methods using detailed simulation include those by Kulon et al. (2006); Bramley/Mynors (2000); Biba et al. (2001). Biba et al. (2001) highlight the importance of disallowing the user from adjusting parameters of the

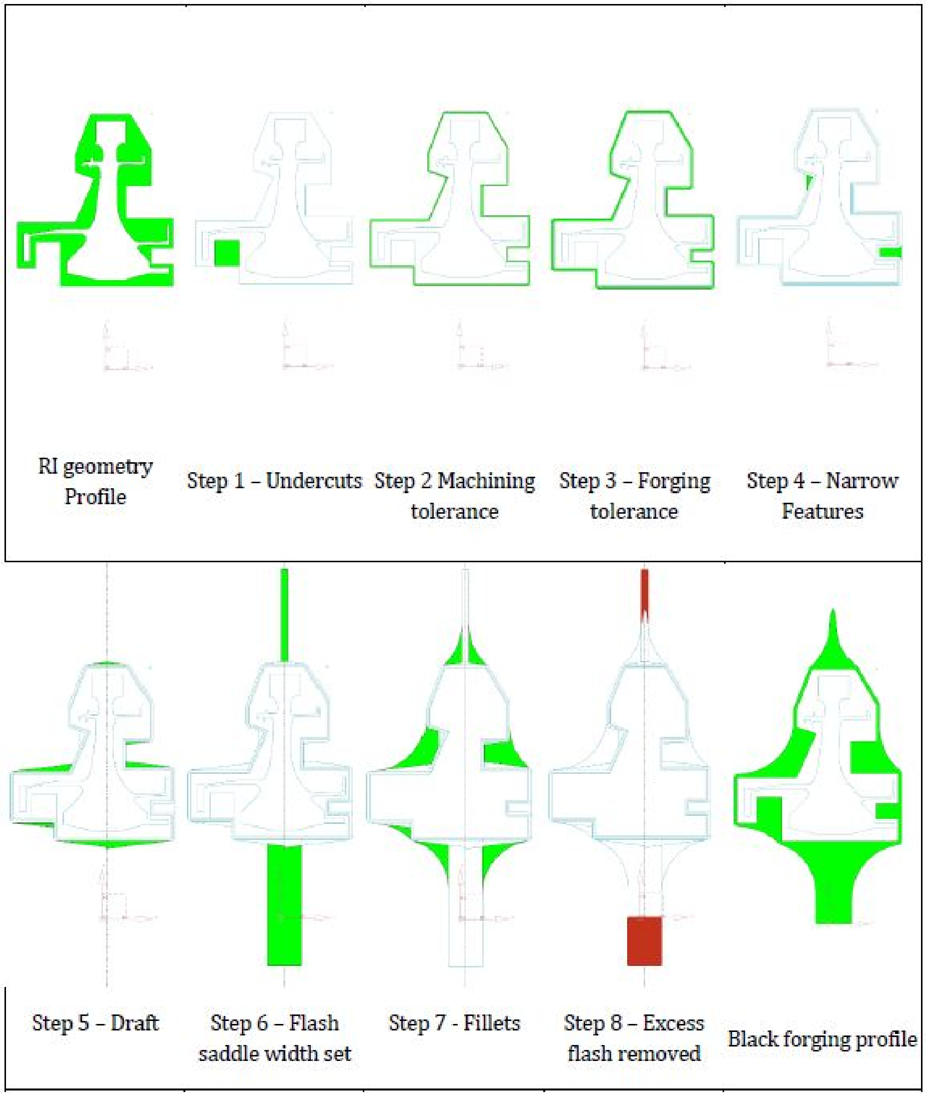


Figure 2.15: The approach used by Meas (2015) (Figure 51) to predict forging shape from an ultrasoncially inspectable COS

simulation, thus preventing the quality of its outputs becoming dependent on the skill of the user, but only the quality of the software. Masel et al. (2010) similarly mention the disadvantages of allowing an expert to vary parameters in terms of the inconsistancy of responses.

The more detailed simulation models could in principle be used to estimate the forming shape, to feed into cost modelling, but to obtain an accurate result they generally require significant computational time. Equally a relatively simple rules based approach such as presented by Masel et al. (2010) could be used, however this approach is currently limited to forging discs and blisks. Thus this research will attempt a new method, to be explained in Section 4, in an attempt to reduce the necessary maintenance of the models. The methodology will attempt to provide estimates for a range of forming processes, not just limited to forging.

2.5 Manufacturing Processes

It can be said that in the past manufacturing was generally a subtractive process, particularly in load bearing components so a manufacturing engineer will generally think in terms of starting with more raw material and subtracting the sections which are unnecessary. However, a designer can use either additive (extrusion) or subtractive (shelling or lofting) methods to achieve the desired shape. In addition, modern methods of manufacturing can include additive processes and more advances methods of forming such as powder bed forming, which should be considered by a process planner.

Additive manufacturing using material deposition is possible, as used in rapid prototyping, and a recent development was published by Karunakaran et al. (2010), the authors used a CNC milling machine retrofitted with a weld deposition capability, thus both additive and subtracted processes can be used in sequence. Although as stated by Skiba et al. (2009) the structural properties of additive processes may be inferior to those of cast or forged components due to factors such as HAZ (Heat Affected Zones).

There are many types of manufacturing processes, and they can be generally classified into several categories: material removal; material forming; joining; clean up; surface treatment and inspection.

Another useful way of looking at some of the manufacturing processes is through a hierarchy as created by Sormaz et al. (2010), which has been expanded and is shown in Figure 2.16.

2.5.1 Material Removal

Material removal generally consists of multiple stages. Primary (or rough) machining with high material removal rates, then fine machining with lower material removal rates and higher tooling costs for finishing work due its superior surface finish and reduced process variability. A large amount of material may be wasted although much of it can be recycled.

1. Turning is a tooling intensive operation, particularly on hard materials such as titanium.
2. Milling machines have a lot of wasted time due to tool changes or CNC manipulation of components. They are also intrinsically slower due to the method of force application; a direct result of the shapes they are designed to machine.

It is possible to obtain mill-turn machines which would reduce the set up time as they can perform both operations but they are more costly than each of the two individual machines.

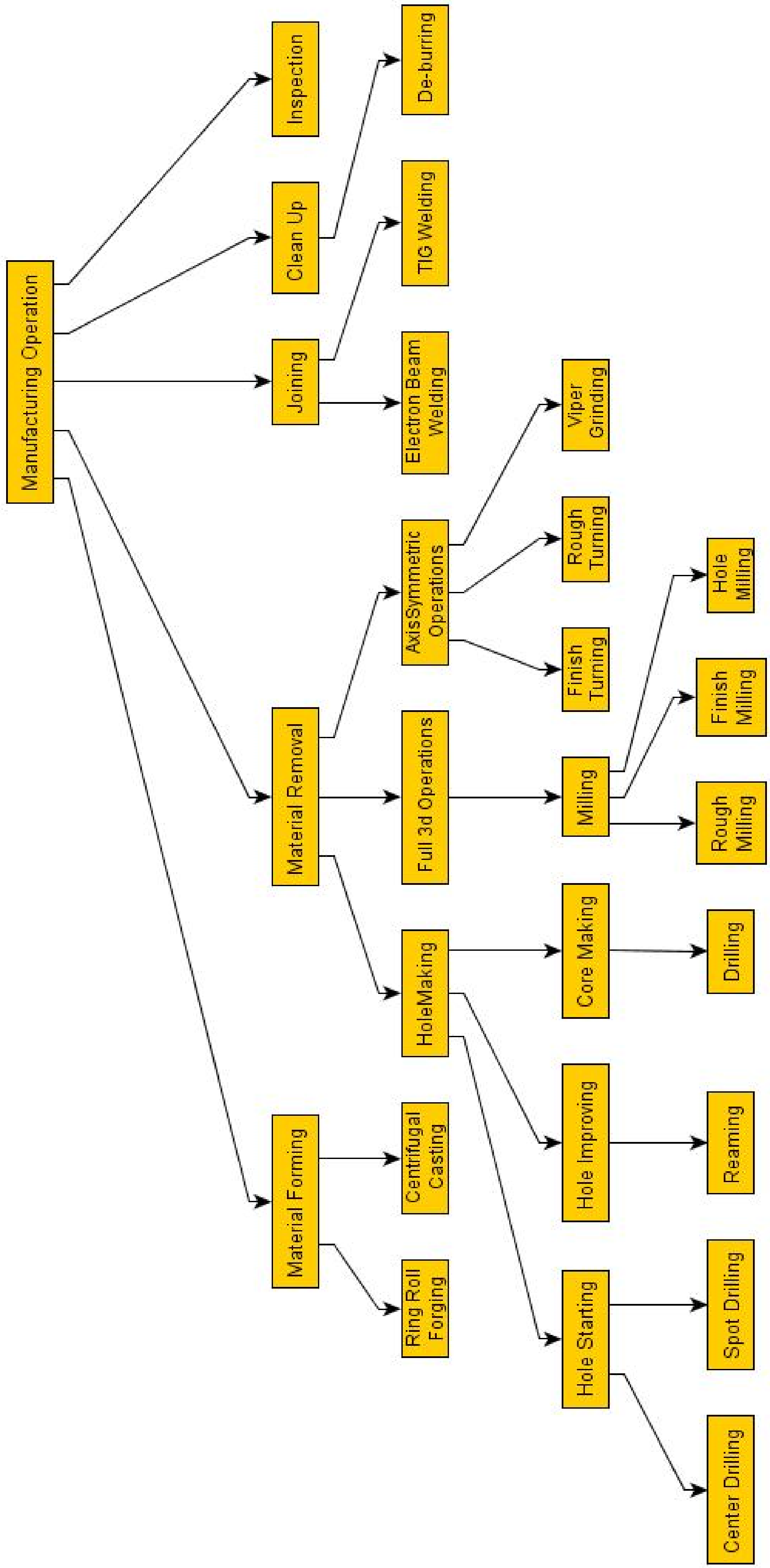


Figure 2.16: A manufacturing Hierarchy

Material removal processes are normally followed by a dimensional inspection and clean-up operations such as deburring.

2.5.2 Joining (Welding)

Electron Beam Welding is used for high quality welding of titanium. Machines can automatically track the join lines, manipulate the object being welded, perform both structural, cosmetic, penetrating and non-penetrating welds. However their limitations can be weld angles, beam dumping requirements, spatter (particularly at high power), they also induce high latent stresses in the component (though only on a very localised area), and cause consistent distortion of the component.

TIG (Tungsten Inert Gas) welding is a manual process, inexpensive and results in good quality cosmetic welds and no weld spatter but the process is slow and results for structural welds can be inconsistent, depending on the skill of the welder. There is also a relatively large heat induced stress zone because the weld rate is slow.

The welding process involves assembly, welding, clean up and an x-ray inspection of a proportion of the total welds, the extensiveness of the inspection depends on the criticality of the component.

2.5.3 Manufacturing processes modelling and optimisation

There has been much research specific to one type of manufacturing process, to understand the process, model it and in some cases optimise it. This aspect of research is defined as out of scope but this type of approach in optimising individual manufacturing processes should contain/create models which could be data-mined to produce the equations necessary for input into other methodologies such as presented in Section 5.

For some forming processes a major contribution to the cost on low volume production is the cost of manufacturing the dies, this is particularly applicable to components forged of extremely hard materials/high temperature materials such as nickel molybdenum alloys which are required for the high temperature and high stress areas of missiles and gas turbines. The work published by El-Mounayri (1997) could be used for this purpose to estimate the die production cost, which could then be amortised over the expected lifetime to improve the accuracy of the forming process cost modelling.

Some modelling research into forming processes is exemplified below.

- Forging by Behrens et al. (2011); Gronostajski et al. (2015)
- Centrifugal casting

- Die casting
- Power HIP (Hot Isostatic Pressing)
- Zhu et al. (2012) for continuous casting
- Park et al. (2006) for composite manufacture
- Taufik et al. (2011) for MMC (Metal Matrix Composite) manufacture
- AML analysis by Thomas et al. (2016); Zhang et al. (2015b)

Subtractive manufacturing processes have also been explored

- Chiu/Weigh (2012) used them to optimise micro-milling
- Laser based processes have been explored by Wiedenmann/Zaeh (2015)
- Milling processes by Leo Kumar et al. (2014); Li et al. (2014b); Ouyang (2015)
- Drilling by Li et al. (2015)
- Optimisation of Electro Discharge Machining (EDM) by Azadi Moghaddam/Kolahan (2015)
- Tolerance and cost of Electrochemical Machining (ECM) is analysed by Ayyappan et al. (2015)
- Turning Sortino et al. (2015)

Finally joining processes

- Linear friction welding Tao et al. (2008)

The detailed optimisation and analysis of individual manufacturing processes is out of scope of this research. This type of approach in optimising individual manufacturing processes should contain/create models which could be data-mined to produce the equations necessary for input into other methodologies such as presented in Section 5.

2.5.4 Fixturing

It is noted from the research of Ribeiro et al. (2012) that tooling design and specification can have a significant effect on the processing time during manufacture and thus a trade off can be made between unit cost and non recurring costs (NRCs). Generally the more capable the fixturing is, the faster the part can be manufactured in general (provided that component deformation during machining is the limiting factor, often the case in aerospace applications). However, a more capable fixturing is generally more customised, and less reusable, and thus the non recurring costs must be amortised over a smaller number of parts.

Equally a more capable fixturing system can be developed such as by Nasr et al. (2015) but of course this requires reconstruction to be changed from one fixture design to another, i.e. the setup costs may be raised at the end of a batch of parts to convert from one to another.

The methodology this thesis describes could take this effect into account through multiple approaches but it is left to future work for this to be implemented.

2.5.5 Summary of manufacturing processes

Limitations of manufacturing processes can be used to identify which processes can be used to create certain features and tolerances combined with process variability will then determine the probability of successfully creating the feature(s).

From the point of view of cost, the manufacturability of components is highly important. Work by authors such as Sanz-Lobera et al. (2015); Dodd et al. (2015) reviewed existing methods and presented proposals on functions to predict the effect of changing tolerance on the manufacturing cost of a component.

In addition the research conducted on manufacturing processes can be used for the purposes of modelling their cost. Either directly or indirectly if the research proposes time generation models. In some ways time generation is better, given that it allows the user of the model freedom to set their own cost rate assumptions, which are in any case somewhat independent of the manufacturing process.

In order to calculate the production cost for a component a process planner must draw on knowledge regarding all of the manufacturing processes used to create the part. In order to derive the lowest cost solution however a process planner must draw on knowledge regarding all of the manufacturing processes *which could* be used to create the part

2.6 Cost Modelling

The purpose of cost modelling a component is so designers know what effect their designs have on manufacturing cost, progress has been made on this for several decades. Most academic work and industrial cost models to date focus on producing/studying the relationship between either top level engineering parameters and developing scaling rules (parametric) or they focus on developing very detailed cost models (generative) based on manufacturing features to calculate machining time. However there is limited availability of multi-fidelity models which can fulfil both purposes. In most cases different models will be used in different stages of the design process with models using simple power-based scaling rules used early in the design process as proposed by Williams (1991).

The unit cost of a component or assembly comprises of a number of factors, including: the materials and consumables used, labour required and the capital employed.

An accurate and flexible cost modelling and optimisation tool, is a necessary pre-requisite to Value Driven Design (VDD) where the *value* of a component is assessed. VDD requires full knowledge of the costs and benefits thereof, including unit cost.

The progression of cost modelling technologies (and the author's view of the future, 6-7) is shown in the following list:

1. Open loop cost analysis
2. Expert estimation
3. Parametric - Simple rules, curve fitting methods
4. Activity Based Costing (ABC) - Static manufacturing route
5. ABC - With process optimisation
6. Integrated Cost/Performance models (VDD)
7. Full computational design derived directly from design requirements

Current literature relevant to this thesis focusses on item four and with focus shifting to item five over the last few years.

Thus in general costing methods are divided into two categories, generative and parametric. Generative cost models which model the actual production process usually using ABC (Activity Based Costing). Parametric models take very general parameters such as major dimensions, material type, mass or other low fidelity data and provide an initial estimate but rely on historical data. In general it was stated by Qian/Ben-Arieh (2008)that:

> "Parametric cost-estimation methods seek to evaluate the costs of a product from parameters characterizing the product but without describing it in detail. Parametric methods use the relationship between the physical characteristics of the part, such as mass or volume, and the cost, but with little or no physical relationship to the process. These methods are fast and accurate for well-defined part families but has minimal necessary result justification."

Research published by Qian/Ben-Arieh (2008) attempts to combine the two methods however all they really do is redefine the parameters to those which would be more typical of generative costing.

In contrast generative cost modelling seeks to logically model the process by which a component is actually produced and thus is expected to yield a greater accuracy on non-standard parts or where there is more sparse data, such as in the aerospace industry. However, generative cost modelling requires a more detailed level of design or some method of high fidelity dimension estimation during early design as some of the data required for this level of modelling will not be known during early design.

In addition to the equation based format typically used by cost modelling, ontologies have also been proposed by Zhang et al. (2014) as suitable for the storage of knowledge related to manufacturing processes. This allows additional complexity and thus more accurate calculation of process modelling outputs such as cost, time and resource consumption.

2.6.1 Parametric Cost Modelling in Early Design

A very simple parametric cost model for a component such as a compressor fan could be:

$$C_1 = C_2(S_1/S_2)^n \tag{2.1}$$

Where S is the fan area, C is the manufacturing cost and n is a number to be tuned. As can be seen there is no manufacturing knowledge used, simply experience from past fan designs and the assumption that the cost scales on some power of relative area. The subscript two denotes a comparable baseline component for which the cost is known.

If desired, learning rules can also be applied to take account of the volume of production. Parametric cost modelling can potentially be a very powerful design tool since according to Marion (2008). "Research suggests that 70% of ultimate product cost is determined during the early-stages of design." A more recent development in parametric cost modelling is described in Jiang et al. (2007) with the testing performed on warship manufacture. The results are good, however, it has been found that much of the cost of high tech fabricated products is due to the detailed geometry which is not known at the early design stages. Therefore a more detailed cost model is necessary. The other difficulty with parametric cost models is that they do not give a breakdown of costs so it is not known what the cost drivers are, simply the overall cost.

A variant of parametric modelling has been developed by Lee/Lee (2011), which instead of parameterising the geometry parameterises the functional requirements, thus enabling prediction of the optimal cost of a product from its functional requirements rather than design geometry. This approach of course assumes that the product is designed and manufactured in the optimal way but does provide an alternative to the construction and analysis of process routes. Their methodology would also indicate at an early stage whether a component with set functional requirements can be achieved at an acceptable cost and provide designers and manufacturing experts with a scientifically set cost target.

2.6.2 Parametric cost modelling in detailed design

It is true to say that parametric type cost modelling can also be used in detailed design. However, given that a parametric model is by definition based on historical data or expertise, it relies on the implicit assumption of no major and rapid changes in the method of manufacture or the design specification. If this is assumed to be fixed, then identification of the cost drivers and creation of parametric models such as by Zbiciak et al. (2015) can be used for design optimisation.

2.6.3 Generative Cost modelling

More complex and detailed activity based costing models such as in Qian/Ben-Arieh (2008); Chen et al. (2009); Zheng et al. (2010); Caputo et al. (2016) can be used later when more parameters are known. However, even this highly detailed cost modelling has its limitations. These limitations are uncertainty in the effective rate of machining (which may be affected by access constraints etc.), but also the real factory where the component will be manufactured and the machines on which it will be produced. It is thus necessary to estimate the cost rate and capability of the individual machines. Since a cost model which incorrectly reflects the actual cost rates will result in decisions being taken to optimise the model, which may increase overall costs. For Qian/Ben-Arieh (2008) the method of costing used was to break down a design and production activity into activities, identify a parameter which most affects the times spent on the activity and allocate a cost rate to the activity. The calculation of cost rates is discussed in another section. One point to note is that it is important to break down the activity into sufficiently detailed sub-activities such that the assumption of a single activity driver is valid. In addition to calculating activity cost rates and identifying activity drivers the final difficulty is in calculating the value of the activity driver. The method used to calculate the activity driver for machining operations is based on a simple formula shown below:

$$t_m = t_{\text{tool change}} + t_{\text{traverse}} + t_{\text{cut}} \tag{2.2}$$

Where the values for t are further decomposed to sums of lengths and feed rates. The authors continue to construct equations for cutting time which are based on geometric equations and different commands available to CNC software, number of cutters is represented by the number of different sized features such as holes. The number of set up operations is calculated from part dimensions. It should be noted that whilst the principles outlined in this paper apply to all components the detailed work applies mainly to rotational parts.

Manogharan et al. (2016) show the ability of generative models to explicitly consider the effect of batch size, highlighting that times calculation is necessary but not sufficient to gain a cost awareness. In addition to batch size, cost rates and material costs are of paramount importance.

It is possible for a cost model to become too detailed due to the fact that this equation is only true if the three features are completely independent, when in general they are not.

$$T_{1,2,3} = \sum_{n=1}^{3} T_n \tag{2.3}$$

This may be due to setup times or shorter tool paths being possible with different feature combinations. Some work has been done on feature libraries, such as by Hoque/Szecsi (2008), which have a selection of features the designer is permitted to use but these have the same limitations, or issues with too many different, but similar, features. The granularity of the definition of a "feature" is also an issue as stated by Scanlan et al. (2006):

> "It is not obvious what level of granularity is appropriate for a feature library. If the library is too coarse it limits design flexibility. If it is too fine it does not map well to manufacturing processes."

Scanlan et al. (2006) continue with the need to develop cost models with the input of designers to ensure that the cost models provide the necessary functionality. It is their opinion that the cost department for the company should focus on providing cost models for the designers rather than simply estimates. As stated in Section 2.1.1, Tammineni et al. (2009) believe that the system best able to provide cost information to designers is a system where a department is responsible for producing cost models using input from the knowledge of manufacturing experts. Designers can then use these models to make informed decisions, trading off functionality against cost. This direction of research begins to enter the area of intelligent rules based CAPP although the models generated appear to be variant based ones as described in Section 2.2.4.

2.6.4 Creation of Cost Models

In general when companies refer to cost modelling what they usually mean is time modelling, where the process times are then multiplied by some static cost rate. There has been some effort to use bi-directional flow of information where the results from some simulation will be reused in the cost model to have a dynamic cost rate (i.e. cost rates are dependent on the utilisation of capital but this is out of scope of this research).

Cost models are produced in different ways depending on the form of the model. If the model is parametric then the appropriate parameters and costs must be known. Since most use power laws with various adjustment factors there needs to be sufficient data to be able to make a reasonable curve fit. If the problem is over-defined then regression may be used. There are many different types of curve which can be constructed including but not limited to polynomials, power laws, exponential functions and Kriging. They can be tuned by various optimisers including genetic algorithms; neural networks; genetic programming; and simple, gradient based, approaches. The one thing all parametric models have in common however is that they do not need to involve physical understanding of the system and therefore they may or may not be accurate in large gaps or beyond the boundaries of the data, and they may have unexplained or unknown discontinuities.

Generative cost models are produced by knowing or deducing the process by which a component is manufactured, decomposing the process into activities and calculating the cost of each activity using a cost driver such as used by Qian/Ben-Arieh (2008). More specifically for manufacturing using the CostMesh tool developed by Rolls-Royce the cost is produced by a cost rate applied to each machine in a factory, the time to manufacture is obtained using volume to remove and a MMR (Material Removal Rate). Loading and unloading times are taken into account but the process rates are deduced from data in ERP software (SAP) which exists only for current components and their corresponding manufacturing methods. Furthermore since there is not really a *standard* method of producing/designing components this method is a good starting point but really only helps assess why cost is incurred by the time it is too late, when the component is already certified and in production. Its accuracy and validity is limited to a narrow envelope around the design parameters, and assuming no new processes have been developed since the last component which should be used in production.

One method for the creation of cost models developed by Tammineni et al. (2009) is a knowledge-based costing tool, this system attempts to calculate feature cost by taking two sets of geometric data, input and output. Cost models for processes and materials are stored in the libraries of a knowledge base. It was developed for turbine disks and allows the designer to specify distributions or use default values for parameters which are not yet certain because of the stage of design.

2.6.5 Cost rates for machinery

Calculating the cost rate for machinery is essentially an exercise in accountancy and is fundamentally simple. However, there are many different ways in which it can be applied, all resulting in different values. Essentially the different accounting methods are attempts to trade off the costs of the acquisition of information against the value of the knowledge which results from it. Traditional accounting uses overhead allocation where an individual machine or resource was allocated overheads according to some cost driver thought to be the most important, which could have been floor area. A more modern method, activity based costing is the most costly and accurate, each activity is individually allocated a cost rate according to the resources it uses. This still has some issue of overhead allocation however and the data is costly to gather unless an automated system has been put in place (in which case the system is usually costly). The data could also be very dynamic due to changing load factors and fixed costs such as salaries so some averaging is necessary to avoid overly complex data. Activity based costing focuses on the resources consumed during a particular operation. One of the issues with cost rates is that since depreciation is generally defined per year and is reasonably independent of the amount of use; if the machinery is used for half the length of time in a year the depreciation portion of the costs must be divided over half the number of components, thus increasing unit costs. Thus process costs are dynamic depending on the number of components passing through the factory which use that particular machine and the time they spend on it. Therefore it follows that for a company making more than one component if they change the manufacturing route to optimise a particular component for cost, they are likely to end up increasing the cost of another component, a situation made worse by fact that they may have to purchase new equipment for the changed manufacturing route. These intricacies of accountancy mean that a holistic view must be taken if the firm is to benefit overall from the optimisation. An optimisation using a too specific view is likely to result in a suboptimal result. ABC allows the cost per operation to be calculated from:

1. Cost of machinery
   (a) Investment Costs
   (b) Maintenance
2. Cost of Operator
   (a) Operator Time
   (b) Skill Level
   (c) Country

3. Process Time
4. Supplier Margin

2.6.6 Using CAM Data for Cost Model Improvement

One of the major issues in current cost modelling and process planning methodologies, is the acquisition of data required to ensure their accuracy. Much of this data should be easy to acquire, at least for currently used manufacturing processes, because CNC code exists on machines. These data are not in a format helpful for cost modelling purposes, however this section explores the changes that are expected to take place, to increase the usability of these data.

According to Xu/He (2004) there are limitations in the code used to transfer data from CAM systems to CNC controlled machinery, these are shown below:

1. The language focuses on programming the path of the cutter centre location with respect to the machine axes, rather than the machining tasks with respect to the part.
2. The standard defines the syntax of program statements, but in most cases leaves the semantics ambiguous.
3. Vendors usually supplement the language with extensions that are not covered in the limited scope of ISO 6983, hence the CNC programs are not exchangeable.
4. It only supports one-way information flow from design to manufacturing. Thus the changes in the factory floor cannot be directly fed back to the designer. Hence, invaluable experiences on the shop-floor are not easily preserved.
5. There is limited control of program execution and it is difficult to change the program in the workshop.
6. The geometry data is not used directly on the machine, instead is has to be processed by a machine-specific post-processor, which creates low-level, incomplete data that makes verification and simulation difficult.
7. ISO 6983 does not support the spline data, which makes it incapable of controlling five or more axis milling.

These limitations can be clearly seen in the way that vendors often need to add unconventional commands to their machinery, to improve performance, because the existing commands are insufficiently well defined in modern manufacturing to accommodate the required functionality without user definition. It has also been calculated by Perez et al. (2008) that a varying feed rate can increase the material removal rate and thus reduce machining time. Typical G-Code, however, would specify a fixed feed rate for a substantial part of the operation which would be limited by cutting force during the most intensive part of the operation, thus most of the operation would be conducted under sub-optimal conditions. Other issues with G-Code according to Tikhon et al. (2004) is that any machining operations which follow parametric lines which are not straight lines or circular require very long CNC codes as many points need to be specified, in order to use the linear of circular interpolation functions. This can cause problems because:

1. The transmission error between CAD/CAM and CNC systems for a wide range of data may easily occur, i.e. lost data and noise perturbation
2. The discontinuity of segmentation deteriorates surface accuracy
3. The motion speed becomes unsmooth because of the linearization of the curve in each segment, especially in acceleration and deceleration

In addition the back transfer of data on its own would be helpful as it has been seen that it is occasionally necessary to adjust variables such as feed rates in response to factors such as vibration. In this case the back transfer of data from operator is manual whereas in principle there is no reason why the following could not occur:

1. The operator could reduce the feed rate on the machine
2. The machine could query whether this is a permanent change
3. If so, the CNC code could automatically be adapted
4. The process planning software could automatically adjust to the new time and potentially begin to learn in what situations this speed reduction would occur to enable future models to be more accurate.

There are many manufacturers across the world who are part of this (STEP-NC) project, particularly in the US however given the legacy of machinery left from previous manufacturing and the certification requirements, the possibility of using more sophisticated and capable CAM would be difficult in existing processes.

It should be possible to incorporate the technology in novel processes or within the supply chain. Siemens and Volvo, Rolls-Royce machinery supplier and supply chain partner respectively are both part of the STEP-NC project. Such a project could be beneficial in process planning because the software may be duplicated or used (to calculate operation time) for input into dynamic process routing software. STEP-NC has been used in research by Rauch/Hascoet (2012); Teich et al. (2010) as a method to identify the limitations of manufacturing processes in order to optimise product design.

The research should take into account potential improvements in CAM technology as this may provide additional information which could be incorporated into process planning software and may allow improvements in future.

2.6.7 Summary of cost modelling

In summary, we can see that of the two main methods of cost modelling: parametric or generative, only generative is suitable for the purpose of direct understanding of the manufacturing process and the flexible manufacturing route that this research is intended to consider.

Given then that we must use generative cost modelling the question then becomes how best to implement this with maximum usefulness and minimum maintenance. The solution to this trade off is in two parts but both involve high levels of automation and computer coding ability.

Cost is broadly a function of time and cost rate, where times can be generated by algorithms that can be created by models produced by experts, and can be improved by data mining factory data, as in Section 2.6.6. The initial algorithms could be produced from literature such as in Section 2.5.

Cost rates (Section 2.6.5) can be data mined and automatically kept up to date (at least in principle) through data mining the company's enterprise resource planning system, through economic intelligence subscriptions or through direct numerical replication of the accounting process.

Where custom tooling or fixturing is used (Section 2.5.4) this amortised cost would need to be incorporated into the operation cost, so here again flexible models would be needed to predict these costs.

For the purposes of this research, it is assumed that these three tasks have been completed and that the database of cost rates and manufacturing knowledge is (or could be) populated and maintained up to date using these or other methods.

2.7 Summary of Literature Review

From the literature review areas such as feature recognition and translation have had much work conducted on them but a comprehensive solution has not been found.

Areas that are not well addressed include:

1. Process planning of assemblies.
2. Integration of supply chain simulation.
3. How the selected forming process influences the manufacturing process choice and sequence

There is also a deficiency of literature related to the combination of choosing the manufacturing processes necessary, sequencing them and optimising for cost in the same optimisation. Most research tends to focus on the choice of manufacturing process or the sequence, but not both.

In the literature there is a limitation of the complexity of parts which can be processed, particularly within rapidly executing applications, due to the processing time required for such applications. This is thought be essential for industrial application; it is only with an application which can run in minutes in low fidelity mode that designers will consider the use of such a tool and the objective to improve designer awareness of the implications of their decisions will be met.

From the possible design methods shown in Section 2.1.2 the most advantageous approach would be (3b), as this maximises the benefit of the human-machine interface, however there are several levels of automation associated with this and insufficient time to fully explore this area.

The approach to be taken will concentrate on (a) - the designer specifies a feature, geometry, tolerances and surface finish and the computer will choose a process route. This approach could also be expanded to (c) in the future.

It is thought that this methodology could be run concurrently with other tools such as Computational Fluid Dynamics, Finite Element Analysis and Mean Time To Failure estimation codes to obtain a value for the design which could be used for cost/performance trade off studies, as suggested by Bao/Samareh (2000).

3 Commercial Costing Systems

This section details and surmises some of the commercial tools available to undertake cost prediction and optimisation from CAD packages and the system internal to the author's industrial sponsor, addressing their strengths and shortcomings.

3.1 Internal Methodologies of Industrial Sponsor

Cost models are produced using Vanguard Studio by Rolls-Royce Product Cost Engineering. They are typically manually produced, although the use of in house developed code components (Vanguard models embedded within the CFT) has enabled acceleration of the modelling process. The CFT cost models typically rely on the implicit assumptions that:

1. Rolls-Royce make the same type of component on a repetitive basis for each engine program in the context of geometrical design features.
2. Rolls-Royce makes all components of a specific type using the same manufacturing processes and sequences.

These are often reasonable assumptions but this does tend to inhibit the use of novel manufacturing processes until there is an overwhelming advantage to doing so.

These models are highly detailed generative cost models, built manually for each component family under investigation. They began as models built automatically from manufacturing route data (Section 1.7.5) and then modified with simple scaling rules; but have since evolved into far more detailed generative models containing a cut-by-cut approach and make use of reusable child component cost models where generic data sets lend themselves to being applied.

This far more detailed (and accurate) approach means, however, that they are more difficult to modify for a change in manufacturing strategy so generally more than one model will be constructed where multiple manufacturing strategies are to be considered.

Process planning at Rolls-Royce is far more conservative than published technology, requiring far more human calculation and analysis.

3.2 aPriori

aPriori is a software tool designed to provide unit cost based on solid model CAD geometry. It uses a feature recognition algorithm to identify manufacturing features followed by an extensive logic based rule set to compute how to manufacture each feature, but it does not consider process sequencing. It considers setup by grouping operations by tool access axis but does not consider process precedences, and apart from this grouping there is no relation between operation sequence and costs in aPriori. In addition all chemical, thermal or surface treatment processes must be mandated for all features or manually added for the required features.

3.2.1 Selection of manufacturing strategy

Manufacturing strategy (termed process group by aPriori) selection is manual, and aPriori will not necessarily inform the user if the strategy chosen is impractical/impossible, for the component chosen. aPriori will provide a cost with warnings about the GCDs (Geometric Cost Drivers) which cannot be met; but this is not infallible. It is shown below that even when all GCDs have been met, the component can not necessarily be manufactured using a selected manufacturing strategy.

The component in Figure 3.1 was tested by specifying sheet metal forming (SMF) and was successfully modelled with no errors. In addition, sheet metal was calculated as the least expensive manufacturing strategy for this component. If an inexperienced designer simply tried to cost model for SMF, they may then blindly believe the result, emphasising that the users of aPriori would need basic manufacturing knowledge. The problem caused by this defect would be magnified if a design optimiser were using aPriori to modify designs, and testing for all manufacturing strategies each time. It would result in the design being optimised for an impossible manufacturing route.

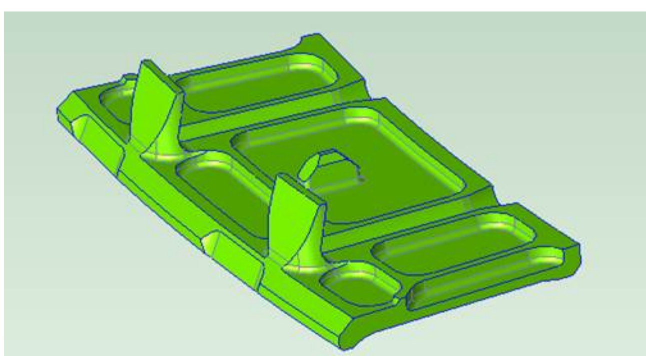


Figure 3.1: Example component cost modelled by aPriori for sheet metal forming

3.2.2 The aPriori Virtual Production Environment (VPE)

The VPE contains all the information necessary for aPriori to choose which processes and machines to use.

CSL (cost scripting language) is aPriori's integrated programming language within the

VPE. It can produce scripts and functions, it is highly flexible and has an integrated JIT (just in time) compiler. In addition CSL does not follow the usual programming rules governing code as variables need not be defined before they are used. However, whilst undefined variables show a referenced error circular logic results in an error with no useful locating information. Tooling costs for forgings are also calculated using the VPE code but this part of the language was unexplored in the three day workshop tested. Finally material properties such as density, hardness and cost rate are stored here.

3.2.3 Selection of manufacturing processes

The processes in aPriori have limits on the tolerance they can achieve and variability data is used to establish the process route but aPriori does not take into account process variability as a cause of scrap or rework.

Where the logic set may result in more than one FMP, aPriori will select the lowest cost using localised exhaustive search of possibilities. However, it will not search all permutations, instead adding additional processes if they are necessary to meet the designated tolerance.

Manufacturing processes are unique to the process group in which they are located, and only one primary process group can be used for a process routing. Secondary process groups consist of machining, heat treatment, surface coatings and other groups, as many secondary process groups as necessary can be used.

aPriori does consider TAD (tool access direction) which results in several instances of the same type of operation as they are produced from different directions.

Tolerances are considered but can only be read directly from CAD geometry using Pro-Engineer at the moment if they are correctly associated in the CAD file.

3.2.4 Selection of machines

Machines can have feasibility rules, such as sizes and are pre-set to only apply to one manufacturing process in one process group, thus machines which can physically conduct more than manufacturing process – including turning roughing, turning finishing, milling and drilling must have multiple instances of the same machine. It can therefore be seen that multiple virtual machines are necessary for one physical machine which can conduct multiple manufacturing operations.

If more than one machine is feasible for an operation aPriori will choose according to CSL logical rules, usually utilising a sort command. If the rules allow more than one choice then aPriori will default to the most cost effective.

Machines have setup times, labour rates, max dimensions etc. They are associated with specific feature sets and the machine data is both highly detailed and expandable with additional fields. No programming language experience is necessary to modify these fields although adding more fields requires use of the in-built programming language.

3.2.5 Cost Modelling

CSL (Cost Scripting Language) is the basis for both the decision on which processes and machines to use, and the calculation of cycle times and costs. The programming can be adjusted by the user as desired and is compiled at run-time, thus eliminating the problem of slow interpreted code against inflexible compiled code. CSL cannot use or create array variables but this is only an occasional annoyance, and is unlikely to be an insurmountable issue.

Multiple cost categories are available, and each category of costs for each feature is calculated by code compiled at run-time. Although aPriori does not consider route sequencing explicitly there is an implicit assumption that similar operations are conducted together. This was identified by noticing that the incremental cost of similar operations is significantly lower than the first operation of a specific type. A key category of costs which is omitted is transport costs, although they could be incorporated within the setup costs.

3.2.6 Assemblies

For assemblies aPriori can calculate the costs of assembly, but the costs of the individual components must be pre-calculated by aPriori, or defined. Assembly; welding, preparation and clean up; and mechanical joins such as riveting can be considered, as can further heat or surface treatments. However, aPriori does not automatically have any assembly automation. Importing an assembly will automatically result in a short operation to take the parts and put them on a workbench. Operations such as welding, bolting, or even snap fitting must be added manually as the feature recognition algorithm cannot identify joins and therefore cannot propose methods of fixing them. In the author's opinion this is a major and unnecessary oversight.

3.2.7 Summary of aPriori

The feature recognition capability of aPriori is powerful and it is certainly a substantial improvement on both other commercial tools and the current approach taken by Rolls-Royce internally. However, there are major points for improvement, some of which are addressed by this research:

- There is no uncertainty analysis for design changes; so there is limited and unclear guidance on what to adjust to reduce cost.
- Changing material utilisation does not affect the machining cost, hence it is clearly not affecting material removal volumes. In addition aPriori will not select the stock shape for the user, instead, rectangular or cylindrical stock must be specified. Allowances for stand-off distance is shown as an option but does not affect the cost, so it is unclear what it is affecting (if anything).
- Output is a set of costs and if necessary a list of the manufacturing methods used but not the order. This reduces its usability to manufacturing experts because of their focus on fixturing and the effect on process rates of part rigidity.
- There are likely to be issues with retrieving data from aPriori if a decision to cease using it were to be taken. The data including costing equations and logic would be difficult to extract and use, raising the exit cost and potentially resulting in a loss of bargaining power with the software vendor. The software vendor has assured that there is an export facility but this is likely to be just a text dump which will require further work to be useful again alternative models.
- For assembly type operations the joining process must be manually selected, even the fact that there needs to be a joining process must be manually identified.
- In some cases features were found and then quietly ignored as aPriori could not model their costs.

Some of the aforementioned shortcomings within aPriori could be overcome with a thorough understanding of CSL and modification thereof. Some, however, are intrinsic to the modus operandi of aPriori and thus more difficult to overcome. It will be demonstrated later in this thesis that the optimum method of manufacture for a feature depends in part on factors pertaining to other features. aPriori has an deterministic tendency to make repeated suboptimal decisions, as it does not take a sufficiently holistic view of the design search space.

3.3 Perfect Costing (Siemens)

Perfect Costing, available from Siemens has advantages in that it interfaces with their CAD package (NX) which is the Rolls-Royce corporate geometry modelling tool. However, following a discussion with a Siemens' representative it was determined that it was in fact a minor improvement on RR CAPP with some tracking and automated recalculation of process times and thus at best no more sophisticated than the aforementioned CFTs (Section 3.1).

Perfect Costing therefore is a simple activity based costing tool with no optimisation, it is limited by the general assumption that the manufacturing route is determined prior to cost modelling and is not affected by it.

The representative from Siemens was unaware of any substantial improvement to Perfect Costing in the near future.

3.4 CATIA

As of 2014 CATIA has had a cost tool integrated within the CAD package (Hiller, 2014), designed for composite components, it allows calculation of the cost of manufacture. It must be provided with detailed information on the composite structure, such as ply patterns etc.

3.5 Solidworks costing

Following testing (Figure 3.2) on a very simple hydraulic cylinder and piston it is clear that the Solidworks costing tool is fully automated, but fundamentally limited. Solidworks assumed a component machined from a solid rectangular billet and ignored some of the features (Figure 3.3). In addition there was no consideration of tolerances and thus no account taken of process variability.

Feature recognition is included to a limited extent, holes are recognised and the program drew a cuboid around the shape as a representation of the billet to be machined, this was used to calculate material cost. Unfortunately basic operations such as turning are not used for axisymmetric features and milling appears to not work (the region around the cylinder is un-costed). There is no account taken of tolerances, inspection, failure or rework, order of operations, or the capability of machines. It does run in seconds owing to its simplicity, however it appears to be at best misrepresentative and a placeholder for further development.

There are no improvements which the author is aware of planned by Solidworks.

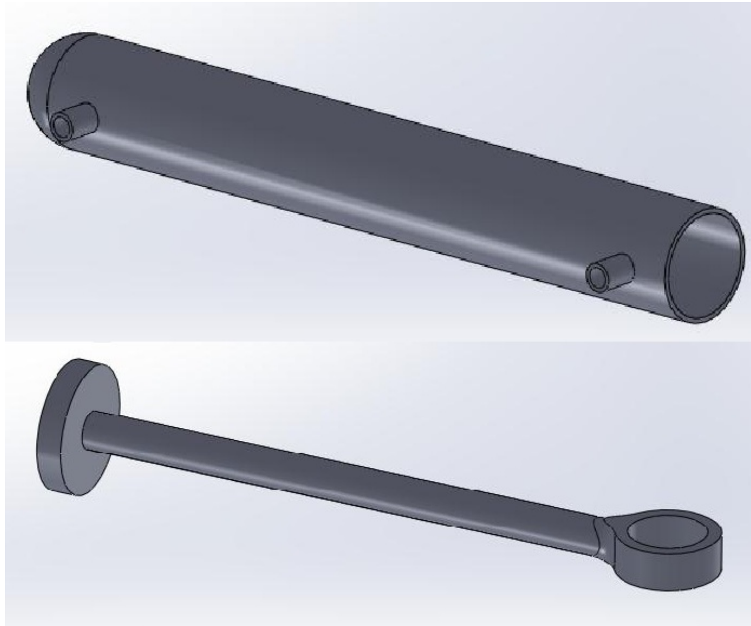


Figure 3.2: Solidworks costing functionality: Piston and cylinder render

3.6 Summary of Commercial Systems

The current commercial systems are lacking in fundamental ways, and all of them operate under the assumption that a rapid response is required. They therefore do not optimise the manufacturing route to minimise costs, instead this type of full global optimisation is limited only to academic work at the current time.

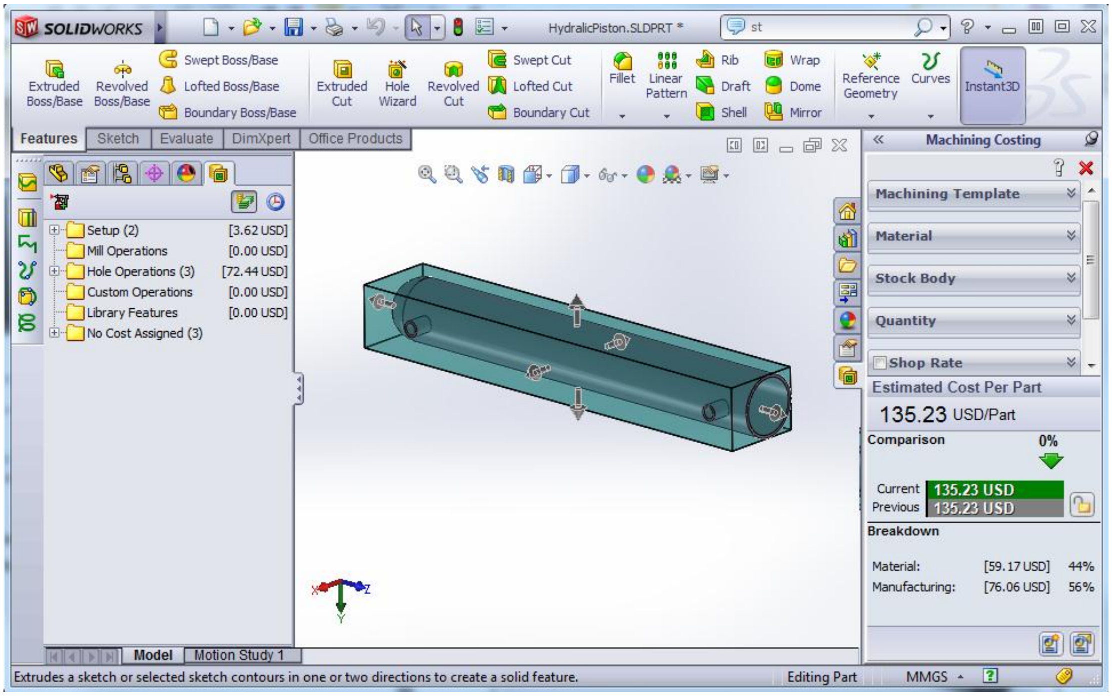
In addition:

Some company's internal methodologies and Siemens' perfect costing suffer from the same lack of automatic flexibility; based on the assumption that the best way of making a component is the way a similar one was made last time.

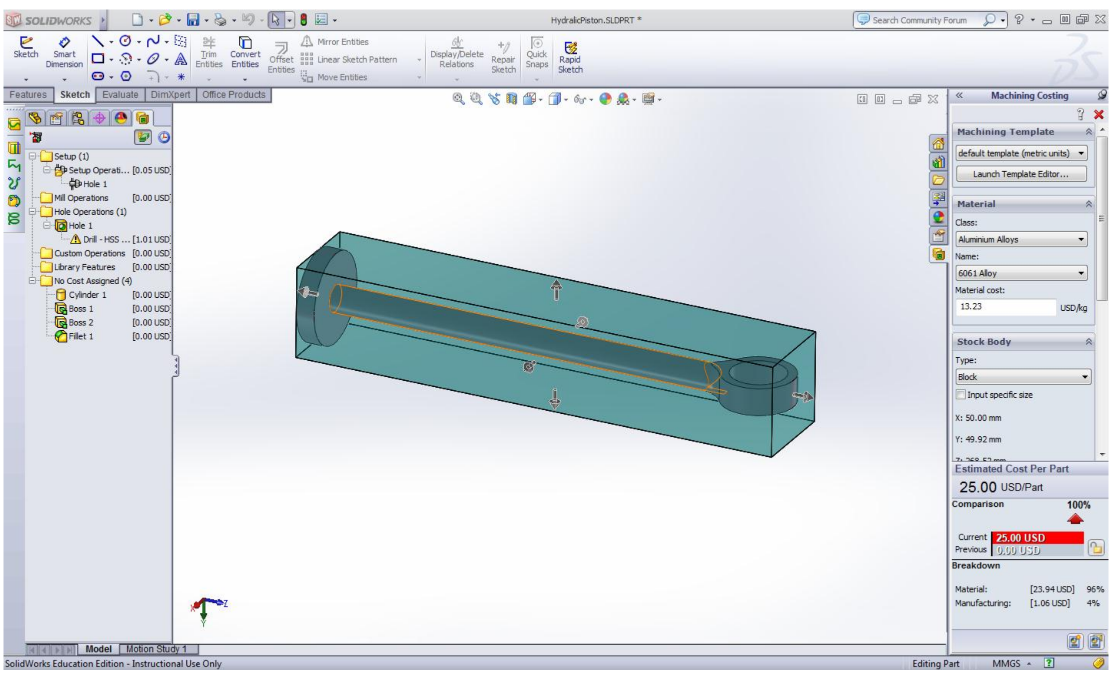
Solidworks' feature recognition causes problems, even on unrealistically simple parts. In addition it does not consider process variability when choosing manufacturing processes.

The most promising of the commercially available tools is aPriori. Its feature recognition, knowledge of some of the effects of tolerance, logic based rule set, the expandable knowledge repository and the flexibility of its in-built programming language are impressive. However its lack of consideration of process sequencing, scrap rates and rework calculations show that much more can be done commercially before the most recent academic research has been fully exploited commercially.

All of the tools and the author's experience indicate that a rapid response is required from a cost system to be considered by industry. Therefore this research will allow the result of a full manufacturing optimisation to be stored and reused. Two options will exist for the



(a) Cylinder cost modelling



(b) Piston cost modelling

Figure 3.3: Solidworks costing functionality

user:

1. A full optimisation will be used to find the optimal method of manufacture when a rapid response is not required such as overnight or over a weekend, with the optimum manufacturing route being stored.
2. A real time response option which reuses the stored manufacturing route in (1) and recalculates costs for new geometry or changes in tolerance.

Approach (2) would compete with commercial systems to give a real time response but the dependency on (1) would require regular reruns of (1) thus keeping the baseline up to date with design changes.

This research will not have feature recognition functionality which is desirable from a commercial and usability perspective. It is therefore thought that future work could be to integrate this with feature recognition which would create a product superior in all ways to current commercial tools.

4 FoSCo (Forming Shape Creator)

The overall system presented in this research attempted to deduce the optimal method of manufacture, and the cost thereof, given the finished part geometry. This methodology, to be executed in a computer or similar device, is submitted for patent in the USA and the EU.

EU patent number 16180366.3 - 1954

US patent submission reference 5895 LMT

Contained within Section 8 is the full patent application for FoSCo.

This research, in accordance with industry practice, assumed a constant specific cost of the selected material, and calculated the shape which can be created during the primary forming process or manufacturing strategy.

The purpose of FoSCo is to allow the design feature file to contain the exact amount of material ‘removed’ during the forming process, thus making possible the accuracy of the Manufacturing Choice Optimisation and Sequencing Tool (MCOST). This virtual removal saves on material cost, although it may increase the cost of the mould or process, due to increased complexity or increased forging time. Awareness of the extent of material removal required after the forming process allows the calculation of times, and thus costs, of later material removal processes.

Thus the objective of FoSCo is to identify the shape which would be made by a forming process, from this the depth of cut required to machine away the excess material (if any), can be calculated. The distances, between the Final Machined Part (FMP) geometry and the formed shape, are then used by MCOST which requires material removal volumes to calculate machining time.

As discussed in Section 2.4, there has been research to determine the geometry which will be created by forging operations. However, this section will present a new method, a method which does not require an extensive rule set to operate. Two approaches were used in this research: face or edge offset for near-net-shape manufacturing strategies, and the methodology to be described where the manufacturing strategy creates smooth shapes around the FMP.

The approach was intended to be modular thus enabling different codes or modules to be used for different component types, or forming methods, and is thus expandable. Because of the nature of the co-ordinate system used by the default FoSCo convergence module, the smoothing algorithm is unsuitable for some types of shape, and for some types of forming process (particularly near net shape methods and shapes with multiple concave surfaces). Methods to resolve this are to be presented in future work. It should be noted however, that whilst this method may not calculate the exact forming shape in all cases, it has not yet failed to find an estimate of the forming shape, which more complex rules based methods can. It will thus give an indication of whether a design change will result in a positive or negative change in cost, and an approximate magnitude even if the absolute value is inaccurate[3].

The two 2d test cases, and their predicted forming shapes, are shown in Section 4.4.12. For the blisk test case there is also:

- The industrial sponsor's manual estimate of the forging shape.
- The estimate of the forging shape from the industrial sponsor's proprietary forging shape creator tool (ACE).
- The finished part axis-symmetric profile.

The 3d version of FoSCo used for test case two (the engine section stator) is intended as an illustration of the flexibility of the methodology presented (Section 4.5.5).

4.1 Precursor Technologies

It is assumed that the manufacture of the test cases (Sections 1.8), involves creating a shape which is everywhere larger than or equal to the finished part geometry (FPG). This shape is referred to as the precursor or forming shape, and is created by a primary forming process. It is apparent that for manufacturing it may sometimes occur, that components are made by joining additional material, usually by welding. However, when this occurs it is proposed that the additional sections are analysed separately, and analysis is conducted for the assembly process. The cost of the assembly process is out of the scope of this thesis, and is outlined for incorporation in future work (Section 6.5.3).

Given that in some primary forming processes additional material will be present (in addition to the FPG). The extraneous material is then removed until the FPG is achieved. For a net shape manufacturing strategy, this material removal will be zero; for near net shape, minimal; and for legacy manufacturing strategies, potentially large such as 12:1 (Blose et al., 2006). Additive technologies could be considered net shape primary shaping processes, and therefore be modelled by this methodology.

However additive technologies have been defined as out of scope for FoSCo, this is because:

1. The additive technology may impose additional requirements intrinsic to the process such as: printed support structures, or powder removal passages. A dedicated tool is thought superior for these technologies.

[3] George Box (1978) - All models are wrong but some are useful

2. The use of additive technologies often enables substantially different geometric design to maximise the benefits. In other words, if a component is to be produced through additive manufacture, it is likely to be too dissimilar to allow direct comparison with traditional manufacturing.

For test cases zero, one and two, at least two different primary forming processes will be considered.

The methodology behind the generation of these shapes is essentially a smoothing algorithm, derived from an iterative implementation of image recognition algorithms Kass et al. (1988). It was modified to a polar co-ordinate system for a closed contour and to give a stand-off distance.

It essentially has three stages:

1. Export of FMP (Finished Machined Part) data from NX, either as a two dimensional set of lines or a three dimensional surface mesh. It does not at present operate with curves or splines but this would be a simple and obvious improvement.
2. Calculation of a closed loop of points (2d), or a point surface (3d), which represents the forming shape.
3. Import of this ring/surface into NX and calculation of the distances between the FMP and the forming shape.

These tasks could all be completed within the NX API but for ease of debugging they were kept outside for the prototype software.

4.2 Nomenclature

Acronyms

Acronym	Meaning
FMP	Final Machined Part
FMPP	FMP Point: A point on the surface of the final machined part
FSP	Forming Shape Point: A point on the surface of the forming/precursor shape
MCOST	Manufacturing Choice Optimisation and Sequencing Tool
POI	The Point Of Interest, defined as the FSP currently being examined

Variables

Symbol	Meaning
B	The vector from the closest FMPP to the POI.
$\hat{B}$	The unit vector of B
C_P	The location of the convergence point (specific to one or more FSPs)
D	The vector between the FSP and the closest FMPP
f_d	Damping factor
L	The Cartesian co-ordinates of the POI
m	Magnitude of movement of the POI
M	The total movement applied to the FSP before time step control and damping
M_P	The closest final shape point (FMPP).
M_1	A constant movement vector towards C_P
M_2	A constant movement vector towards the closest FMPP
M_3	An inverse square movement vector away from the closest FMPP
n	The vector from the POI to the Convergence Point to which it is assigned
$\hat{n}$	The unit vector of n
N_P	The Cartesian co-ordinates of the closest FMPP
r	The radius of the POI
x	Minimum euclidean distance between adjacent FSPs

Co-efficients (numerical value specific to each forming technology)

Co-efficient	Meaning (Specific to each manufacturing technology)
k_1	The co-efficient which controls the magnitude of M_1
k_2	The co-efficient which controls the magnitude of M_2
k_3	The co-efficient which controls the magnitude of M_3
k_4	The co-efficient which controls the magnitude of the curvature smoothing correction

4.3 FoSCo Architecture

The architecture for FoSCo is shown in Figure 4.1. It can be seen that the basic architecture of both the 2d and 3d versions is the same, the only differences between them are in the method of import of the FMP geometry, and the simplified equations for the 2d version (due to the z co-ordinate being always zero, and only two adjacent points for smoothing).

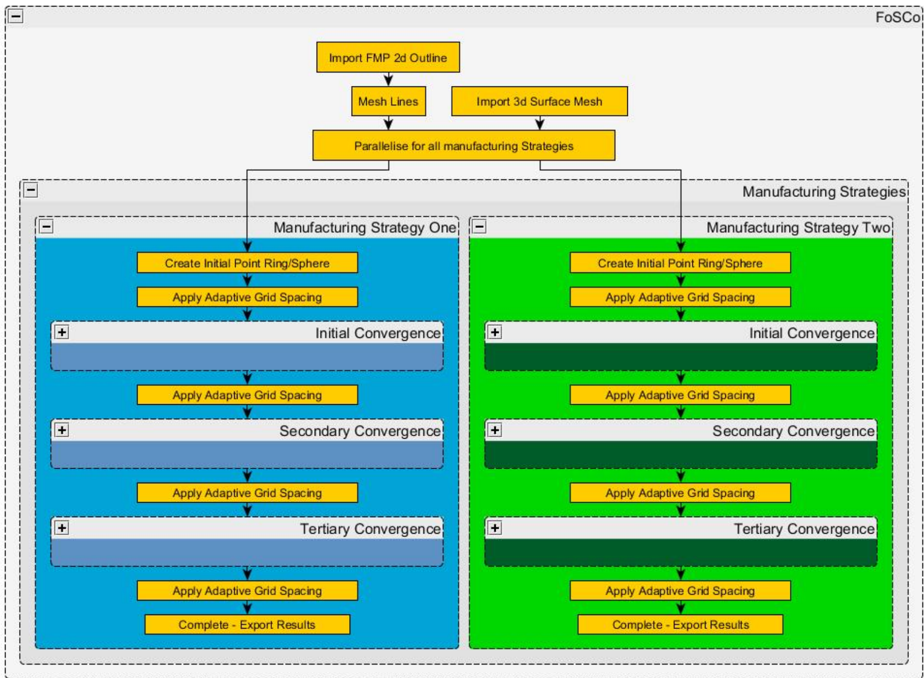


Figure 4.1: FoSCo architecture

For both the 2d and 3d versions of FoSCo the secondary convergence (Section 4.4.8) uses an energy balance type equation, with a movement vector force pulling the Forming Shape Point (FSP) towards the nearest Final Machined Part Point (FMPP) and an inverse squared movement vector pushing it away. A centre convergence movement vector is also used to move points towards the centre when their closest point is perpendicular to their unit vector towards the origin. An excessive convergence filter within secondary convergence is also used, and described later.

Tertiary convergence is a smoothness correction necessary for the shape generation, described for FoSCo 2d in Section 4.4.9. This algorithm was not implemented in FoSCo 3d and this is left as future work.

An algorithm for primary convergence is also described, it has no effect on the final result but it reduces the number of iterations required in secondary convergence, therefore reducing the time taken to evaluate the forming shape. It was implemented for both 2d and 3d versions of FoSCo.

Both versions also use adaptive grid spacing, to provide adequate definition of the shape generated without excessive computation.

Clearly for the two dimensional version the z co-ordinate is always zero, simplifying the calculations, in addition there are only two adjacent points in the 2d version, simplifying the smoothness correction.

4.4 FoSCo methodology (2d)

Creation of the forming shape is a multiple step process involving primary convergence, secondary convergence, and a smoothing algorithm. The primary convergence is a low overhead initial phase, to reduce the solution time. The secondary phase is a balance between convergence forces and inverse squared repulsion from the closest FMPP (Final Machined Part Point). The smoothing algorithm will increase the radius of points which are too far out of alignment with their neighbours. In addition to the above algorithms, there is adaptive grid spacing to ensure an even distribution of points. The stored data on each point are: its angle (theta); its radius (r); and whether it is deemed to have stopped. Theta is fixed for each point, but the radius varies.

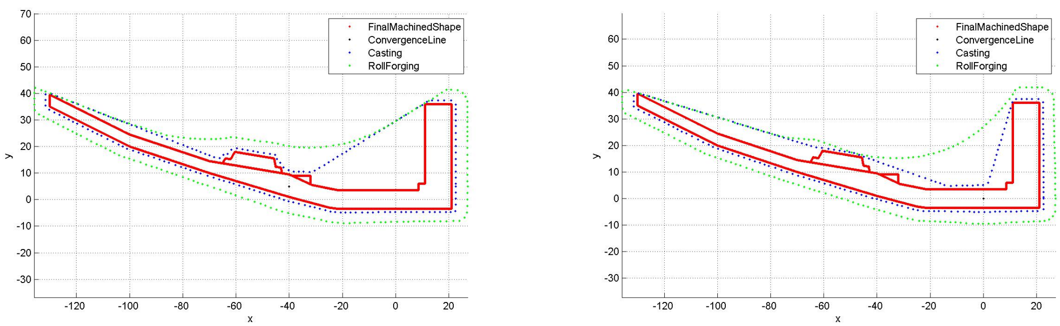
The step by step explanation of the methodology begins in Section 4.4.5, with the methodology importing a set of lines which form a closed contour from CAD geometry. Prior to this some supporting functions should be explained, namely the co-ordinate system, convergence points, adaptive grid spacing, and finding the appropriate FMPP from the set.

4.4.1 Co-ordinate Systems

One of the hardest questions to answer when attempting to generate forming shapes is how to define a co-ordinate system, in particular the consideration of neighbours. A Cartesian system of points can specify any point in n dimensions, but has no implicit concept of neighbours, a radial/spherical co-ordinate system on the other hand can use the angle(s) to define neighbours. It can easily be seen that neighbours are essential to predicting the shape to be generated by a forming process, as to suggest that a point on a relatively smooth surface can be unaffected by the position of adjacent points is axiomatically false.

So, given that a closed contour or surface must clearly be formed, either by designating a set of points, or an equation which can be solved. Given also that the placement of a particular point on the surface (representing the surface of a formed shape) is dependant on several factors including the position of adjacent points, some method of defining adjacent points and preferably keeping the same neighbours throughout the algorithm is necessary. The method presented here is the use of a combination of the Cartesian and polar co-ordinate systems, where adjacency can be defined using the theta value in polar co-ordinates, but placement calculations can be conducted after a polar to Cartesian co-ordinate transformation. One of the issues with the use of a polar co-ordinate system is that of shading, certain regions of space become shaded (Figure 4.2). On the first graph, Figure 4.2a, some regions are inaccessible by a shrinking circle without first passing through the finished part geometry, and the methodology fundamentally prohibits this. For simple shapes this issue [shading] can be alleviated with careful placement of the shape relative to the origin (Figure 4.2b) of the polar co-ordinate system.

For more complex shapes it can become impossible to locate the origin such that shading is not an issue somewhere. Hence the creation of the convergence line, a concept whereby adjacent forming shape points may use a similar co-ordinate system, but with a different origin in Cartesian space. Or to put it another way, there are multiple polar co-ordinate systems contained within the Cartesian space. This concept means that areas considered shaded and thus inaccessible for one FSP and its co-ordinate system are accessible to others. An algorithmic method of determining the convergence point is now required, described in Section 4.4.2.



(a) Origin in original position, casting shape clearly unrepresentative

(b) Origin moved to reduce shading of casting process

Figure 4.2: The issue of shading when a point is used for convergence (Test case two geometry)

4.4.2 Finding the convergence point

One of the important aspects of this algorithm is that the FSPs do not all converge to the same point (that is if the radius of all the FSPs were zero, a line would be formed). It is this aspect of the algorithm which allows the shape created to be close to the FMP shape in a near net shape manufacturing process. The convergence line is at present manually defined and should be enveloped by the FMP shape, the end points of the line should not be too close to each other relative to the origin (0,0) of the native co-ordinate system. This is because when the convergence line is constructed the end points are assigned a range of angles, based on the line from the origin to the end point $\pm$ 15º, as shown in Figure 4.5, within which all FSPs are assigned to them [the end points]. Ergo if the start and end points are too close to each other in theta value relative to the native co-ordinate system, these 15 degree triangles could overlap. The convergence line defined has start and end points with angles from the positive x-axis in the anti-clockwise direction of λ and ϕ respectively.

For each FSP (Forming Shape Point), a convergence point is defined. Which convergence point is used is dependant only on the θ value assigned to that point (which does not vary).

It can thus be seen that from the moment an FSP is defined, its θ value is defined and constant, and thus its convergence point is also defined and constant. FSPs which do not fall in the range of $\lambda \pm 15^{\text{o}}$ or $\phi \pm 15^{\text{o}}$ (the range of θ angles applicable to the start or end points respectively) are assigned one of the intermediate convergence points in accordance with Algorithm 1.

Algorithm 1 is used, and the mapping for an example convergence line (Figure 4.3) can be seen in Figure 4.4a Figure 4.4b. Essentially whilst $\theta = \lambda \pm 15^{\text{o}}$ we are held at index zero, as we traverse the angular space between the upper bound of $\lambda \pm 15$ and the lower bound of $\phi \pm 15$ we move up through the indices of the convergence line, whilst $\theta = \phi \pm 15^{\text{o}}$ we are held at the maximum index of the convergence line, then as we continue on around increasing θ we return back down the indices.

This angle (15^{o}) is detemined by the fact that some FSPs should be present within its arc, thus the angle is determined by the interaction of the stand off distance of the mnufacturing process, the distance from the end of the convergence line to the FMP shape, and the spacing of the FSPs. It is left to future work to derive this angle automatically, but tests did not show a significant deterioration of the algorithm performance when the angle was changed to 10^{o} or 20^{o}.

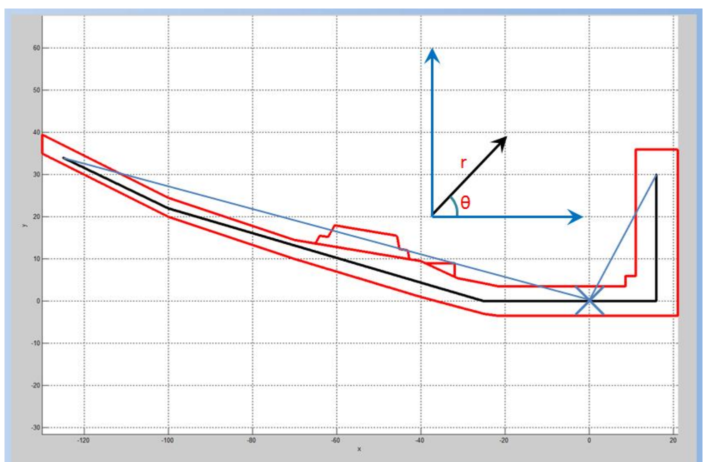
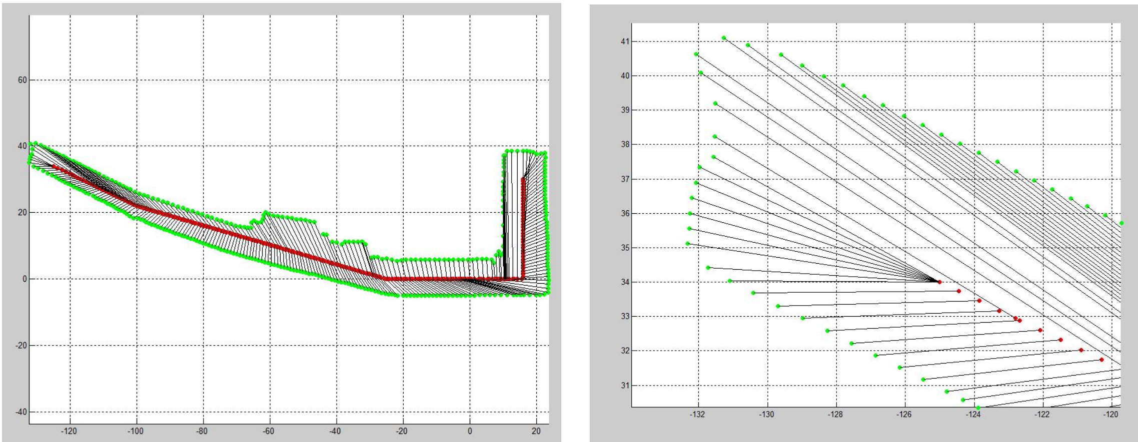


Figure 4.3: The convergence line with start and end points shown



(a) The convergence line with an intermediate set of FSPs (lines connect the FSP with its convergence point

(b) A magnified graph of Figure 4.4a

Figure 4.4: FSPs linked to their convergence points

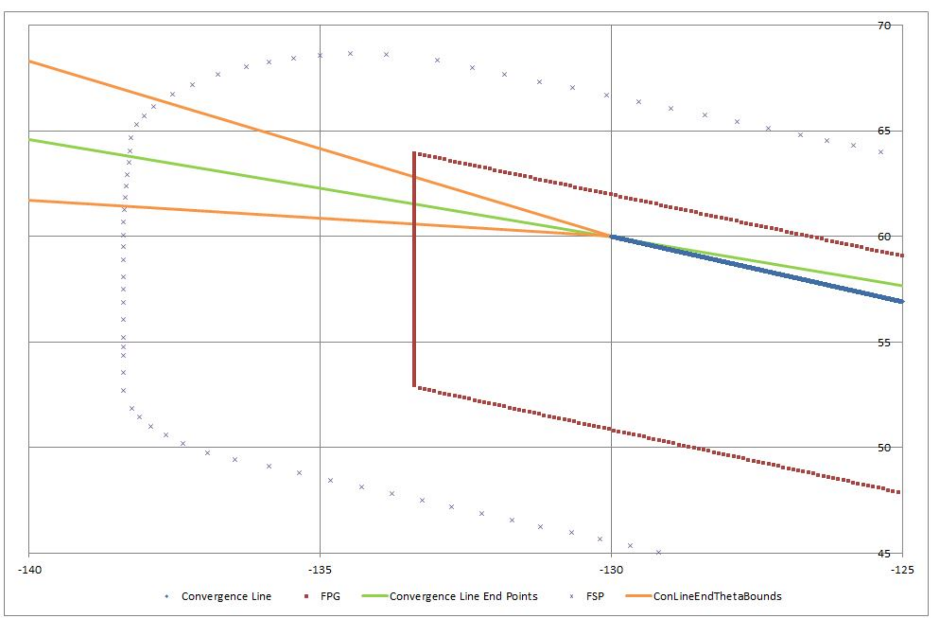


Figure 4.5: Graph showing the range of theta values within which FSPs belong to the convergence point at the end of the convergence line

Algorithm 1 Finding the *index* of the convergence point applicable to this POI

```
Min = 0.5, i.e. the smallest number which (when rounded) will refer to the second point on
the convergence line.
Max = Number of points on convergence line - 1.5000000001, i.e. the largest number which
(when rounded) will refer to the penultimate point on the convergence line.
Difference = Max - Min

if (θ is between λ - 15º and λ + 15º) // Use FUNCTION IsAngleBetween
    Index = 0
else if (θ is between φ - 15º and φ + 15º)
    Index = Last index of Convergence Line
else if (θ is between λ + 15º and φ - 15º))
    NonDimDistanceAround = GetNonDimensionalDistanceAroundAnArc((φ − 15º) −
(λ + 15º), θ, λ + 15º)
    Index = Round to integer (NonDimDistanceAround * Difference + Min)
else if (θ is between φ + 15º and λ − 15º))
    NonDimDistanceAround = GetNonDimensionalDistanceAroundAnArc((φ + 15º) −
(λ − 15º), θ, φ + 15º)
    Index = Round to integer (Max - NonDimDistanceAround * Difference)

WHERE
FUNCTION GetNonDimensionalDistanceAroundAnArc(ArcLength, Theta, ArcStart)
    if (θ < ArcStart)
        Add 2π to θ
    return (Theta - ArcStart) / ArcLength

FUNCTION IsAngleBetween(Angle, LowerBound, UpperBound)
    if (LowerBound > UpperBound)
        LowerBound -= 2 * Math.PI
    if (Angle >= LowerBound AND Angle <= UpperBound)
        return true
    else
        Angle -= 2 * Math.PI
        if (Angle >= LowerBound && Angle <= UpperBound)
            return true
    return false
```

4.4.3 Finding the nearest FMPP

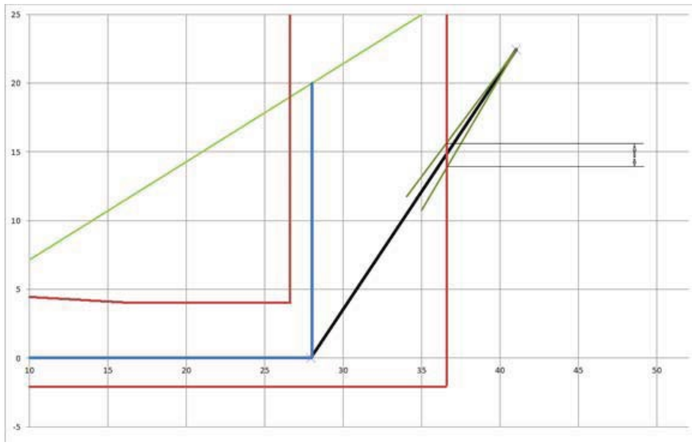


Figure 4.6: Graph showing the range of FMPPs to be considered by the POI using the 3º arc criterion

Finding the nearest FMPP is used in both primary and secondary convergence. The algorithm identifies the FMPP closest in absolute distance to the POI (Point Of Interest). However the point must be within a 3º cone (Figure 4.6) of the line between POI and convergence point. If a point cannot be found within this cone the method will report an error, the solution is to re-mesh the FMP shape with a finer mesh. The reason behind this technique is to prevent the POI being influenced by closer FMPPs than one in front of them, where they are passing it and a more relevant point is directly in front. The angle of this cone is determined by the fact that at least one FMPP must exist within it, and is thus determined by the mesh spacing of the lines which make up the FMP section. It is used to avoid an FMPP substantially outside the line of travel of the FSP having an effect on the movement of the FSP. Note that in Section 7.5 there are is a graph which demonstrates the effect of an inadequet number of FMPPs. Note also that for a forming process which can achieve closer to net shape the spacing of the FMPPs must be reduced or the three degrees increased to preserve the requirement for at least one FMPP to be present inside the arc.

4.4.4 Adaptive Grid Spacing

Adaptive Grid Spacing is used to provide a predicted forming shape with approximately equidistant points. The algorithm works by identifying the distance between adjacent points, if it is less than a pre-set minimum (m) then the second point is removed, if it is greater than 2.1m then a new point is added with a theta midway between the two points, and a radius of the average between the next and previous radii.

4.4.5 Import of Final Machined Part (FMP) shape

As shown in Figure 4.1, the first step is to import the outline FMP shape from NX sketches. The lines which comprise this shape are meshed using an equidistant (user specified) point spacing including start and end points. Where a non-integer number of points is requested the number of points is rounded up. The order of specification of the lines and the start and end point of the lines is immaterial in this algorithm.

4.4.6 Creating Initial Shape

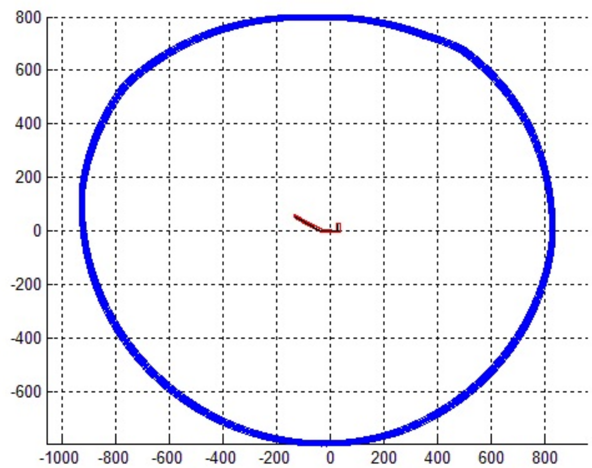


Figure 4.7: Initial shape for test case zero: A shape of constant radius (each point is a fixed radius from its convergence point)

A large closed-contour is created around the meshed shape (Figure 4.7). Each point on the closed-contour has a constant theta value and one degree of freedom (radius). This theta and radius determines the vector from the POI to its convergence point. Clearly this shape will be circular if and only if the convergence line is infinitesimal. With a straight, single segment convergence line with an infinite number of points, an ellipse would be generated.

4.4.7 Primary Convergence

The algorithm for primary convergence is shown in Figure 4.8.

Points move to 10 units (or slightly more) away from their nearest FMPP. If the POI, nearest FMPP and the convergence point are exactly aligned then the POI FMPP distance after movement will be 10 units, if the POI to FMPP line is 3^{o} different from the POI to CP line (the maximum allowed) and assuming a FMP mesh of 2 units spacing, then after movement the POI will be 10.023 units away if it starts from an 19.08 units away from the FMP (Figure 4.9). The reason behind this algorithm is secondary convergence is relatively

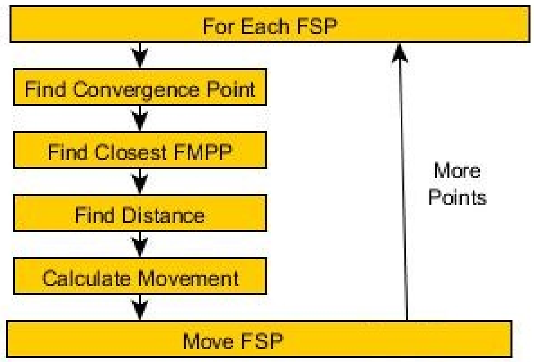


Figure 4.8: Primary Convergence Algorithm

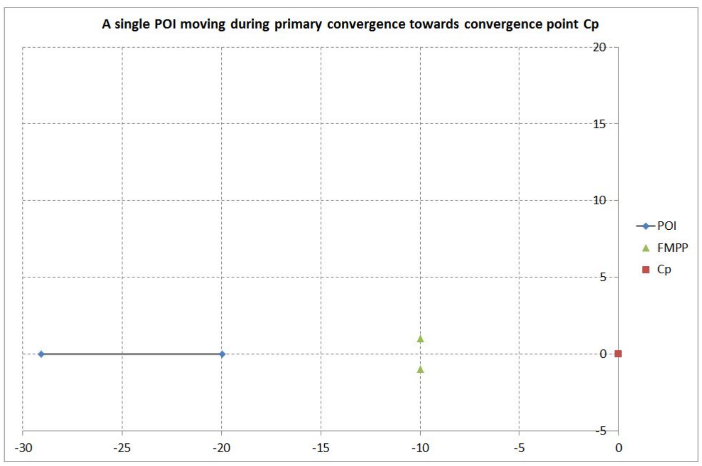


Figure 4.9: Movement of one POI where FPG mesh is two unit spacing and Cp is at (0,0)

slow and computationally expensive, this algorithm reduces the distance secondary convergence must cover, thus reducing time taken. The end result for primary convergence is shown in Figure 4.10. As can be seen the shape is now much closer to the finished part geometry, giving much less distance for the secondary convergence algorithm to cover.

The reasoning behind the ten unit value is, forging for the type of components modelled would be expected to leave a stand off distance of six units. The number of 10 units therefore results in bringing the POIs close to the FPG but not so close that they would immediately move away. It would be feasible to set the value to exactly the stand off distance for the manufacturing method thus minimising secondary convergence time, but this would not change the end result.

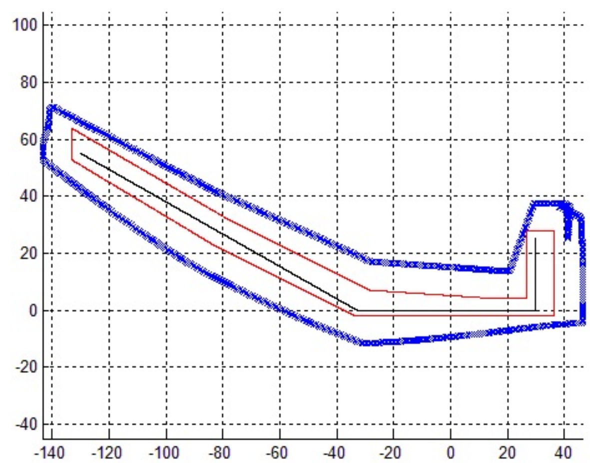


Figure 4.10: After the primary convergence algorithm, test case zero

4.4.8 Secondary Convergence (for n2 iterations)

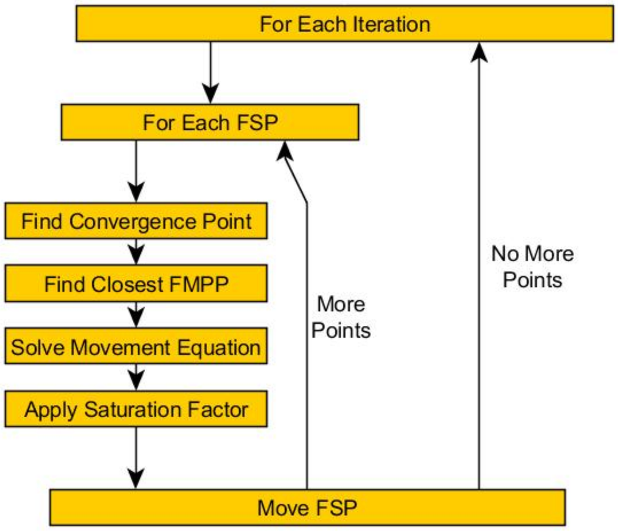


Figure 4.11: Secondary Convergence Algorithm

The points are drawn along their line of freedom according to: A force pulling them towards their convergence point (F1) A force pulling them towards the nearest FMPP (F2) A force pushing them away from the FMPP (F3) So the red points (FSPs) are pulled in where further away from the blue points (FMPPs) and pushed out where closer to the FMPPs.

Movement governed by following equations:

Definitions:

$$C_P = \begin{pmatrix} x_c \\ y_c \\ 0 \end{pmatrix} \tag{4.1}$$

$$L = c_p + r \begin{pmatrix} \sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} \tag{4.2}$$

$$n = r \begin{pmatrix} \sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} \tag{4.3}$$

$$\hat{n} = \frac{n}{|n|} \tag{4.4}$$

$$B = L - M_p \tag{4.5}$$

$$\hat{B} = \frac{B}{|B|} \tag{4.6}$$

Forces

$$F_1 = k_1 \cdot \hat{n} \tag{4.7}$$

$$F_2 = k_2 \cdot \hat{B} \tag{4.8}$$

$$F_3 = k_3 \cdot \frac{\hat{B}}{|B|^2} \tag{4.9}$$

$$F = \sum_{n=1}^{3} F_n \tag{4.10}$$

$$m = F \cdot \hat{n} \tag{4.11}$$

Important points:

The Equations for the forces are shown above; the coefficients are tuned to the manufacturing process.

F_1 is a constant force pulling the Forming points towards the convergence point.

F_2 is a constant force pulling them towards the closest FMPP.

F_3 is an inverse squared force pushing away from the closest FMPP.

Different values of k_1, k_2, k_3 are used for different types of forming process (example values in Table 4.1). These values were found by trial and improvment. By increasing all values by the same factor, the end result should be unchanged and arrived at sooner, however, it was found that a high k_3 value will tend to result in numerical instability in what is effectively euler integration.

In Section 7.5, graphs are shown giving an example of the directions and magnitude of F1, F2, F3, and the resultant movement vector. These graphs show that when the FSP is at point (8,8) there is still a net force moving it inwards, by the time it has moved to (7,7) the resultant force is outwards. This implies that the point will come to rest somewhere between (8,8) and (7,7). Note that these graphs show that in this case the convergence point is at (0,0), there is only one FMPP within the three degree arc described in Section 4.4.3, and this is located at (3,3).

Saturation limits are placed to reduce movement where it exceeds a value x. Where x is reduced linearly to zero at the end of secondary convergence. The reasoning behind these was a tendancy in some cases for the equations to result in dynamic instability and an oscillating divergence.

		Casting	Die Forging	Roll Forging
Secondary Convergence	k_1	1	1	1
Secondary Convergence	k_2	1	1	1
Secondary Convergence	k_3	-5	-50	-50
Tertiary Convergence	r	0.1	0.005	0.005

Table 4.1: Convergence co-efficients - Example values of k and r used

4.4.9 Smoothing (Tertiary Convergence)

This is executed iteratively until all points exhibit no more movement or until a user defined iteration limit is reached.

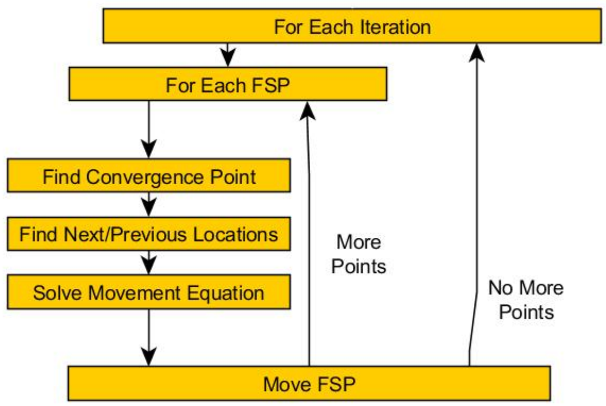


Figure 4.12: Tertiary Convergence Algorithm

Where a point is too far inside a line traced between the previous and next points, the radius is increased to meet the requirement.

As shown in Figure 4.13. This is done by finding the vector between the points either side of the POI, drawing a line (A) between them, calculating the shortest vector (B) between the POI and A. The allowable magnitude of B is defined as r$|A|$ where r is a value defined by the forming process (example values in Table 4.1). If $|B| >$ the allowable magnitude then the point is moved out along its position vector until it no longer exceeds the allowable magnitude (Algorithm 2). Clearly moving a point out may mean neither, one or both adjacent points now deviate by more than the allowable amount, so the algorithm repeats until no points deviate excessively. The end result of tertiary convergence is shown in Figure 4.14. The process is of course iterative because as one point moves out, its neighbours then become too far in. The method used, simply repeating this process is less efficient than possible, more efficient ways of knowing which points might need to move (i.e. the neighbours of moved points) could be implemented, but this would have no effect on the final solution.

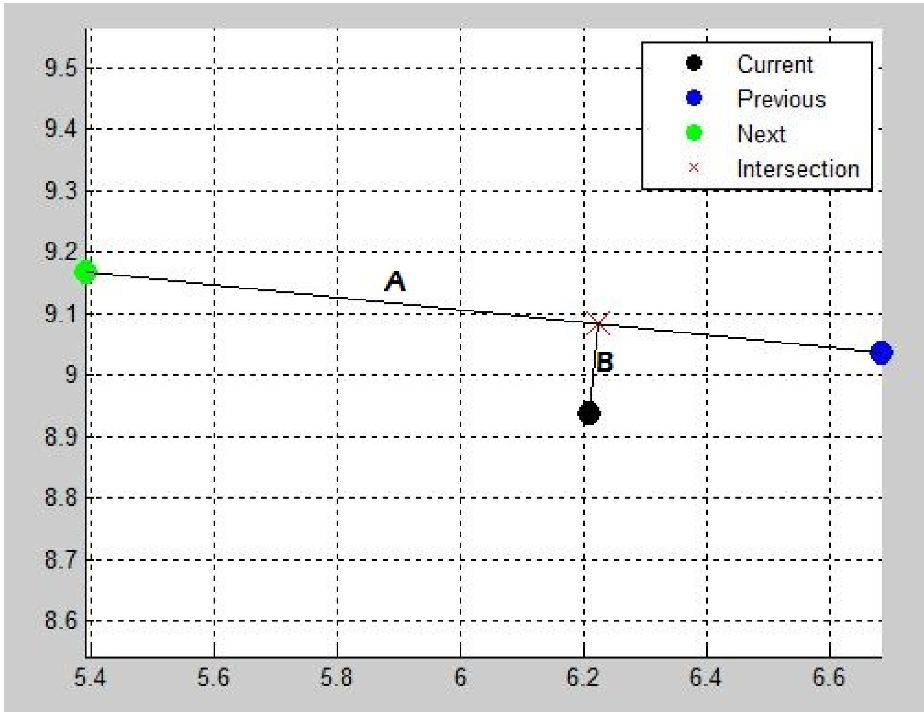


Figure 4.13: Teritary convergence - previous, current and next and vectors

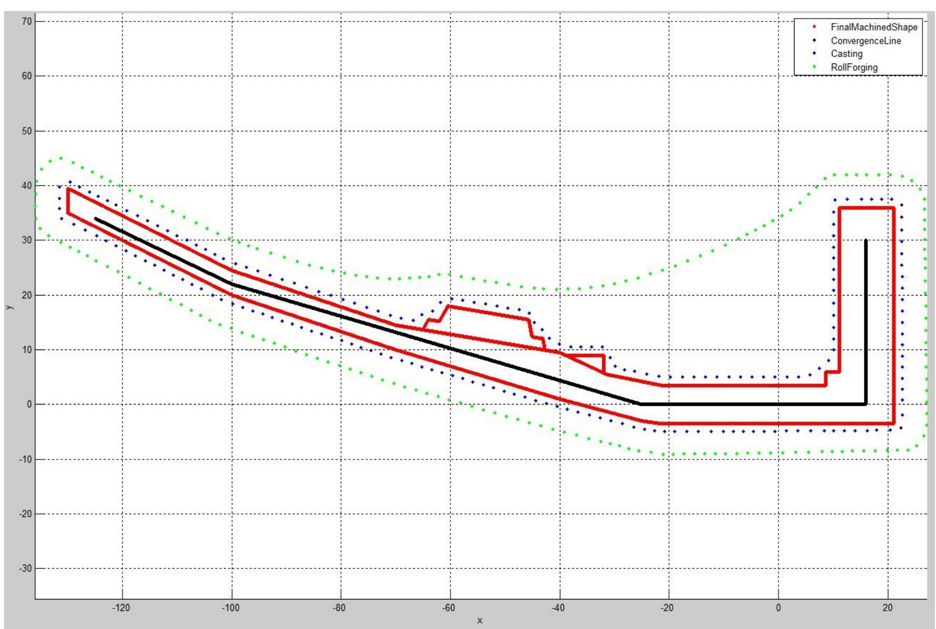


Figure 4.14: Test case two - end of tertiary convergence

Algorithm 2 The tertiary convergence algorithm

For all POI
    if (DotProduct(B, UnitVectorToConvPoint) < 0) // if point is inside VectorNextPrevious
        if (|B| > r / |A|)
            Successful = false; // When no points move in the iteration, tertiary convergence will terminate.
            Movement = |B| - (r / |A|) * 0.999 // This value will always be positive.
            Increase the radius of POI by *Movement* // A positive value will increase radius of the POI by 'Movement'

4.4.10 Fixed points

In FoSCo2d, there is the option to fix edges of the forming shape. This is to allow for constraints such as in blisks where the centre is not hollow, not a fundamental requirement of forging but required by strain rate restrictions and cost reasons. Rules such as this could be included within a rule based method.

At the present, it is allowed for by specifying an angular range and a fixed x or y displacement. It allows for effects such as shown in Figure 4.20. In this example, points with a theta of 270 degrees ± 3 are specified to lie n mm below the absolute origin of the model within the geometry creation CAD package. This corresponds to the location of the axis around which the shape is revolved. Three degrees was used to give the appropriate width of material at n mm below the origin of the CAD package. This angular range and displacement for fixed lines could be deduced from data read from the CAD package in future through the use of appropriately named lines or a *.dlx* based solution.

This capability was used to generate the disk forging shape shown in Figure 4.19.

4.4.11 Import of created shape into CAD package

The forming shapes are imported back into NX as new sketches, the 2d version is imported as a ring of points onto the same plane as the sketch from which the main shape is revolved).

To analyse the depth of material removal required, the start and end points of each relevant line in the model are analysed to find the closest point in a direction perpendicular to the type of line (i.e. it is defined as a left, right, top or bottom line). A reference dimension is then created in the appropriate direction for later export from NX. These reference dimensions can be used in the interface spreadsheet between NX and MCOST in the knowledge they will be regenerated with the same names when the part is modified. Dimensions are created automatically according to the systematic naming convention: "COS2" + "END" OR "START" + Name(Line).

4.4.12 FoSCo 2d Results

The graphs below (Figure 4.16, Figure 4.17) show plots FoSCo 2d on test cases zero and one. It can be seen that the convergence of casting for test case zero is imperfect, this is due to the nature of the algorithm for finding the convergence point, and could be rectified by using adaptive grid spacing for the convergence line. On test case zero there is also some stagger in the points around the edge of the flange, caused by a mismatch of convergence points, i.e. in radial co-ordinates they are not adjacent. This is caused by

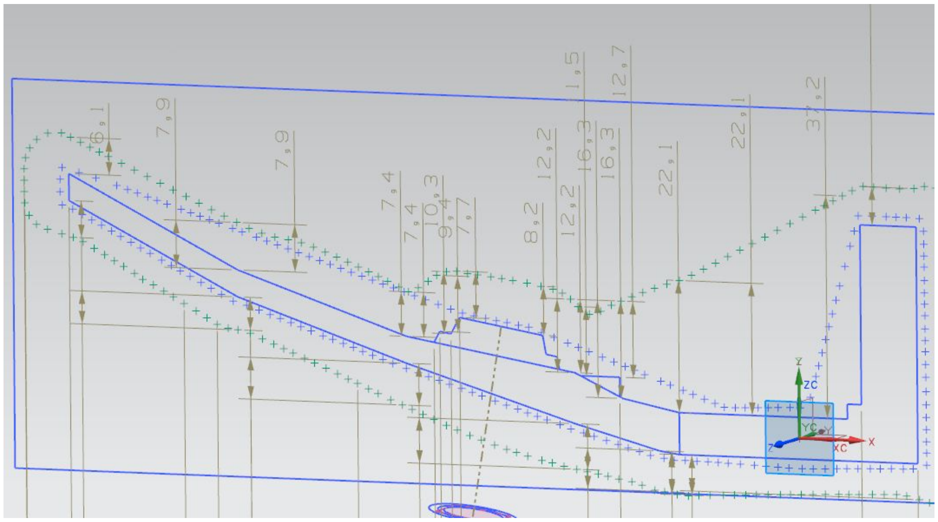


Figure 4.15: Diagram of the 2d imported point ring plus automatically created reference dimensions

primary convergence causing an irrevocable convergence, however it does not affect the validity of the shape from a cost point of view.

Adjustment of the FPG (Finished Part Geometry) Changing the FPG and the adjustments that FoSCo will make to the forming shape are shown in Figure 4.18.

In this way, by running FoSCo and MCOST after design changes have been made, the effect on the cost can be rapidly evaluated, as of course the changes will affect the machining depths and thus times and costs of the manufacturing operations.

FoSCo was also run for blisk geometry, the resultant shape (Figure 4.19) is generated for a blisk. The co-efficients used for the k values and smoothing algorithm were identical to those used for ring roll forging for test cases zero and one, suggesting the flexible validity of the method. The only requirement was to set fixed points to account for a roll forging (not hollow) as opposed to a ring roll forging (hollow).

4.4.13 Calibration and testing

FoSCo 2d was calibrated against expert estimation of the die forging shape for a rear outer casing. It was then tested against a blisk (using the same k_n co-efficients) Table 4.2. The three images in Figure 4.20 show: the finished part geometry, an expert's estimation of the die forging shape (although the expert provided a rectilinear estimate rather than the curved shapes normally associated with die forging), FoSCo's estimate of the die forging shape. To use FoSCo for die forging it was also necessary to incorporate a method of preventing hollow shapes as these are not created using die forging, this was implemented by fixing some of the points (in the ring of points) at a specified radius. This fixing was

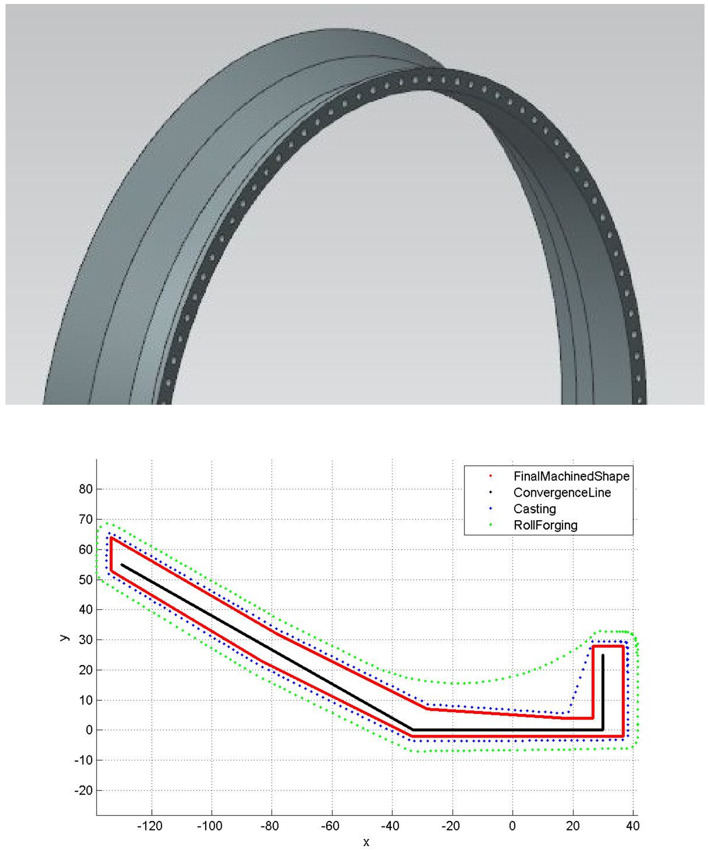


Figure 4.16: Test Case Zero, Simplified RoC (Rear Outer Casing) and its forming shape plot

completed manually through code but the requirement to fix the central section at non zero thickness is not a fundamental constraint requirement of die forging. The industrial sponsor's die forged shapes do not have zero thickness due to cost reasons and structural consequences of the strain rate required to achieve zero thickness.

Description	Type	Volume (dm^3)
Finished Part Geometry	FPG	8.68
FoSCo estimate (co-efficients as for ring roll forging)	Black Forging	21.39
Sponsor's Estimate (manual)	Black Forging	22.13 (+3.3%)
Sposor's Estimate from rules based tool (ACE)	Black Forging	23.27 (+8.08%)

Table 4.2: FoSCo's estimate of forging volume against professional estimate from industrial sponsor

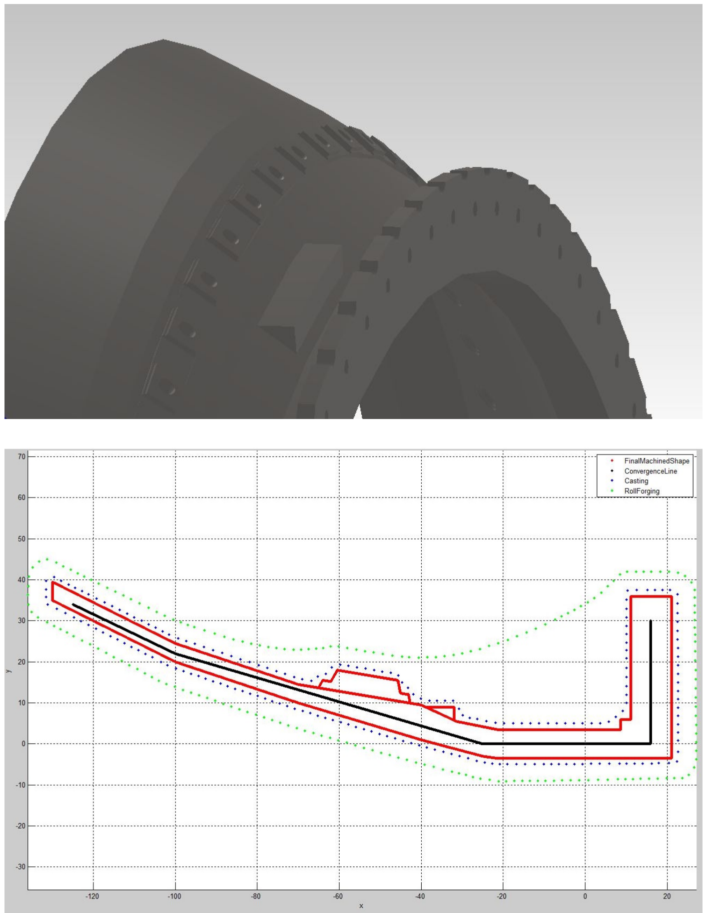
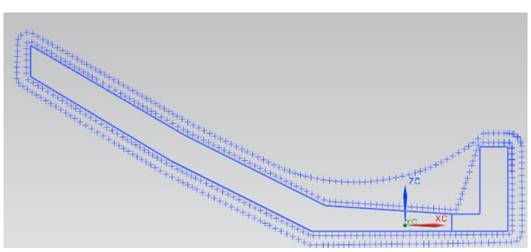
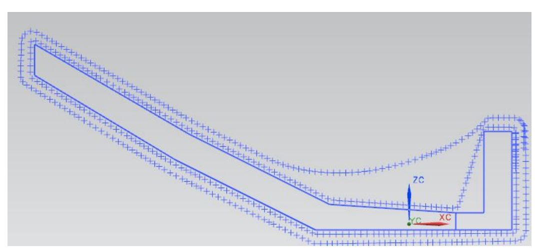


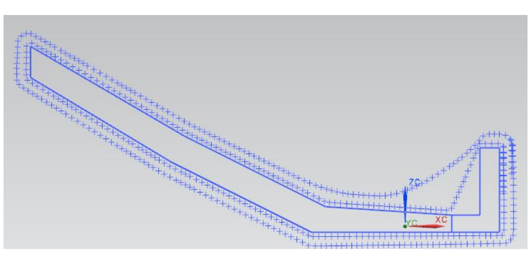
Figure 4.17: Test Case One, RoC (Rear Outer Casing) and its forming shape plot



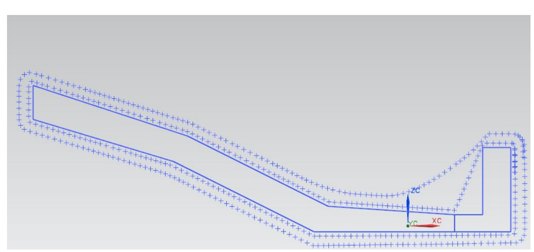
(a) Original shape



(b) Increased flange height



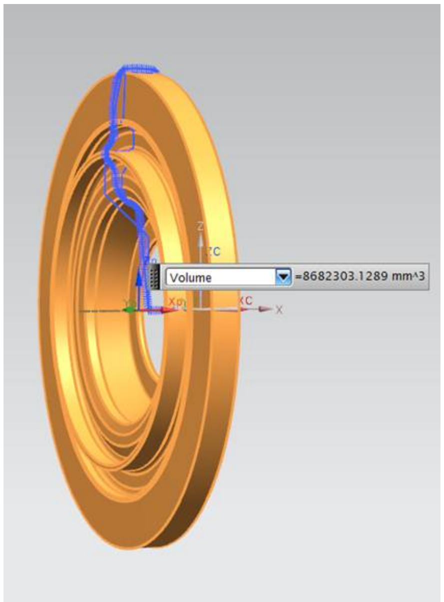
(c) Reduced flange thickness



(d) Reduced casing angle

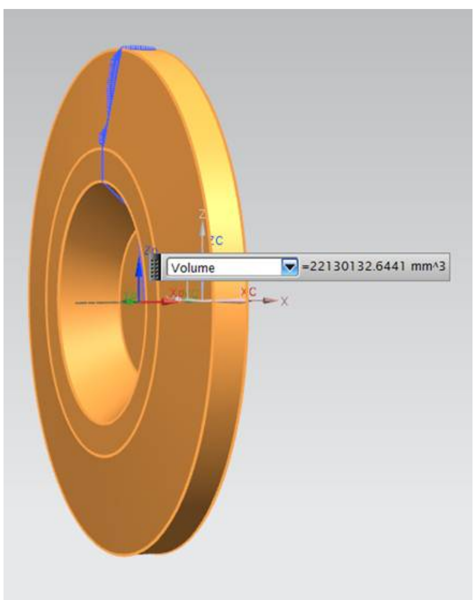
Figure 4.18: Design changes and their effect on the formed shape (of Test Case Zero)

Figure 4.19: FoSCo 2d with a blisk input as finished part geometry, governing co-efficients as set for test case zero and one

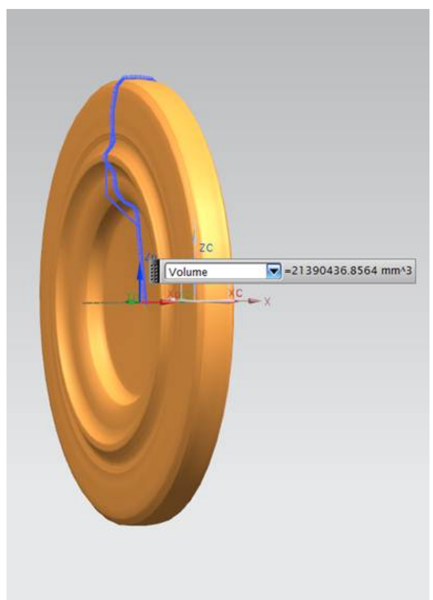


(a) Axis-symmetric finished part geometry

(b) Forging Shape - estimated by industrial sponsor using ACE (rules based tool)



(c) Forging Shape - estimated by industrial sponsor (ignoring fillets)



(d) Forging shape - estimated by FoSCo

Figure 4.20: Blisk FoSCo 2d testing

4.4.14 FoSCo 2d on minor test cases

As discussed in the introduction, an additional 20 test cases were provided by the industrial sponsor, these were the same as used by the sponsor to investigate the quality of the commercial tool aPriori. Ten of these were mainly axis-symmetric shapes and thus suitable for FoSCo 2d. These test cases and the forming shape predicted by FoSCo 2d are shown in Table 4.3.

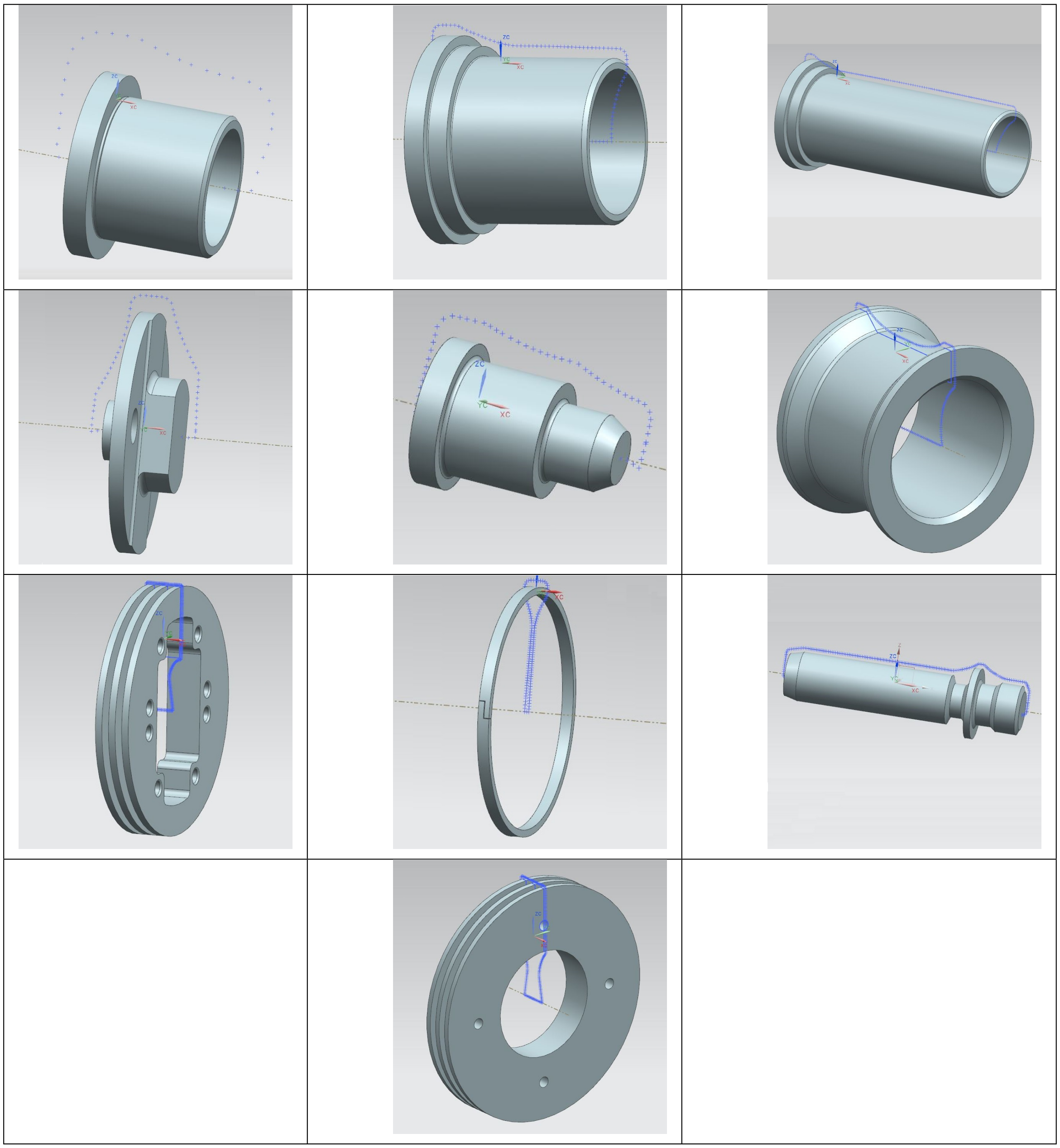
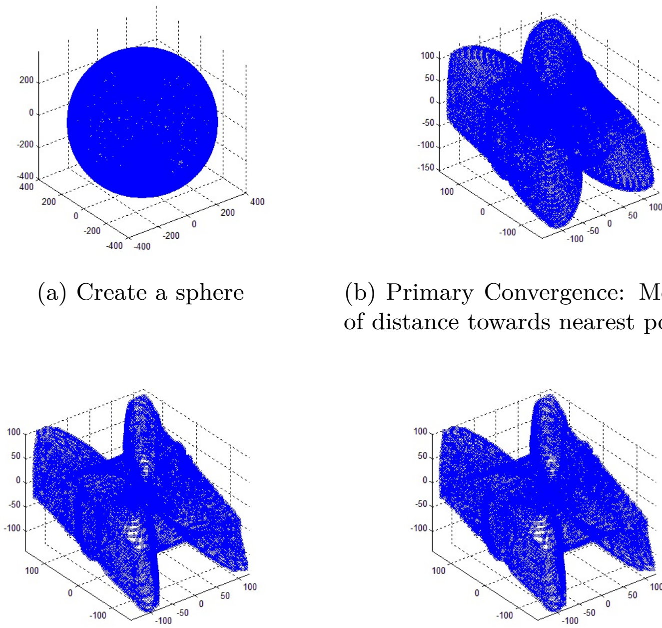


Table 4.3: FoSCo 2d on minor test cases

4.5 FoSCo methodology (3d)

The 3d version of FoSCo is less developed than the 2d version, simple further work is outlined and is left as further work. Plots of the overall methodology are shown in Figure 4.21. Finding the closest FMPP is not restricted to Cartesian distance without any consideration of angle as in Section 4.4.3.



(a) Create a sphere

(b) Primary Convergence: Move 90% of distance towards nearest point

(c) Secondary convergence: based on rules and equations above

(d) Finished

Figure 4.21: Stages of FoSCo 3d convergence

4.5.1 Initial shape

The 3d version is similar to the 2d version except that the 3d version starts with a sphere of points created from multiple circles. The nature of the co-ordinate conversion used creates the effect shown in Figure 4.22. FoSCo3d at present uses only a convergence point, rather than a line, polygon or polyhedron. This could potentially be future work to improve convergence.

1. Parallelise for all forming methods to be considered.

   (a) For n iterations

      i. The points are drawn along their line of freedom according to:

         A. A force pulling them towards the origin

         B. A force pulling them towards the nearest mesh point

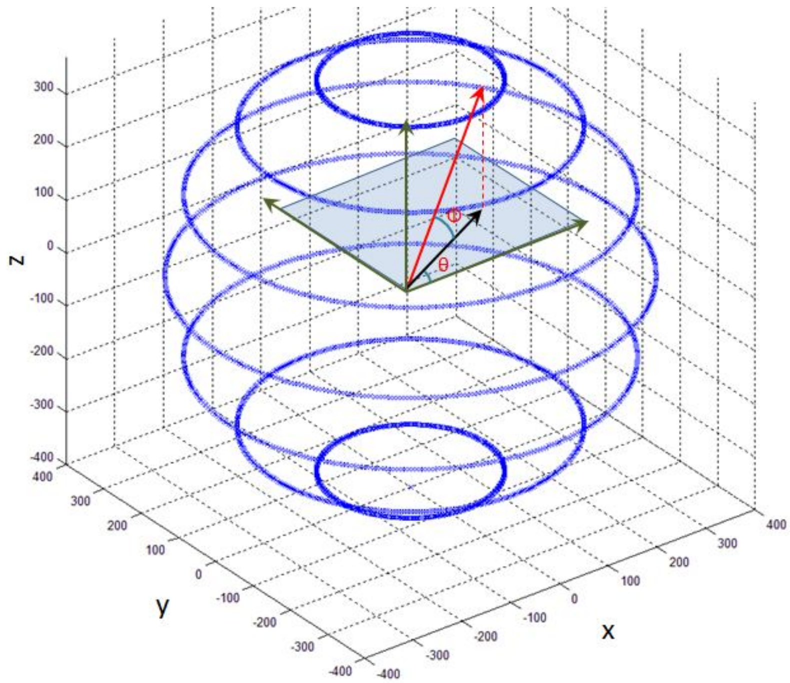


Figure 4.22: The initial placement of points (example shows 7 circles, 400 points per circle)

C. A force pushing them away from the mesh point

ii. Their movement is then calculated and the iteration is complete.

(b) The creator is complete when a set number of iterations have occurred.

As shown in Figure 4.1, initially a mesh is imported directly from an NX surface mesh export.

The methodology then divides into parallel threads, one for each manufacturing strategy. The surface of a sphere is created around the meshed shape where each point on surface has a radius of the radius of the furthest point on the FMP plus 10mm.

The convergence towards the forming shape then consists of primary and secondary convergence phases, with differing governing equations. During convergence the points are allowed to vary only in radius and the user specifies the number of iterations in the primary and secondary phases. Adaptive grid spacing is used in the 3d version, but only between points on the same circle (points of constant φ). Changing this to consider spacing between adjacent circles is left to further work.

4.5.2 Primary convergence

For primary convergence the points are moved inwards 90% of their distance to the nearest FMPP. This is simply to bring the circle of points closer to the FMP to minimise the number of iterations required in the secondary phase. The algorithm used is the same as a previous version of the algorithm used in Section 4.4.7. In the 2d version it was replaced with an algorithm to move the point to approximately 10 units away, which is more computationally efficient.

4.5.3 Secondary convergence

For secondary convergence the movement value of each point is calculated using Equation 4.12. During secondary convergence adaptive grid spacing is used as in the 2d version (Section 4.4.4).

The equation which defines the movement of the point during secondary convergence is:

$$m = (M \cdot \hat{n})\, t_s\, (1 - f_d) \tag{4.12}$$

$$M = \sum_{n=1}^{3} M_n \tag{4.13}$$

$$\begin{aligned} M_1 &= k_1 \hat{n} \\ M_2 &= k_2 \hat{D} \\ M_3 &= k_3 \hat{D}/d^2 \end{aligned} \tag{4.14}$$

$$\hat{D} = \frac{L - N_P}{|L - N_P|} \tag{4.15}$$

$$\begin{aligned} \hat{n} &= \begin{pmatrix} -\cos(\theta)\cos(\phi) \\ -\sin(\theta)\cos(\phi) \\ -\sin(\phi) \end{pmatrix} \\ L &= r \begin{pmatrix} \cos(\theta)\cos(\phi) \\ \sin(\theta)\cos(\phi) \\ \sin(\phi) \end{pmatrix} \end{aligned} \tag{4.16}$$

It can be seen that the equations for the 2d version are the same equations (in 2d ϕ is always zero, so z is zero).

As with the 2d version the 3d version also has checks to ensure smoothness and to check that the point ring does not violate the finished part geometry. However, whereas the 2d version has a separate algorithm for smoothness implemented after secondary convergence, in the 3d version smoothing is at present located within secondary convergence.

For smoothness

If (AverageRad > ThetaRadContinue.Item2[i0] - Movement + Forces[3])

Movement = -(AverageRad - Forces[3] - ThetaRadContinue.Item2[i0]);
//Movement = -0.5;

This means that if the average radius of the radii of the two adjacent FSPs exceeds the (after movement) radius of the FSP under consideration plus an allowed amount of divergence, the movement is reset to -0.5. This creates a small outward movement. The lower Forces[3] is set the smoother the shape will be.

Despite the 3d nature of the sphere representing the FSPs there is no interaction between adjacent circles making up the sphere, this is left as further work. To accomplish this is a minor extension of the work to identify which of the points on adjacent circles is closest since not all circles have the same number of points on them.

For excessive convergence

if (Abs(Movement) > Forces[0] + Forces[1] + 1) // too much movement, point is probably within FMS (and thus being pushed inwards by M_3.

Movement = -1;

This is effectively a saturation monitor. If the movement is too great either inwards or outwards then the movement is set to -1, a small outward movement, the movement will normally be too great after the FSP violates the FMP since there will be no balancing force after violation.

4.5.4 Output and NX import

The final shape of the point surface is then exported to a .txt file which is used by code written in NX to import the point cloud as a 3d surface (graphics can be enabled or not) to which is used to analyse the depth of material removal required:

1. A 3x3 grid is placed over each surface in the component
2. The normal direction at the centre of the surface and its location relative to the NX origin (used as a proxy for centre of gravity in the test case) are analysed to give an indication of the machining direction.
3. The closest forming point in the appropriate direction (see 2) is found, and the distance in that direction is stored for the 3x3 grid. The average is calculated from the points to give an average depth of cut for that face. The average is unweighted because the algorithm which constructs the 3x3 grid provides points where the area closest to each point is approximately equal.

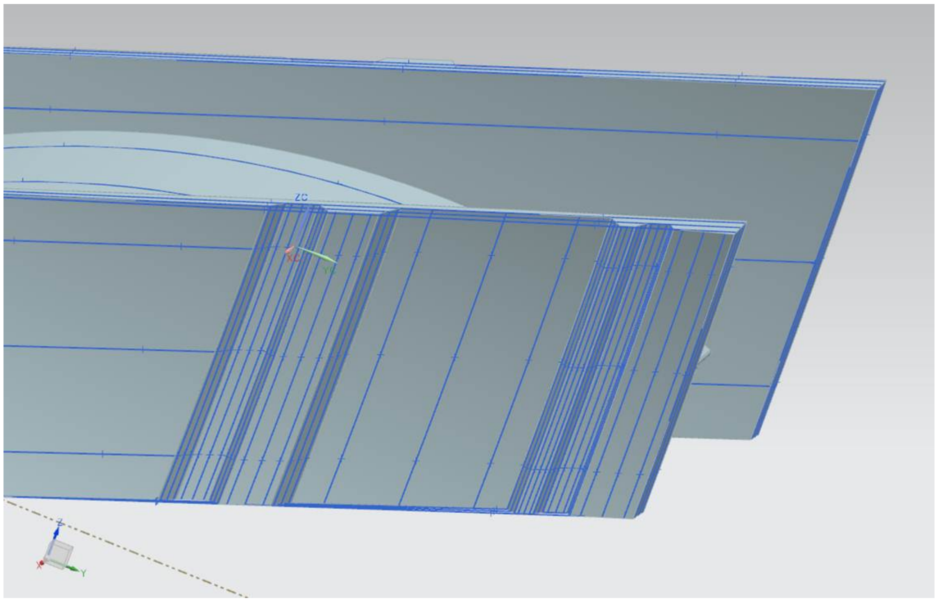


Figure 4.23: Diagram of the face surface grid created

4.5.5 Results of 3d FoSCo

The results of FoSCo 3d for test case two (4.25) for several forming processes are shown in Figure 4.26.

The result of FoSCo 3D for ARCAM and casting (Figure 4.26) is clearly inaccurate for a near net shape process. However, where supporting structure is required this could potentially be modelled using the approach used in FoSCo 2d (Section 4.4.10 Fixed Points). To model the contours of a net shape accurately however the two dimensional approach of a convergence line must be expanded, to allow a polygon in three dimensional space. This was not explored in detail but alluded to in the future work section. This lack of accuracy for near net shape processes is the cause of overestimation of the cost of test case two by MCOST (Section 5.12.4).

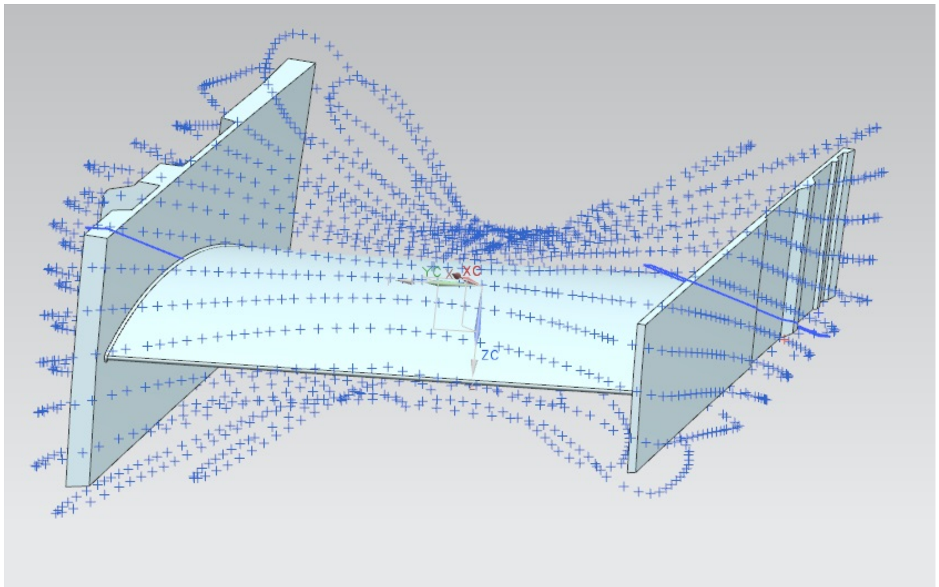


Figure 4.24: Point cloud imported into NX

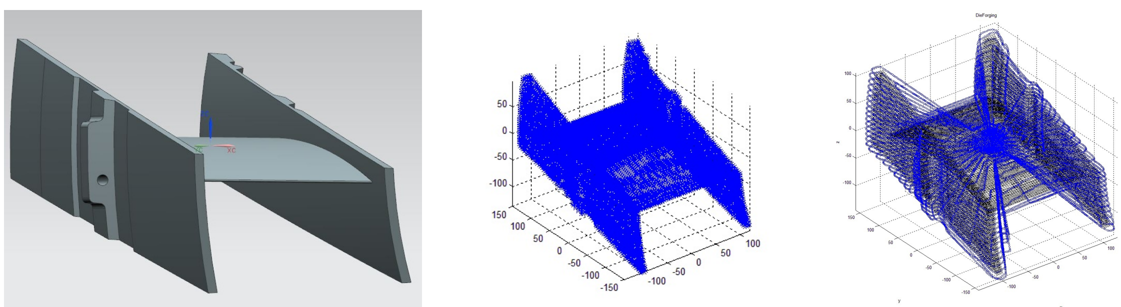
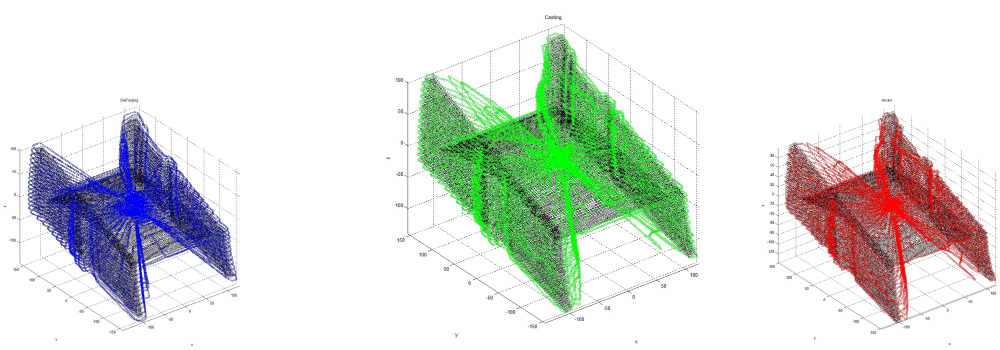


Figure 4.25: Test Case Two, ESS (Engine Section Stator) and its forging shape plot



(a) Die Forging (b) Casting (c) ARCAM

Figure 4.26: FoSCo 3d results for ESS

4.6 Alternative modelling methods

Alternative methods were considered, a rules based method was considered unattractive due to the time required to populate the rules for the specified component to be examined. Alternative simple methods were an offset method, where lines are simply offset a specified distance x, and a rolling circle method proposed by Toal (2012) where a circle traces a line around a shape and the centre of the circle delineates the forming shape. These were considered and found to be inadequate for a key aspect of commonly found geometry, an axisymmetric groove.

As can be seen in Figure 4.27d the issues with the forging estimation in the line offset or rolling ball method are not encountered with FoSCo, FoSCo does have casting shape generation issues, the rectification for which will be discussed in Section 6.4 but it essentially could work by allowing concentration of the convergence line points in compact regions of the finished part geometry.

The issues with the two methods below could of course be resolved with a smoothing algorithm. However, it is difficult to see how (in three dimensions) the smoothing could be resolved without the benefit of a spherical co-ordinate system giving adjacancy meaning. This adjacancy, resulting from a closed contour in polar co-ordinates (for axis-symmetric shapes), was used to allow the concept of neighbours. The concept of neighbours is essential to the generation of smooth shapes in two or three dimensions and therefore to this method.

4.6.1 Rolling Ball Method

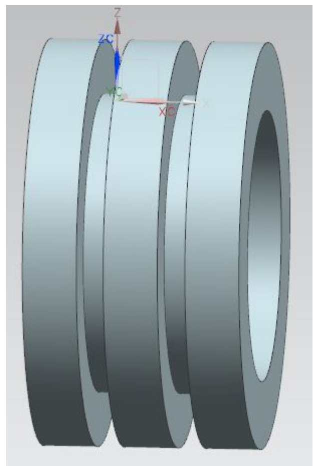
A method of calculating the forming shape using a rolling ball was proposed. Where a ball would roll round the shape and the trace of the centre of the ball would denote the forming shape. This method, in addition to its other issue discussed below results in sharp internal corners and smooth external corners.

A short example of where this method would not accurately succeed is below. Let us assume we have a component as shown (a pulley) with two grooves (a and b) and we are attempting to approximate its cast or forged shape using a rolling ball of diameter d_b. The grooves have widths of $d_b \pm \delta$. The FMP shape and the forming shape which would be generated by a simple ball are shown in Figure 4.27b. This method would allow an infinite depth of an infinitesimally thin groove for a groove width of $d_\mathrm{b} + \delta$ which is clearly nonsensical in the context of manufacturing forming methods. In addition it results in internal vertices with infinitesimal radii.

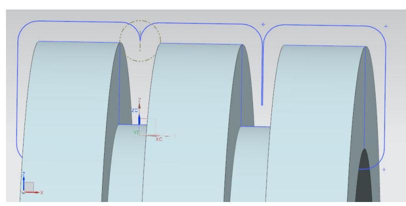
4.6.2 Simple offset Method

This method results in sharp external and internal apexes, discontinuities of gradient are a physical impossibility in manufacturing processes but could result in a good enough approximation in some forming processes. Particularly forming processes which can achieve sharp vertices and edges.

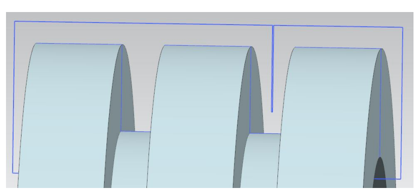
A simple line offset method previously mentioned as an alternative would produce a shape as shown in 4.27c, with discontinuities in the gradient for internal and external corners, and the same problem of infinitesimally thin grooves as in the rolling ball method.



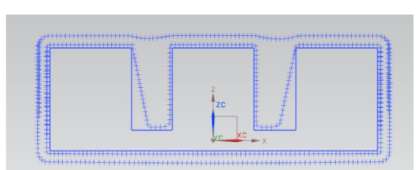
(a) Final Machined Part (FMP)



(b) Rolling ball estimate, forming shape of (a)



(c) Line Offset estimate, forming shape of (a)



(d) FoSCo 2d

Figure 4.27: Illustration of alternative forming estimation methods

5 Manufacturing Choice Optimisation and Sequencing Tool

5.1 Introduction to Manufacturing Choice Optimisation and Sequencing Tool (MCOST)

The objective of MCOST is:

> Given the finished part geometry and tolerance data, parameterised as a set of design features, and given the precursor shapes for one or more manufacturing strategies (i.e. the choice of primary forming process) find the lowest cost feasible method of manufacture.

Thus MCOST is an optimiser around a feasibility and cost model which finds the optimum manufacturing route for a component after decomposing the design features into elemental features. Elemental features would be, in extremis, features which cannot be represented by combinations of other, simpler, features.

MCOST has several tasks, it must:

1. Take a set of design features and their associated geometry and convert them into elementary features (to be discussed in Section 5.7.3)
2. For G_O iterations - select the manufacturing processes which comprise the most cost effective way to make all elemental features:
    (a) Choose a set of manufacturing processes to make each elemental feature
    (b) Amalgamate the sets of processes (create Complete Process Set)
    (c) For G_I iterations - sequence the processes to meet feasibility constraints:
        i. Sequence the set of manufacturing operations (create Method of Manufacture).
        ii. Check whether the sequence is feasible, if not return to i.
            A. Model the time of the sequence (operation time is assumed independent of machine choice)
            B. Choose the machine and TAD (Tool Access Direction) for each operation
            C. Model the cost of the sequence
        iii. Continue to optimise the sequence, return to i.
    (d) Continue to optimise the choice of process sets, return to (a)

5.2 Nomenclature

5.2.1 General Nomenclature

Symbol	Meaning
FMP	Final machined part
FS	Forming Shape
b_1	time to cost model a single operation
b_2	average operation combinations per elemental feature
b_3	average operations per elemental feature
c_1	$b_1 \cdot b_2 \cdot b_3$
G_O	Maximum allowable generations in the outer loop (Process selection)
G_I	Maximum allowable generations in the inner loop (Operation sequencing)
t_{eval}	Time to cost model a method of manufacture
$\mathbf{X}$	The matrix of numbers determining choices for manufacturing process selection

5.2.2 Scrap and rework calculation

Symbol	Meaning
$P_{\text{ns},n}$	Probability of not scrapping due to a single dimension (n) success
P_{nso}	Probability of not scrapping as a result of a single operation
P_{so}	Probability of scrapping as a result of a single operation $= 1 - P_{\text{nso}}$
$P_{\text{nr},n}$	Probability of not reworking due to a single dimension (n) success
P_{nro}	Probability of not reworking as a result of a single operation
P_{ro}	Probability of reworking as a result of a single operation $= 1 - P_{\text{nro}}$

5.3 Definitions

Design Feature A feature used by a designer to achieve a certain objective (for example: affix to another component, provide structural rigidity, provide an aerodynamic surface of a specified shape). Design features can be additive or subtractive.

Elemental Feature A feature which need not be broken down into other, simpler features. Design features are composed of one or more elemental features.

Manufacturing Feature A feature created by a single manufacturing process. There is a many-to-many mapping between elemental and manufacturing features.

Manufacturing Operation A specific instance of a manufacturing process, operating on a specific elemental feature.

Manufacturing Process Any manufacturing process associated with the manufacture of the component, including inspection type operations.

Word/Phrase	Acronym	Meaning
Manufacturing Process		A manufacturing process such as turning, milling, heating, forging
Manufacturing Operation		An instance of a manufacturing process (for an elemental feature)
Manufacturing Strategy		The primary manufacturing process (the process used to make the precursor shape)
Process Set	PS	All of the manufacturing operations used for an elemental feature
Complete Process Set	CPS	All of the manufacturing operations used for the component
Final Machining Process	FMP	A manufacturing process which can create a feature to within all defined tolerances
Method of Manufacture	MoM	The complete process set after sequencing
Route Creator Structure	RCS	A datastructure which stores the CPS and the precedence relationships between processes
Route Sequencing Structure	RSS	A datastructure which stores the CPS during the sequencing phase (inner loop)
Best Inner Loop Cost	BILC	The lowest cost found in the current optimisation for this CPS
Best Global Cost	BGC	The lowest cost found in the current optimisation
Parental Feature	DFC	Design Feature undergoing Conversion (to elemental features)

Table 5.1: Meaning of words and phrases used during the optimisation process

Manufacturing Strategy The class of manufacturing process associated with the manufacture of the precursor/forming shape, it is the primary additive manufacturing process used for the component.

Process Set The set of processes necessary to manufacture all of the elemental features on/in a component.

Process Sequence The process set constructed into a sequence of processes/operations.

KnowledgeBase A central database (Figure 5.1) which stores:

1. Design features
2. Elemental features
3. Design to elementary feature mapping
4. Manufacturing processes: their capabilities, variability, cycle time, costs, predecessors and successors
5. Machines: their location, cost rates, labour requirements and capabilities
6. Intra and inter factory transport costs

Outer Loop The optimisation loop which constructs and modifies the process set in accordance with the optimiser rules. Having completed the inner loop it identifies whether to keep the previous process set or move to the new one. The data associated with the process set are stored in the RC (Route Creator) DataStructure.

Inner Loop The optimisation loop which operates fully within the outer loop. It constructs the process sequence from the process set and calculates times and costs for each operation. The data associated with the process sequence are stored in the RS (Route Sequencing) DataStructure.

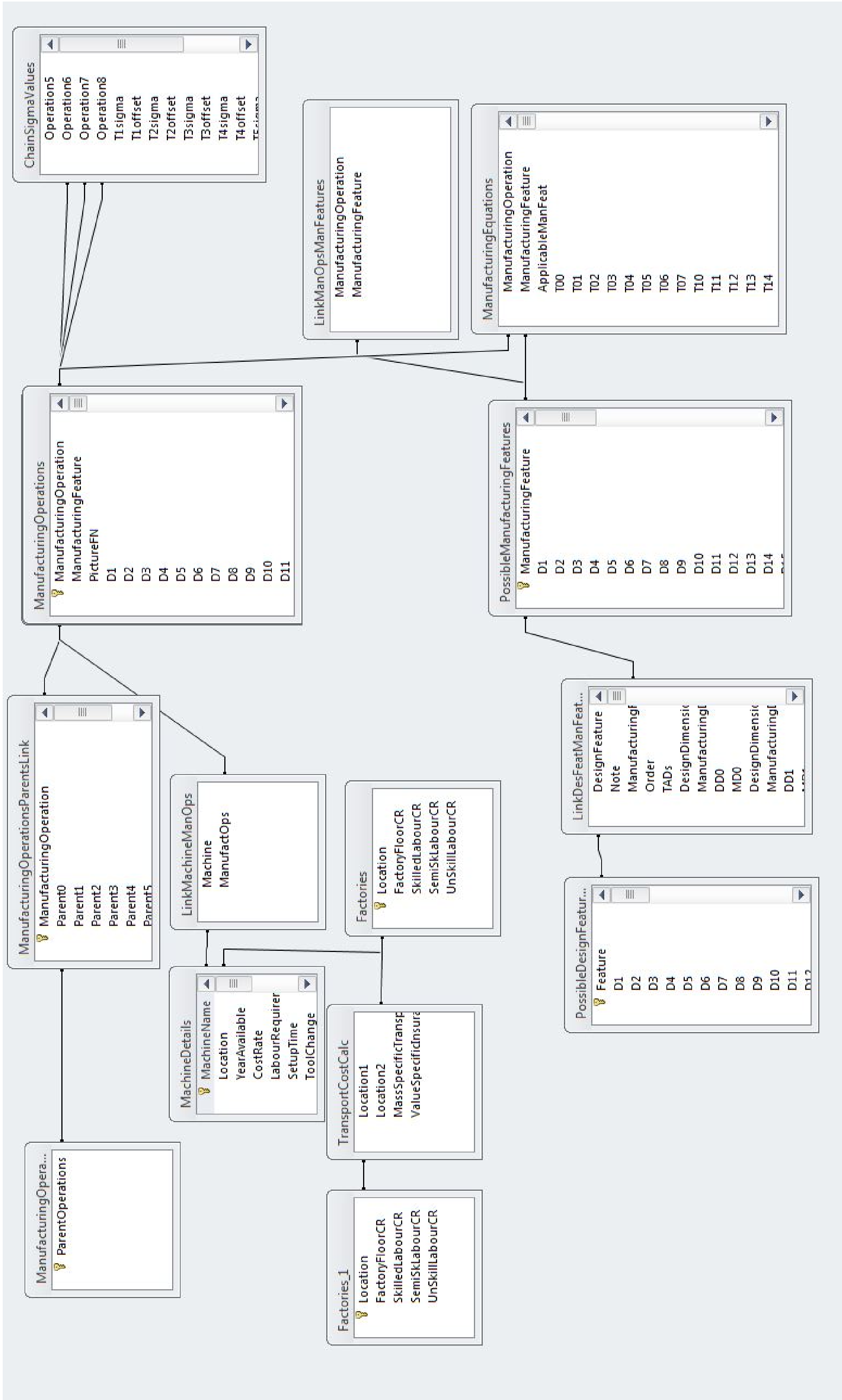


Figure 5.1: The MCOST database

5.4 Justification for a holistic view

As can be seen from the literature review, there has been substantial research on the construction of manufacturing routes. However, most of this research begins with the assumption that the Complete Process Set (CPS) is already chosen and all that is necessary is to sequence it to create a Method of Manufacture (MoM). This would only identify the global optimum MoM if the optimal Process Set (PS) for every feature was dependent only on characteristics known at the commencement of the optimisation. It will be shown in this section that the optimal PS is partially dependant on the sequence of operations and the PS for other features, and thus a method which assumes that the CPS is already fixed may frequently find suboptimal solutions. Amongst other factors the optimal PS can depend on:

1. Tolerances (of feature and other features)
2. Geometry (of feature and other features)
3. Material (of the component)
4. Starting/precursor shape
5. Process sequence

Several of these responses are shown in the following section.

This can be explored on a simple feature of a ring of holes. Given a cost prior to the operation, and a set of dimensions and tolerances, it can be determined which manufacturing process is the optimum method of manufacture for the given feature. This entails exploring which manufacturing processes give the lowest cost after completion of the operation and taking into account the scrap rate. The equation used is a simplification of the equations stored in the manufacturing process database but is sufficient to demonstrate the aforementioned point. We can use an equation amalgamating the costs into a simpler equation rather than using the detailed cost equations present in the optimisation because here the breakdown is unnecessary. The equation used for the generation of Figure 5.2 is:

$$C_O = (1 + F_R)\,(C_B + (T_0 + T_S)\,C_R) - C_B \tag{5.1}$$

In the four graphs shown in Figure 5.2, the optimal process set for the elemental feature is the lowest line at the point in the graph denoted by the independent variable.

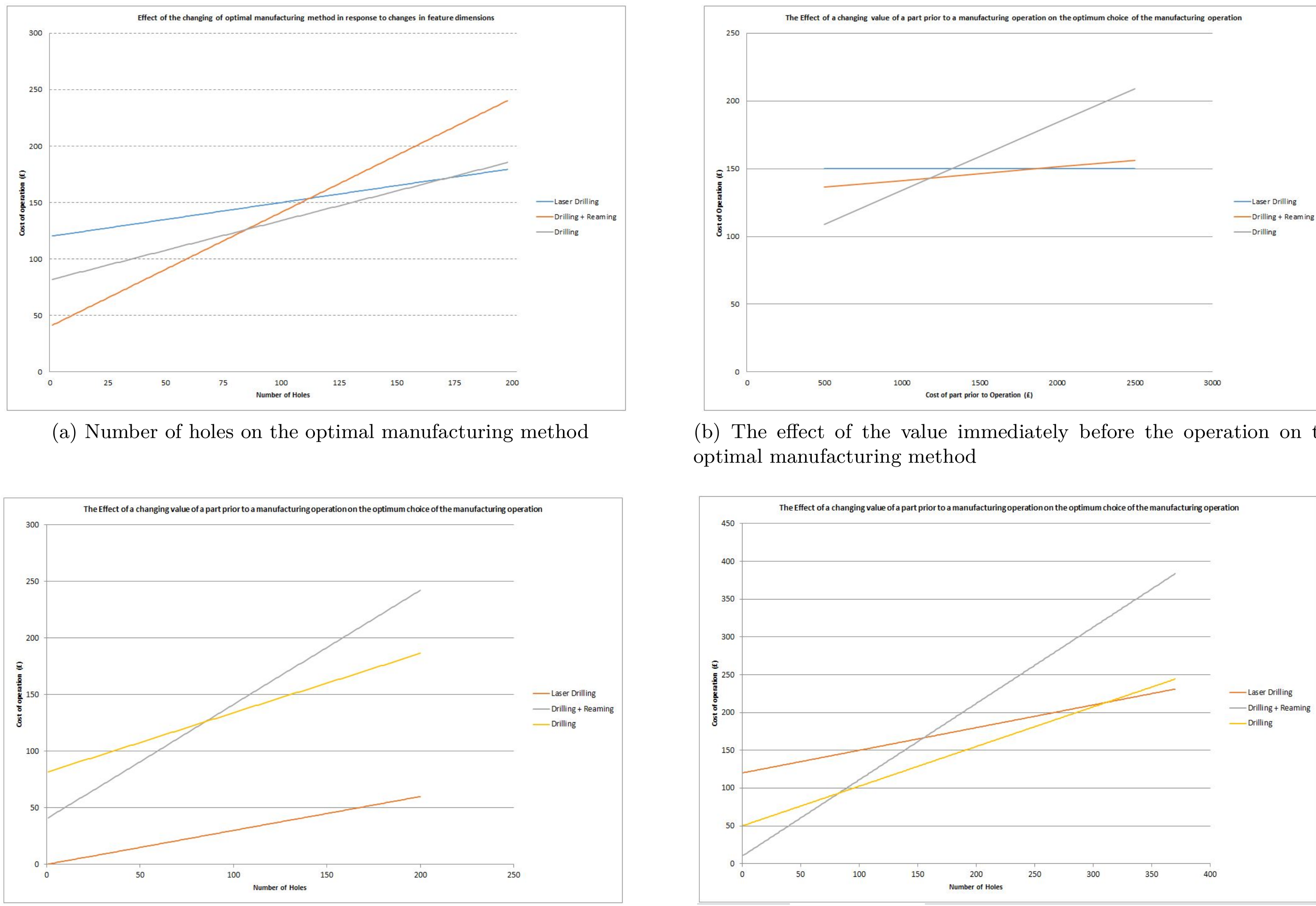
Firstly exploring the effect on the operation cost (C_O) of simply changing the number of holes (Figure 5.2a). This will explore the effect of setup time (T_S), operation time (T_O) (number of holes multiplied by time per hole), and machine cost rate (C_R). So drilling and reaming is optimal up to about 75 holes, then drilling is the least costly up to about 175 holes when the low value of the part and the higher probability of scrap fails to outweigh the cost saving from a less accurate process. Finally at around 170 holes the benefits of laser drilling outweigh the additional setup cost. It should be noted that in the methodology at present there is the assumption that the probability of creating scrap/rework is independent of the number of elements (such as holes), this is clearly untrue in some cases and the need to improve this is highlighted in Section 6.5, however it does not invalidate the argument presented in this section or this thesis. The effect espoused upon will still be present albeit at potentially different points of the independent variable.

Secondly, the effect of a change in the prior cost (Figure 5.2b). The prior cost (C_B) is the cost incurred getting the component to its current state, which will be wasted if scrap (with a scrap rate of (F_R)) is generated in this operation. Rework rate has been ignored in this example, as this simply increases the operation cost and can thus be modelled in (C_R) .

Thirdly and finally, exploring the response of the methodology to the previous machine in the operation list. If the previous operation is conducted on a mechanical drilling machine for reasons of size of the hole of the other hole based feature, the response of the next operation to number of holes is shown in Figure 5.2c, the mechanical drill should be used unless the hole count exceeds 310. Assuming now that that the other hole feature, for reasons of tolerance, cannot be satisfactorily mechanically drilled, but must be laser drilled (Figure 5.2d), and must be conducted prior to the feature under investigation. Now assuming that there are no constraints requiring separation of manufacturing of the two features. The setup cost of the laser drilling method can now be zero, and the optimal manufacturing process for any number of holes (in this example) is laser drilling.

The method presented in this section of the thesis will thus change the manufacturing method in response to both feature level stimuli and route level stimuli, the latter being of particular interest and novelty. All of the above changes result in a change in the choice of manufacturing method for a given feature. The first is a change internal to the feature but the others are external and are therefore harder to predict as a result of epistasis. There are many other responses of the optimum manufacturing method for a feature in response to both internal (to the feature) changes and external.

It can therefore be seen from the examination of these cases that asking the question, "*What is the least expensive manufacturing process for this feature?*" is not at all straightforward. the answer depends not only on the characteristics of the feature, but also on the preceding manufacturing route, both the immediately preceding operation and all others, where a possibility of generating scrap exists. Given this logic it also follows that the optimal choice of process also depends on the immediate and subsequent downstream processes. When this argument is accepted, it follows that a logic based process of decisions made on incomplete information but affecting subsequent decisions will sometimes fail to reach an optimal decision; this leaves exhaustive search and optimisation routines, and exhaustive search rapidly becomes too large (Section 5.6) a computational



(a) Number of holes on the optimal manufacturing method

(b) The effect of the value immediately before the operation on the optimal manufacturing method

(c) Component already on a laser drill (varying hole quantity)

(d) Component already on a mechanical drill (varying hole quantity)

Figure 5.2: The least costly method of manufacture for a single feature (approximation)

problem to solve for all but the most simple test cases.

5.5 Optimisation

Optimisation methodologies are extensively used to explore simulation/evaluation problems where exhaustive search of a design space is intractable or infeasible. Some of the commonly used optimisation routines which are used in this research are outlined below:

Artificial Neural Net A machine learning algorithm for machine decision making, constructs a decision tree which can be modified, usually using weighted branches.

GA A genetic algorithm, comes in many variants, inspired by natural selection, usually new individuals are created using recombination of pairs of individuals and mutation of chromosomes, a variant exists with a population size of one, described below.

RMHC A specific sub-category of a GA which has a population size of one thus making recombination of pairs impossible. This type of GA relies only on mutation and keeps the new individual if it is better than the original, or discarding if it is not. There is much literature about the relative benefits of each type of GA. Initial tests were conducted early on in the research which resulted in superior performance from the RMHC (i.e. fewer calls of the objective function necessary) however it is left to future work to re-investigate.

SA Simulated Annealing, a search similar to an RMHC but which can descend to lower fitness individuals with some probability, this probability of descending to lower fitness will reduce over time, it should be considered complementary to other optimisation techniques as it does not specify the nature of the mutation, it only modifies the acceptance criteria.

5.6 Elemental features and computational limitations

As illustrated by Musharavati/Hamouda (2011) it is difficult to find an optimum process plan by testing all feasible solutions, the author's own derivation of the size of the design search space is shown below.

The computational time taken for an exhaustive search of this problem is derived below:

$$t_{\text{eval}} = b_1 \cdot n_{\text{ops}}$$

$$\text{size}_{\text{optimisation}} = b_2 \cdot n \cdot n_{\text{ops}}!$$

$$n_{\text{ops}} = b_3 \cdot n$$

$$T_{\text{total}} = \text{size}_{\text{optimisation}} \cdot t_{\text{eval}}$$

So:

$$T_{\text{total}} = b_2 \cdot n_{\text{features}} \cdot (b_3 \cdot n)! \cdot b_1 \cdot b_3 \cdot n \quad (5.2)$$

$$T_{\text{total}} = b_1 \cdot b_3 \cdot b_2 \cdot n^2 \cdot (b_3 \cdot n)!$$

$$T_{\text{total}} = c_1 \cdot n^2 \cdot (b_3 \cdot n)! \quad (5.3)$$

Where:

T_{total} total time to run exhaustive search of the design search space

b_1 time to cost model a single operation

b_2 average operation combinations per elemental feature

b_3 average operations per elemental feature.

$c_1 = b_1 \cdot b_3 \cdot b_2$, i.e. a constant determined by manufacturing knowledge.

n Number of features in the component to be cost modelled.

n_{ops} Number of operations in the manufacturing route.

So:

Minimising the number of elemental features is key to ensuring a rapid optimisation as this will reduce the size of the design space with a relationship as shown above, the time increases with $n^2 \cdot (b_3 \cdot n)!$. Given this relationship it can be seen that increasing the number of features will hugely increase the run time of the optimisation. Thus some features which could be decomposed into simpler features are therefore better left in a non-simplified form. For example a slot is composed of multiple faces but this is an unnecessary increase in the number of features, indeed in this research even multiple similar features are treated as elemental features provided they share tool access directions and are of a similar type.

It is of course true that an exhaustive search cannot reasonably be conducted on any but the simplest components but given the high level of epistasis in this optimisation the total number of evaluations necessary to achieve a reasonable chance of convergence to the global optimum will grow with an increase in the total number of possibilities.

In addition to the reduction in the number of elemental features other methods of reducing computational time are:

- Filtering out or repairing infeasible solutions
- Superior optimisers requiring fewer objective function calls
- Use of an objective function with a more rapid evaluation (i.e. a simpler and/or more efficient cost model)

5.6.1 Features

Of particular interest in this research is the consideration of features. There is a fundamental difference in thinking between design and manufacturing engineers, and it is this difference which makes the process planning problem complex to solve.

Designers think of features in terms of solving fundamental engineering issues such as: the direction and control of working fluids, the transfer and control of stress, control of fatigue, and resistance to heat, chemicals and erosion.

Manufacturers think in terms of achieving geometry to within tolerance, usually by removing or forming material, or joining parts together by welding, riveting or bolting.

It is clear that the manufacturing process must be deduced by the methodology in order to maintain flexibility, and we do not wish to filter out any potential solutions and given that manufacturing features are created by each manufacturing operation. The question therefore is how to define features such that the manufacturing method need not be predetermined, and it is clear that a new type of feature must be defined since design features themselves would introduce unnecessary complexity. To demonstrate this we can take a set of n bosses with a hole in the centre such as in test case one (Section 1.8, Table 1.2). This feature will at the very least require a hole making operation and some shaping or machining. The two processes are fundamentally different and need not be conducted consecutively, in fact they are unlikely to be conducted consecutively because to do so would increase the cost due to machine/tool changes.

The research thus introduces the concept of an elemental feature, a feature which can be produced in one or more operations but need not be broken down into other, simpler, features in order to allow lower cost manufacturing sequences. When all elemental features of a design feature are combined the geometry of the combination is identical to that of the design feature.

Given the substantial level of research into feature recognition, it was decided that a complementary method would be created; hence feature based manufacturing optimisation using machine decision making. This will include a feature mapping from design to elementary features which could be replaced by feature recognition in future work. A discussion on features is presented later but a brief definition of the features to be used is shown in Section 5.3. Some feature recognition technologies recognise elemental features but some recognise design features.

5.7 MCOST Structure

Written in C#, a schematic for the whole of MCOST is shown in Figure 5.3. The interfaces of MCOST with: the Access KnowledgeBase, Excel component data, and its own internal DataStructures are illustrated.

As can be seen MCOST is structured as: preliminary work; two optimisation loops, one inside the other; and post-processing.

1. Preliminary work includes import of the KnowledgeBase and component data; conversion of design to elemental features and initialisation of the matrices for the first objective function evaluation.
2. The outer loop is tasked with selecting the manufacturing operations which comprise the complete process set, identifying possible machines and TADs. It also calculates scrap and rework rates and the operation precedence list and provides all this information to the inner loop for sequencing operation.
3. The inner loop sequences these complete process sets to find the minimum cost feasible MoM.
4. Post processing includes bringing together the final results of the optimisation, and preparing them for user interpretation. This includes showing the uncertainty with regards to times and costs of the manufacturing operations.

The reasoning behind the internal and external loops is that this structure allows continuity of optimisation. Given that not all complete process sets generated will have the same number of operations, and given that the number of operations for each feature may differ, if the complete process set is changed each time, it is challenging to preserve advantageous parts of the sequencing genome. It is possible that a good CPS will be disregarded through an unlucky set of sequences which fail to maximise on its potential but it is thought (though not tested) that the benefits of continuity exceed this detriment. Future work could be to create and test a methodology for preserving the aspects of the sequence which can be, whilst allowing the CPS to change every objective function evaluation.

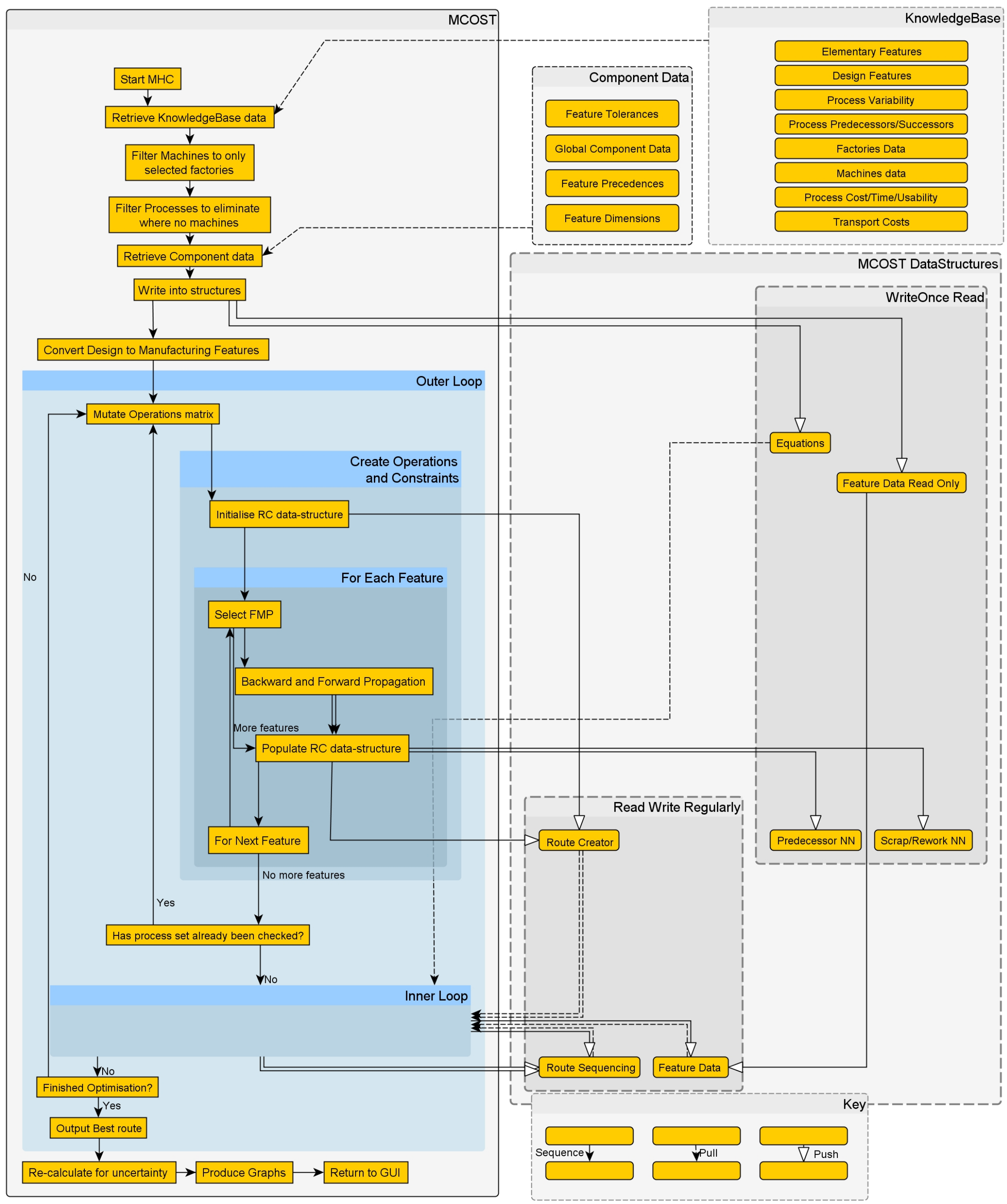


Figure 5.3: The architecture of MCOST, Inner loop architecture shown in Figure 5.4

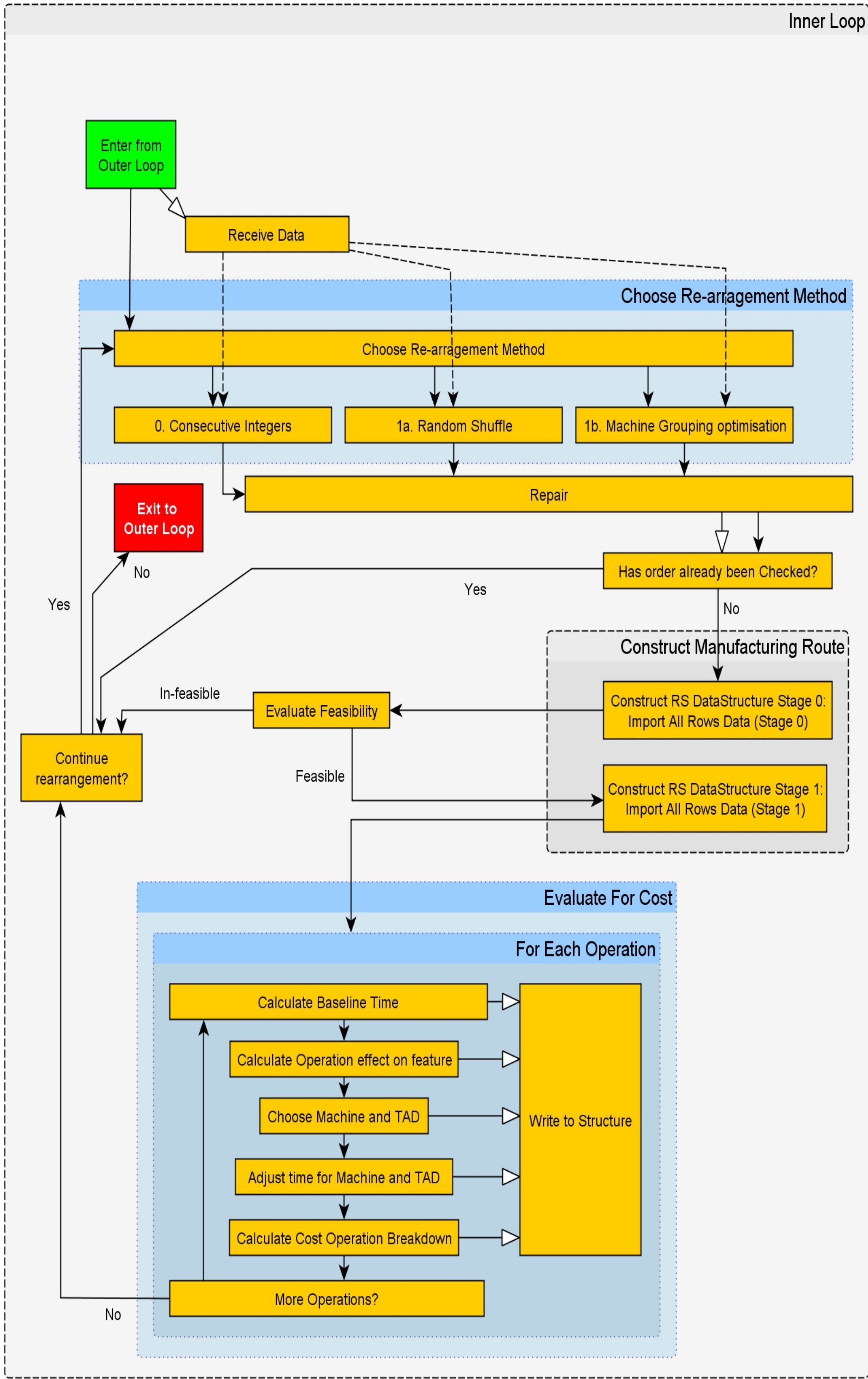


Figure 5.4: The inner loop of MCOST

5.7.1 CAD geometry (NX Data) and MCOST Interface (NX Data to Excel Spreadsheet)

Data from the NX component is exported to an Excel spreadsheet to interface with MCOST in order to allow the use of formulae when required. These data from NX are simply a list of named dimensions.

In Excel the data are converted to a list of design features and their associated G&T (geometric and tolerance) data. There is also data pertaining to the component such as material choice, production volumes and year of production which is relevant to cost but not stored in the NX model.

This Excel spreadsheet is manually created but automated methods of creation do exist, including feature recognition or design using standard features. It needs to be created only once per component template however; all components which can be made using that set of design features can then be constructed from the template.

In this template there are also some constraints which must be created. These are the constraints which exist between design features. For example a constraint may be that the holes in a set of bosses must be manufactured after the bosses themselves are created. The formation of these constraints would be expected to be part of a feature recognition algorithm, if this methodology were to be integrated with one.

5.7.2 KnowledgeBase (Access database)

MCOST relies on a database (Figure 5.1) previously described. In more detail this contains:

- Data on machines and factories such as size, costs, energy consumption, operations that the machine can complete
- Process predecessors and successors and criteria if any exist for that relationship
- Process set variability
- Process time/cost equations
- Design features, elemental features and their mappings

After the KnowledgeBase and NX data is imported the KnowledgeBase is filtered to keep only machines and process which are allowable (due to the year of production commencement and choice of suppliers).

5.7.3 Design to Elemental Feature conversion

A simple, one to many, user adaptable mapping is used for design to elemental feature conversion. A single design feature (DFC) is split into n elemental features, where precedences can be created between the different elemental features. Geometric and Tolerance data of the elemental features are directly mapped from the G&T data of the design feature.

An example of the design feature dimensions and their conversion to elemental features for test case one is shown in Figures 5.6, 5.7, 5.8. For many features the mapping values may be the same, if this is the case it means that the design feature and elemental features are identical. For others, generally more complex design features, the mapping values are not the same. The information in these tables is as follows:

D-Values Dimensional values, contain only one double which corresponds to a particular design dimension.

T-Values Tolerance values, contain only one double which corresponds to a particular design tolerance.

L-Values Dimensional values, contain a list of doubles which correspond to a particular design dimension. Usually used for values which change over time (such as remaining depth) where different manufacturing strategies will have differing values. For example a value in L0 of a multi stage groove (Figure 5.5) might be: CC,RF;1.06,5.81. This means that after centrifugal casting 1.06 mm remain for machining whist for Roll Forging 5.81 mm remain.

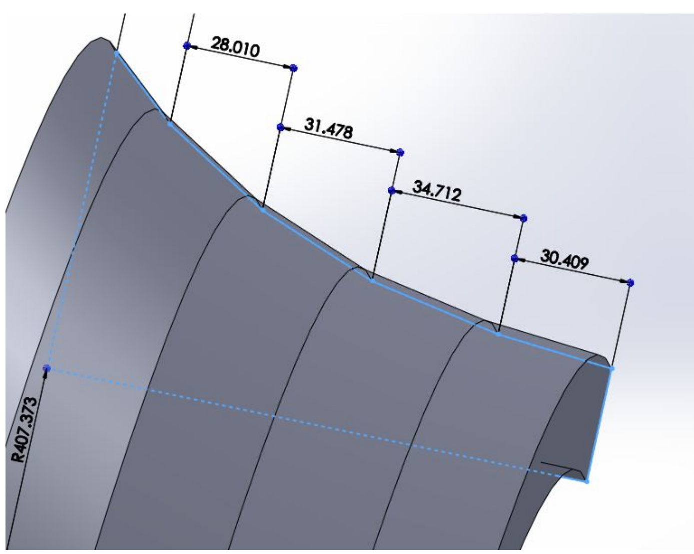


Figure 5.5: An example design feature: multi-stage groove

All these are defined for a particular design feature in the design feature table of the KnowledgeBase.

FeatureName	FeatureType	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D1
Rear Outer Casing	Condition of Supply - Cylindrical	180.00	490.00	65.00															
Pads	Pad	10.00	257.65	3.26	7.00	3198.35	45.97												
Weight Relief Slots	Slot Group Set	126.00	10.00	5.00	22.13														
Bolt and Dowel Holes	Through Holes Counterbored	126.00	10.00	0.50	8.00	10.00													
Guide Vane Bosses	Bushes	51.00	39.92	5.00	3254.58	37.02	20.90												
Guide Vane Holes	Through Holes Double Counterbored	51.00	10.00	12.00	15.00	11.90	4.00	2.00											
Mounting Flange	Flange	3119.60	3119.60	39.50	7.00	10.00	0.00	19.00	51.50	30.00									
Slots Inner	Multi-stage Groove	37.50	25.00	15.50	9.00	6.75	3361.50	3267.26	3204.42	3147.88	3122.74	30.00	30.00	30.00	15.00	3.58			
Slots Outer	Multi-stage Groove	23.00	35.50	43.00	47.50	50.50	3361.50	3267.26	3204.42	3147.88	3122.74	30.00	30.00	30.00	8.58	10.00			

(a) Design Features: D-Values (approximations)

FeatureName	FeatureParent	FeatureType	TAD	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14
Bolt and Dowel Holes:0	Bolt and Dowel Holes	Through Holes	Axial+	126	10	8											
Bolt and Dowel Holes:1	Bolt and Dowel Holes	Blind Holes	Axial+	126	0.5	10											
Guide Vane Holes:0	Guide Vane Holes	Through Holes	Radial+,Radial-	51	11.90113	10											
Guide Vane Holes:1	Guide Vane Holes	Blind Holes	Radial+,Radial-	51	4	12											
Guide Vane Holes:2	Guide Vane Holes	Blind Holes	Radial+,Radial-	51	2	15											
Mounting Flange:0	Mounting Flange	Slot Group Set Rotated	Radial-,Axial+,Radial+	1	3119.602	6	39.5										
Mounting Flange:1	Mounting Flange	Slot Group Set Rotated	Radial-,Axial+	1	3119.602	19	10										
Mounting Flange:2	Mounting Flange	Slot Group Set Rotated	Radial-	1	3119.602	51.5	30										
Mounting Flange:3	Mounting Flange	Slot Group Set Rotated	Radial+,Axial+	1	3119.602	6	30										
Pads:0	Pads	Slot Group Set Rotated	Radial-	1	3198.353	45.96625	7										
Pads:1	Pads	Slot Group Set	Radial-	10	257.6451	3.263764	7										
Rear Outer Casing:0	Rear Outer Casing	Condition of Supply - Cylindrical	Radial+,Radial-	180	490	65											
Slots Inner:0	Slots Inner	Slot Group Set Rotated	Radial+	1	3267.256	25	30	1	3204.425	15.5	30	1	3147.876	9	15	1	3122.743
Slots Inner:1	Slots Inner	Slot Group Set Rotated	Radial+,Axial-	1	3361.504	37.5	30										
Slots Outer:0	Slots Outer	Slot Group Set Rotated	Radial-	1	3267.256	35.5	30	1	3204.425	43	30	1	3147.876	47.5	8.579051	1	3122.743
Slots Outer:1	Slots Outer	Slot Group Set Rotated	Radial-	1	3361.504	23	30										
Weight Relief Slots:0	Weight Relief Slots	Slot Group Set	Radial-,Axial+	126	10	5	22.12624										

(b) Elemental Features, D-Values (approximations)

Figure 5.6: Test case one design and elemental features (D-Values)

FeatureName	FeatureType	T1	T2	T3	T4	T5	T6	T
Rear Outer Casing	Condition of Supply - Cylindrical	0.15	1.1	0.15				
Pads	Pad	0.15	0.15	0.15				
Weight Relief Slots	Slot Group Set	0.15	0.15	0.15				
Bolt and Dowel Holes	Through Holes Counterbored	0.15	0.15	0.15	0.15			
Guide Vane Bosses	Bushes	0.15	0.15	0.15				
Guide Vane Holes	Through Holes Double Counterbored	0.15	0.15	0.15	0.15	0.15	0.2	
Mounting Flange	Flange	0.05	0.15	0.15				
Slots Inner	Multi-stage Groove	0.05	0.2	0.2				
Slots Outer	Multi-stage Groove	0.3	0.2	0.2				

(a) Design Features: T-Values (approximations)

FeatureName	T1	T2	T3	T4	T5	T6	T7	T8	T9
Bolt and Dowel Holes:0	0.15	0.15							
Bolt and Dowel Holes:1	0.15	0.15	0.15						
Guide Vane Holes:0	0.15	0.15							
Guide Vane Holes:1	0.15	0.15	0.15						
Guide Vane Holes:2	0.15	0.15	0.2						
Mounting Flange:0	0.05	0.15	0.15						
Mounting Flange:1	0.05	0.15	0.15						
Mounting Flange:2	0.05	0.15	0.15						
Mounting Flange:3	0.05	0.15	0.15						
Pads:0	0.15	0.15	0.15						
Pads:1	0.15	0.15	0.15						
Rear Outer Casing:0	0.15	1.1	0.15						
Slots Inner:0	0.05	0.2	0.2						
Slots Inner:1	0.05	0.2	0.2						
Slots Outer:0	0.3	0.2	0.2						
Slots Outer:1	0.3	0.2	0.2						
Weight Relief Slots:0	0.15	0.15	0.15						

(b) Elemental Features, T-Values (approximations)

Figure 5.7: Test case one design and elemental features (T-Values)

FeatureName	FeatureType	L0	L1	L2	L3	L4
Rear Outer Casing	Condition of Supply - Tube	CC,FO;2000,1500				
Pads	Pad					
Weight Relief Slots	Slot Group Set					
Bolt and Dowel Holes	Holes Counterbored					
Guide Vane Bosses	Bushes					
Guide Vane Holes	Holes Double Counterbored					
Mounting Flange	Flange	CC,RF;3.023,29.2345	CC,RF;1.4,5.9545	CC,RF;1.4885,5.7845	CC,RF;1.327,5.1685	
Slots Inner	Multi-stage Groove	CC,RF;1.3315,6.0585	CC,RF;1.552,6.267	CC,RF;1.612,6.0325	CC,RF;1.4865,5.72	CC,RF;1.4785,5.576
Slots Outer	Multi-stage Groove	CC,RF;1.38,5.623	CC,RF;1.538,6.884	CC,RF;1.571,9.591	CC,RF;2.0965,13.5885	CC,RF;2.2075,18.0685

(a) Elemental Features, L-Values (approximations)

FeatureName	L0	L1	L2	L3	L4
Bolt and Dowel Holes:0					
Bolt and Dowel Holes:1					
Guide Vane Holes:0					
Guide Vane Holes:1					
Guide Vane Holes:2					
Mounting Flange:0	CC,RF;1.4359372789416,7.99383080496144				
Mounting Flange:1	CC,RF;1.45894540271951,6.43112057406942				
Mounting Flange:2	CC,RF;14.6215870069661,29.6437817369422				
Mounting Flange:3	CC,RF;0.942721472848015,6.34858928264372				
Pads:0					
Pads:1					
Rear Outer Casing:0	CC,RF;2000,1500				
Slots Inner:0	CC,RF;0.793858079539272,6.24513664501747	CC,RF;0.882651134566254,7.2946852026012	CC,RF;1.0510650462658,7.19535246976574	CC,RF;1.09947468053427,6.74039177928331	
Slots Inner:1	CC,RF;1.05539966062049,5.81266713522639	CC,RF;0.793858079539272,6.24513664501747	CC,RF;0.882651134566254,7.2946852026012	CC,RF;1.0510650462658,7.19535246976574	CC,RF;1.09947468053427,6.74039177928331
Slots Outer:0	CC,RF;6.89246989315684,7.67634343426388	CC,RF;5.83277086189493,9.83988948107859	CC,RF;4.80278918928522,14.2514724678256	CC,RF;4.55152087224173,19.1600489810252	
Slots Outer:1	CC,RF;3.68749940253449,7.00701219903208	CC,RF;6.89246989315684,7.67634343426388	CC,RF;5.83277086189493,9.83988948107859	CC,RF;4.80278918928522,14.2514724678256	CC,RF;4.55152087224173,19.1600489810252

(b) Elemental Features, L-Values (approximations)

Figure 5.8: Test case one design and elemental features (L-Values)

In addition to the dimensional mapping there needs to be a mapping of constraints. All constraints between DFC and other design features are copied to all elemental features and in addition new constraints between elemental features of DFC may be created according to KnowledgeBase rules.

5.7.4 Data-Structures Used (C# DataStructures)

Route Creator This structure stores the process set to be sequenced and evaluated constructed by the method listed in 5.8.1. It stores the operations to be used and the constraints which exist.

Route Sequencing This structure stores the process route after sequencing, the choices of machines and TAD and also the breakdown of costs.

Predecessor Neural Net (NN) There are two parts to this structure: the precedences part and the successors part. When constructing the process set the predecessors and successors for each operation on each feature are stored to reduce the calculation time in future when the same query is made.

Scrap/Rework Structure Similar in reason to above

Feature Structure This keeps track of the dimensions of each feature during the production process. Features can have dimensions independent of forming process (D values), tolerances (T values) and dimensions dependent on forming process (L values). The L value format was developed to ensure that consideration of more forming processes would not require re-coding of design features.

Equations This is to promote faster cost modelling as it stores co-efficients and indices in double and integer form thus reducing conversion times. There is also a Boolean array for if a cell is empty. This structure will be presented in more detail as it is fundamental to the speed of the optimisation and the method of data storage for the time and cost calculations.

5.8 MCOST Outer Loop

The MCOST outer optimisation loop is shown in Figure 5.3, the objective of the outer loop is to identify the optimal complete process set (CPS). It will be left to the inner loop to identify the optimal MoM.

Optimisation is conducted using a random mutation hill climber (RMHC) adjusting values which guide MCOST through the process set selection algorithm shown in Section 5.8.1, the optimisation of the random number matrix this algorithm uses is described in Section 5.8.3.

A Genetic Algorithm (GA) rather that a RMHC was initially used but the RMHC showed more rapid convergence, there may however have been better ways to code the GA than used and it is left to future work to improve further on the optimisation routine used.

5.8.1 Process set selection

For each elemental feature a set of processes is required. This process set should make the feature in the most cost effective way possible, however there is epistasis in this problem as how another feature y is made will determine the optimal way to make feature x. This is due to machine setup, transport between and inside factories and the effect of scrapping components. For example a variable process which could be optimal at the beginning of the process route when the accumulated cost of a component is relatively low may not be optimal later after substantial machining cost has been incurred.

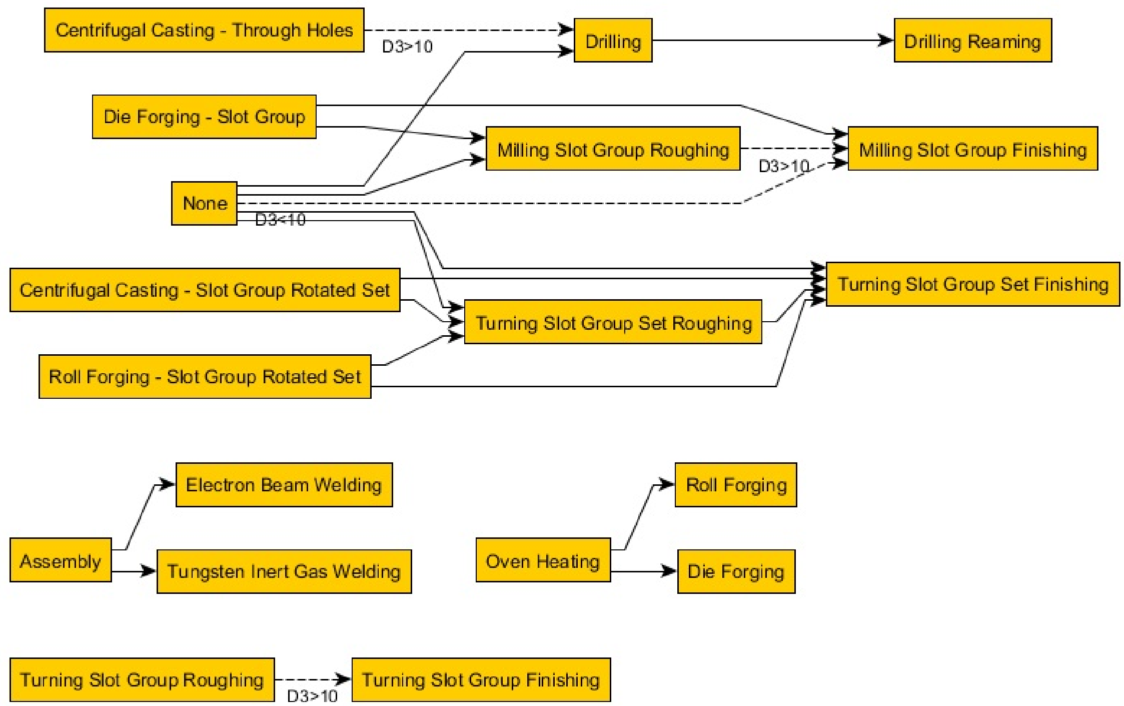


Figure 5.9: Predeccessors graph

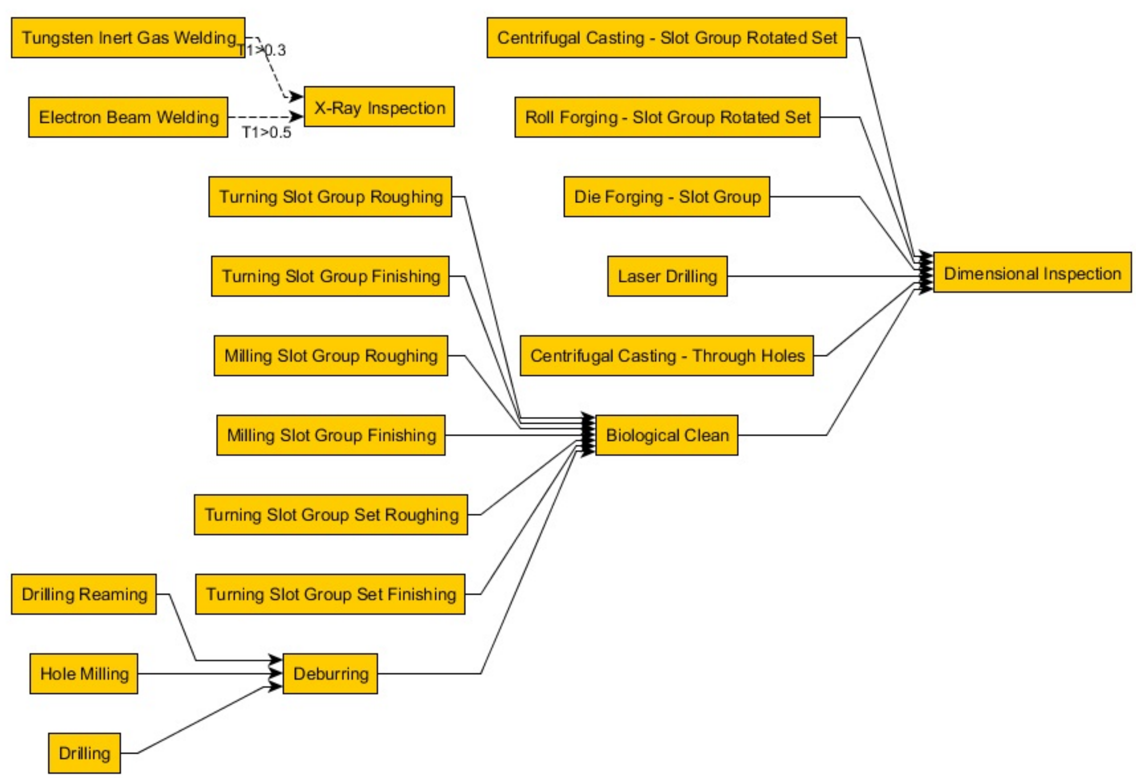


Figure 5.10: Successors graph

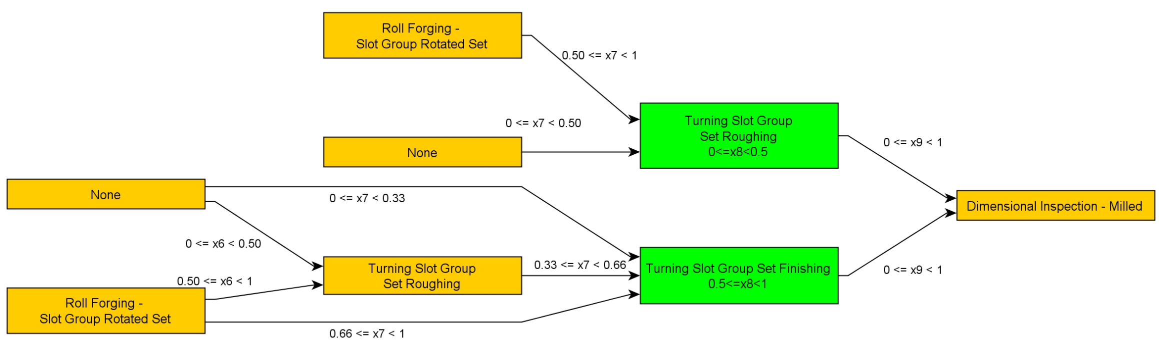


Figure 5.11: Backward and forward propagation options for an elemental feature: Slot Group Set Rotated

N.B. Because of the radius of the axis-symmetric groove to be formed, milling is presumed impractical if not impossible.

The concept of the FMP (Final Machining Process) is used here to mean a manufacturing process (usually chip forming but not necessarily) which creates a feature to within all defined tolerances. From the FMP we can use backward and forward propagation to construct a chain of processes which are conducted on this elemental feature. For this methodology it is also assumed that the FMP and any prior operations have an effect on dimensions and variability of the elemental feature and thus an effect on scrap and rework whereas subsequent operations do not. Numbers between zero and one are stored and controlled by the optimiser are used at each decision point, these numbers are stored in **X**, a matrix of size (n_{features},19) and are mutated randomly prior to process set selection. The cell of this matrix which refers to the choice of manufacturing strategy is not mutated. Where X is a row vector in matrix **X, t**he central value (X[8] in each row determines the FMP (green in Figure 5.11), values X[<8] control the decision making for predecessors whilst values X[>8] control the successors. Where a decision with one or more possibilities is encountered the choices are assigned equal selection bounds between zero and one. The value in X[n] is used to choose and the options are stored for future queries. Once a 'None' process is reached or there are no predecessors/successors to choose from any remaining values in row X remain unused. The requirement for the 'None' process stems from the optionality behind predecessors, they are not necessary, and their inclusion may have a beneficial/deleterious effect on the cost.

This process creates the Route Creator structure shown in Figure 5.12. The list of operations is the grid view to the left, to the right is a grid view showing the list of constraints which must be met.

The operation ID is used primarily for constraint checking. An example from the Figure 5.12 would be (Rear Outer Casting:0.7) It consists of three parts: A:B.C.

A The design feature from which the elemental feature is derived (*Rear Outer Casting* in this example)

B The number of the elemental feature from the mapping, zero based (*0* in this example)

C The operation number (*7* in this example), if the OpId is:

- **8** the operation is the FMP,
- **< 8** the operation is a predecessor (e.g 7 is the predecessor to the FMP, 6 is the predecessor to 7 if applicable etc.)
- **> 8** the operation is a successor (e.g 9 is the successor to the FMP, 10 is the successor to 9 if applicable etc.)

All constraints are listed in the DataTable on the right of this figure

Constraints between design features have a different feature name (A), they are created by the designer in the NX to MCOST interface file, an example of feature-feature constraints

RCFeature	RCFeatureType	RCOperation	RCOperationParent	RCOpId	RCMachine	RCTADs	ScProb	RwProb	MatLocation	Predecessors	Successors	ConstraintType
Rear Outer Casing:0	Condition of Supply - Cylindrical	Roll Forging	Roll Forging	Rear Outer Casing:0.8	Roll Forger;	Radial+;Radial-;	0	0	[11,8]	Rear Outer Casing:0.8	Bolt and Dowel Holes:0.7	Total
Rear Outer Casing:0	Condition of Supply - Cylindrical	Oven Heating	Roll Forging	Rear Outer Casing:0.7	Pre-Forging Oven;	Radial+;Radial-;	0	0	[11,7]	Mounting Flange:0.8	Bolt and Dowel Holes:0.7	Machining
Bolt and Dowel Holes:0	Through Holes	Drilling Reaming		Bolt and Dowel Holes:0.8	Drilling;Mill/Drill;	Axial+;	0	0	[0,8]	Mounting Flange:1.8	Bolt and Dowel Holes:0.7	Machining
Bolt and Dowel Holes:0	Through Holes	Drilling		Bolt and Dowel Holes:0.7	Mill/Drill;Drilling;	Axial+;	0	0	[0,7]	Rear Outer Casing:0.8	Bolt and Dowel Holes:1.7	Total
Bolt and Dowel Holes:0	Through Holes	Deburring		Bolt and Dowel Holes:0.9	Manual Deburring;	Axial+;	0	0	[0,9]	Mounting Flange:0.8	Bolt and Dowel Holes:1.7	Machining
Bolt and Dowel Holes:0	Through Holes	Biological Clean		Bolt and Dowel Holes:0.10	Dishwasher;	Axial+;	0	0	[0,10]	Mounting Flange:1.8	Bolt and Dowel Holes:1.7	Machining
Bolt and Dowel Holes:0	Through Holes	Dimensional Inspection		Bolt and Dowel Holes:0.11	Robotic Inspection;	Axial+;	0	0	[0,11]	Rear Outer Casing:0.8	Guide Vane Bosses:0.8	Total
Bolt and Dowel Holes:1	Blind Holes	Drilling Reaming		Bolt and Dowel Holes:1.8	Drilling;Mill/Drill;	Axial+;	0	0	[1,8]	Slots Outer:0.10	Guide Vane Bosses:0.8	Total
Bolt and Dowel Holes:1	Blind Holes	Drilling		Bolt and Dowel Holes:1.7	Mill/Drill;Drilling;	Axial+;	0	0	[1,7]	Slots Outer:1.10	Guide Vane Bosses:0.8	Total
Bolt and Dowel Holes:1	Blind Holes	Deburring		Bolt and Dowel Holes:1.9	Manual Deburring;	Axial+;	0	0	[1,9]	Rear Outer Casing:0.8	Guide Vane Bosses:1.8	Total
Bolt and Dowel Holes:1	Blind Holes	Biological Clean		Bolt and Dowel Holes:1.10	Dishwasher;	Axial+;	0	0	[1,10]	Slots Outer:0.10	Guide Vane Bosses:1.8	Total
Bolt and Dowel Holes:1	Blind Holes	Dimensional Inspection		Bolt and Dowel Holes:1.11	Robotic Inspection;	Axial+;	0	0	[1,11]	Slots Outer:1.10	Guide Vane Bosses:1.8	Total
Guide Vane Bosses:0	Slot Group Set Rotated	Turning Slot Group Set Roughing		Guide Vane Bosses:0.8	Turning Machine;	Radial-;	0.106	0.03	[2,8]	Rear Outer Casing:0.8	Guide Vane Holes:0.8	Total
Guide Vane Bosses:0	Slot Group Set Rotated	Biological Clean		Guide Vane Bosses:0.9	Dishwasher;	Radial-;	0	0	[2,9]	Guide Vane Bosses:0.8	Guide Vane Holes:0.8	Machining
Guide Vane Bosses:0	Slot Group Set Rotated	Dimensional Inspection		Guide Vane Bosses:0.10	Robotic Inspection;	Radial-;	0	0	[2,10]	Guide Vane Bosses:1.8	Guide Vane Holes:0.8	Machining
Guide Vane Bosses:1	Slot Group Set	Milling Slot Group Set Roughing		Guide Vane Bosses:1.8	Milling;Mill/Drill;	Radial-;	0.03	0.03	[3,8]	Slots Outer:0.10	Guide Vane Holes:0.8	Total
Guide Vane Bosses:1	Slot Group Set	Biological Clean		Guide Vane Bosses:1.9	Dishwasher;	Radial-;	0	0	[3,9]	Slots Outer:1.10	Guide Vane Holes:0.8	Total
Guide Vane Bosses:1	Slot Group Set	Dimensional Inspection		Guide Vane Bosses:1.10	Robotic Inspection;	Radial-;	0	0	[3,10]	Rear Outer Casing:0.8	Guide Vane Holes:1.8	Total
Guide Vane Holes:0	Through Holes	Drilling		Guide Vane Holes:0.8	Mill/Drill;Drilling;	Radial+;Radial-;	0.067	0.067	[4,8]	Guide Vane Bosses:0.8	Guide Vane Holes:1.8	Machining
Guide Vane Holes:0	Through Holes	Deburring		Guide Vane Holes:0.9	Manual Deburring;	Radial+;Radial-;	0	0	[4,9]	Guide Vane Bosses:1.8	Guide Vane Holes:1.8	Machining
Guide Vane Holes:0	Through Holes	Biological Clean		Guide Vane Holes:0.10	Dishwasher;	Radial+;Radial-;	0	0	[4,10]	Slots Outer:0.10	Guide Vane Holes:1.8	Total
Guide Vane Holes:0	Through Holes	Dimensional Inspection		Guide Vane Holes:0.11	Robotic Inspection;	Radial+;Radial-;	0	0	[4,11]	Slots Outer:1.10	Guide Vane Holes:1.8	Total
Guide Vane Holes:1	Blind Holes	Drilling		Guide Vane Holes:1.8	Mill/Drill;Drilling;	Radial+;Radial-;	0.067	0.067	[5,8]	Rear Outer Casing:0.8	Guide Vane Holes:2.8	Total
Guide Vane Holes:1	Blind Holes	Deburring		Guide Vane Holes:1.9	Manual Deburring;	Radial+;Radial-;	0	0	[5,9]	Guide Vane Bosses:0.8	Guide Vane Holes:2.8	Machining

Figure 5.12: A tabulated representation of the Route Creator Structure (RCS)

for test case one is shown in Figure 5.13, a solid line is binding on all operations associated with the feature. On the other hand a dashed line is binding only on operations up to and including the FMP, i.e. later operations such as clean up/inspection can be conducted later.

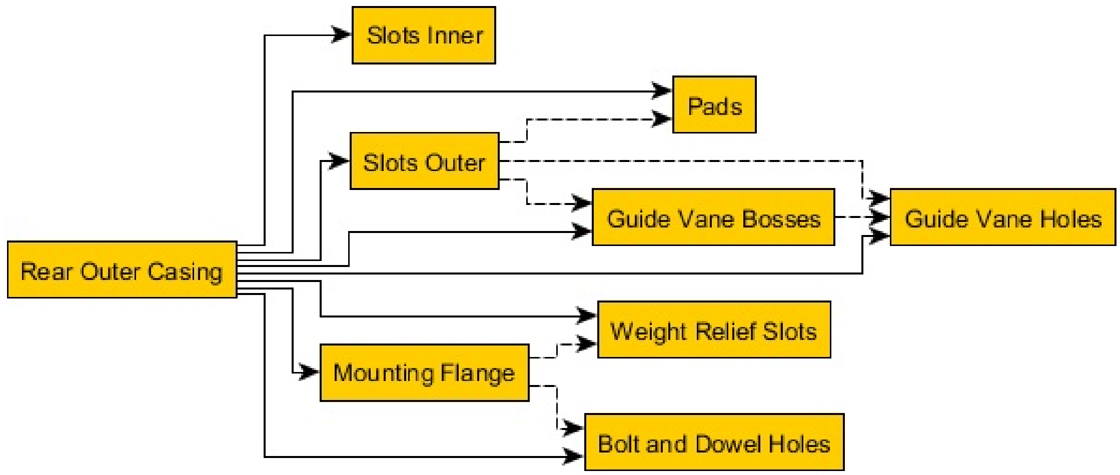


Figure 5.13: Feature-Feature constraints - Test case one

Constraints between different elemental features from the same design feature are defined by precedences created during the mapping from design to elemental features, they can be identified by the same A value but a different B value. This relationship is created in the KnowledgeBase.

Finally constraints are created by the backward and forward propagation process, these will have identical A and B values but different C values (e.g. *Rear Outer Casing:0.8* before *Rear Outer Casing:0.9*).

There is currently no such thing as an immediate constraint (i.e. two operations must be consecutive but this may be a useful addition to future work as this would limit the total number of possible feasible combinations.

5.8.2 Calculation of failure/rework probabilities

MCOST has data on the tolerance bounds allowed to various aspects of the feature and in addition has process variability data relevant to the process chain which operates on the feature. The probabilities of scrapping the part or reworking the feature are calculated using a multi-variant uncorrelated normal distribution, for which one, two and three dimensional examples are shown in 5.15, 5.16, 5.16, this method is capable of using more than 3 dimensions but it becomes difficult to visualise. Once the process set rework/scrap values are calculated they are stored for future use. An extract of the process set variability data is shown in 5.14.

It is assumed that for each defined tolerance on the feature, the process set chosen to make it has a known variability. For each defined tolerance of the elemental feature the probability of not scrapping the part is calculated. All of these not failing probabilities are then multiplied to obtain the overall probability of success. This is subtracted from one to give the scrap probability due to the creation of a single elemental feature. All this is then repeated for rework instead of scrap.

Using the method of calculating the integral of a normal distribution developed by Zelen/Severo (1964), for the calculation of rework and scrap probabilities we use the following logic:

$$P_{\mathrm{nso}} = \prod_{n=1}^{m} P_{\mathrm{ns},n} \tag{5.4}$$

$$P_{\mathrm{nro}} = \prod_{n=1}^{m} P_{\mathrm{nr},n} \tag{5.5}$$

$$P_{\mathrm{so}} = 1 - P_{\mathrm{nso}} \tag{5.6}$$

$$P_{\mathrm{ro}} = 1 - P_{\mathrm{nro}} \tag{5.7}$$

Where

$$Z = \left| \frac{\mathrm{SigmaDesignTolerance}}{\mathrm{SigmaMachiningVariability}} \right| \tag{5.8}$$

$$K = \frac{1}{1 + a_0 Z} \tag{5.9}$$

Operation6	Operation7	Operation8	T1sigma	T1offset	T2sigma	T2offset	T3sigma	T3offset	T4sigma	T4offset	T5sigma
	Milling Slot Group Set Finishing	Slot Group Set Polishing	0.002	0	0.03	0	0.03	0			
Casting - Slot Group Set	Milling Slot Group Set Finishing	Slot Group Set Polishing	0.002	0	0.03	0	0.03	0			
Die Forging - Slot Group Set	Milling Slot Group Set Finishing	Slot Group Set Polishing	0.002	0	0.03	0	0.03	0			
Milling Slot Group Set Roughing	Milling Slot Group Set Finishing	Slot Group Set Polishing	0.002	0	0.03	0	0.03	0			
Milling Slot Group Set Roughing	Milling Slot Group Set Finishing	Slot Group Set Polishing	0.002	0	0.03	0	0.03	0			
Milling Slot Group Set Roughing	Milling Slot Group Set Finishing	Slot Group Set Polishing	0.0021	0	0.03	0	0.03	0			
	Milling Slot Group Set Roughing	Slot Group Set Polishing	0.002	0	0.08	0	0.08	0			
Casting - Slot Group Set	Milling Slot Group Set Roughing	Slot Group Set Polishing	0.002	0	0.08	0	0.08	0			
Die Forging - Slot Group Set	Milling Slot Group Set Roughing	Slot Group Set Polishing	0.002	0	0.08	0	0.08	0			

Figure 5.14: A sample of process set variability data

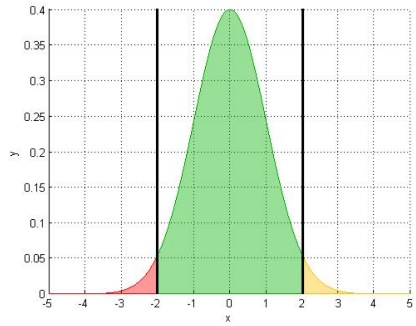


Figure 5.15: One dimensional normal distribution with scrap/rework (red/amber) regions shown

$$P_{\mathrm{nx},n} = 1 - \left(\frac{1}{\sqrt{2\pi}}\right) e^{\left(-\frac{z^2}{2}\right)} (a_1 K + a_2 K^2 + a_3 K^3 + a_4 K^4 + a_5 K^5); \tag{5.10}$$

Where

$$\begin{matrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{matrix} = \begin{matrix} 0.2316419 \\ 0.31938153 \\ -0.356563782 \\ 1.781477937 \\ -1.821255978 \\ 1.330274429 \end{matrix}$$

It is appreciated that this will double count the corners of the graphs as where a part if scrapped in one dimension it will clearly not be reworked in another but for reasonably capable processes it is thought that the effect of this slight over-calculation of rework is trivial and in any case this section of the code is modular and stores the results, thus negating the need for a complete recalculation every iteration of the outer loop, this allows the potential for more complex and expensive methods of correlated multi-variant normal distributions (MVNDs), for example.

It is also noted that work was conducted concurrently at the University of Southampton (Dodd, 2015) to calculate scrap and rework probabilities of correlated MVNDs and there is the potential at a later stage to implement this work within MCOST.

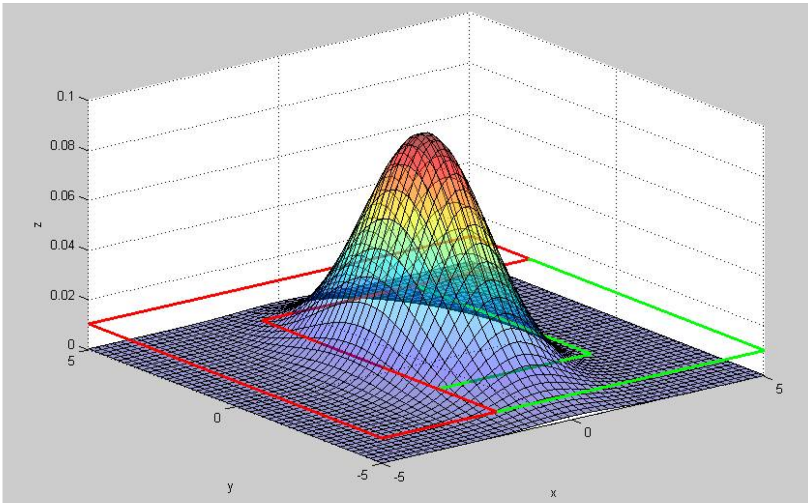


Figure 5.16: Two dimensional normal distribution with scrap/rework (red/green) regions shown

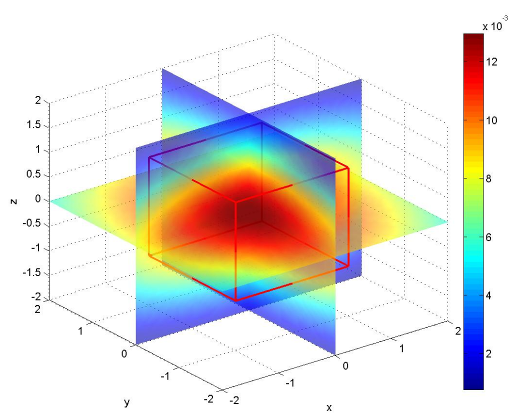


Figure 5.17: Three dimensional normal distribution with the acceptable region shown (the red cube)

Algorithm 3 Psuedocode showing the running of the outer optimisation loop

```
For i1 = 0:Rows-1
  For i2 = 0:Columns-1
    If rand(0,1) > threshold
      X[i1,i2] = rand(0,1)
    End If
  Next
Next
Construct Route Creator Structure and check if this complete sequence has already been tested. If not
  BILC, Xnew = RunInnerLoop()
End If
If (BILC < BGC)
  Keep Xnew
Else
  Revert to previous X
Repeat
```

5.8.3 Outer loop optimisation

Optimisation of the outer loop is accomplished by RMHC (Random Mutation Hill Climber) this is a variant of a GA with a population size of one, hence no crossover is possible. An MHC is analogous to the evolution of forms of life which reproduce asexually, such as microbes and many plants. However, in a typical RMHC the fitness measure will never decrease with increasing generation count as the better of the new or original individual is retained.

Inner loop optimisation was described in section 5.9.3. Outer loop optimisation simply changes the numbers in matrix **X** (the route construction matrix), this can result in a change in the choice of manufacturing processes used for the elemental features. The probability of an individual gene being mutated is specified by the user.

This mutation may not result in a change in the manufacturing route if all of the mutated numbers in **X** have no effect, either because: they are unused, there is only one option at this decision point, there is more than one option but the new mutated value refers to the same choice as the previous value.

Pseudocode for the algorithm is shown in algorithm 3:

Having mutated the process choice matrix and run the inner loop optimisation the Best Inner Loop Cost (BILC) is compared against the best global cost (BGC) so far. If the BILC is less than the BGC then the mutated matrix is retained, if not MCOST reverts to the prior matrix and the outer loop begins again.

5.9 MCOST inner loop

A schematic for the inner loop of MCOST is shown in Figure 5.18, the objective of the inner loop is to identify the optimal order of manufacturing operations by optimising and cost modelling the possible methods of manufacture (MoM) available from the current complete process set. Clearly the number of possible MoM is equal to the factorial of the number of operations, although many of these will be infeasible, however computational time must still be invested to discover whether they are feasible or not.

5.9.1 Repair function

Given the number of constraints which exist, it is likely that a mutated individual, even one from a feasible parent, will be infeasible. To prevent excessive computation therefore after mutation, and during initial route construction, a repair function is executed on the chromosome.

1 Forming Method Grouping This shifts all operations related to the forming operation to the start of the process sequence, thus satisfying constraint one (see Section 5.9.4).

2 Constraint Swap This swaps the positions of any pairs of manufacturing operations which breach their individual precedence constraints (see Section 5.9.4).

5.9.2 Initial process sequence initialisation

The start of the process sequencing algorithm is conducted as described below.

To begin with the processes are simply placed in an order determined by how they were generated by the outer loop. Effectively this means the sort will be by elemental feature and within this FMP, predecessors in reverse order, then successors. Clearly this will only generate a feasible route where no predecessors exist to any of the FMPs, and all feature to feature precedences are fortuitously met.

The repair function is then run, this will find a feasible solution, if at least one exists, for the process set chosen. If no feasible solution is possible after the first instance of the repair function the algorithm reverts to the process set selection phase.

5.9.3 Process sequencing

The optimisation loop is then run. It is comprised of two more methodologies in an iterative loop to find the optimum sequence.

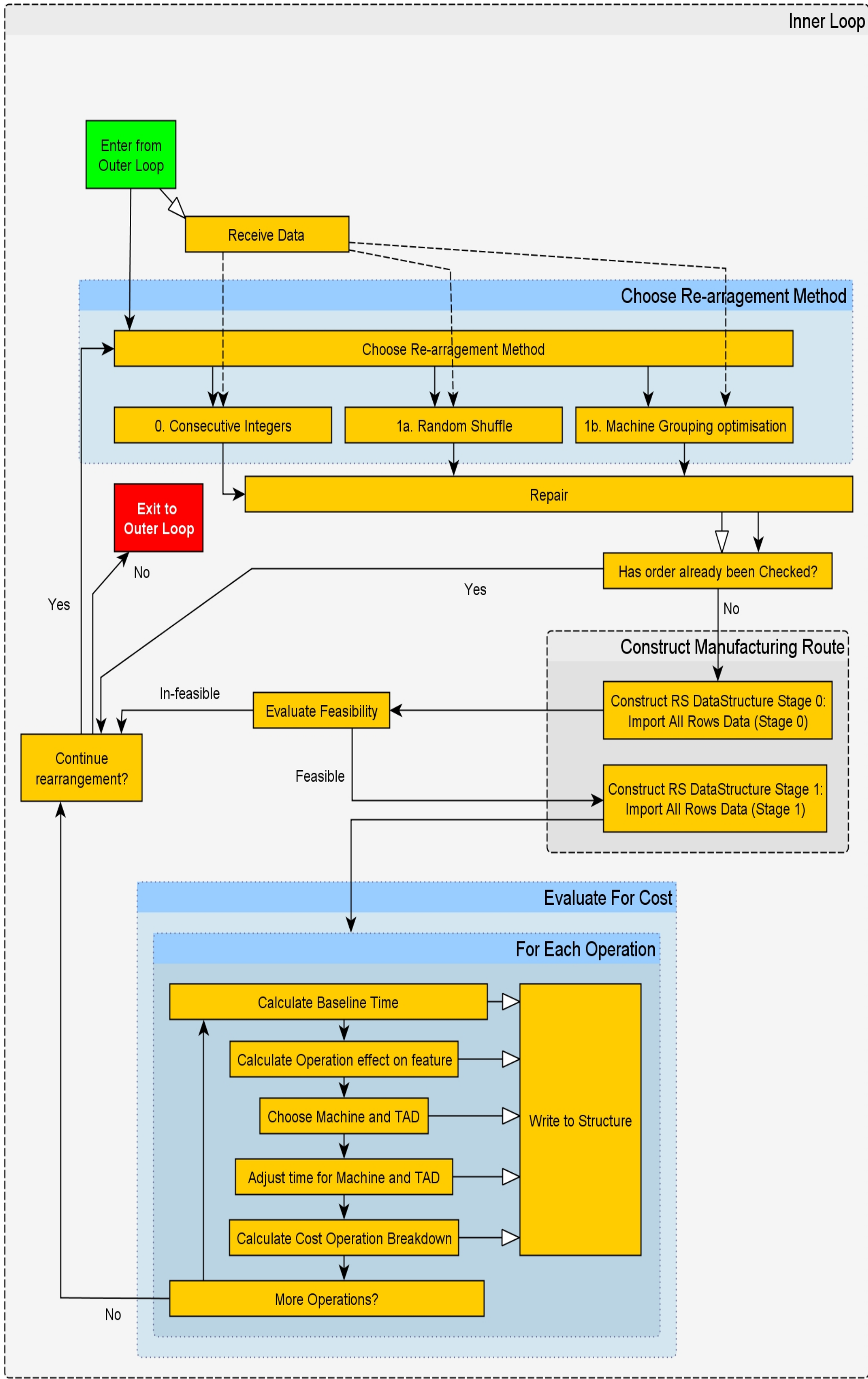


Figure 5.18: The inner loop of MCOST

1a Random re-sequencing This is a random rearrangement to attempt to reduce cost using random methods to break out of local optima, it works in a similar way to a playing card shuffle, by moving a section of the integer list from where it is to between two other values, for example 1,2,3,4,5,6,7,***8,9,10***,11,12 could become 1,2,3,***8,9,10***,4,5,6,7,11,12. The purpose of this algorithm is to attempt to keep beneficial groups together once they have been discovered.

1b Machine This is a local search based on the assumption that grouping operations where the same machine is required will have the effect of minimising setups and transport and thus the costs associated with them.

1c Machine and TAD Grouping Similar to 1b this groups operations where the same machine and TAD are required, it is expected to have the same effect as 1b but with a further reduction in setup due to no change in component/machine orientation.

After each iteration of one of these operators the repair algorithm previously described is run. The output from the two rearrangement algorithms followed by repair is a sequence of consecutive integers between zero and the number of operations:

After each rearrangement there is a multi-stage process test feasibility and cost, this is to maximise the computational efficiency of the algorithm. It begins with the RCS and a sequence of integers containing 0 to n and follows algorithm 4:

Algorithm 4 Feasibility and cost modelling of potential MoM

```
If sequence has not already been evaluated:
  Import only the OperationID Column (Figure 5.19).
  If the MoM is feasible (all constraints are satisfied)
    Import all other non calculated columns (shown in Figure 5.19).
    Calculate baseline time for all operations.
    Choose Machines and TADs using DTMS (Section 5.9.10).
    Calculate other times and all costs (creates Figure 5.21).
  End If
```

In this way the most computationally expensive parts of the algorithm (particularly the cost/time modelling but to a lesser extent the feasibility check) are conducted only if the MoM has not already been checked.

5.9.4 Feasibility Analysis

Constraint analysis: A feasible manufacturing route will have a fitness value of zero. Fitness is calculated by querying whether the forming process operations occupy the initial positions of the operation sequence, if they do not 15 is subtracted from the total. Secondly all predecessor constraints (Figure 5.12) are checked and one is subtracted from the fitness for each constraint breached. If the fitness is equal to zero, does the algorithm

Feature	FeatureType	Process	ProcessParent	Machine	TADs	OperationID
Rear Outer Casing:0	Condition of Supply - Cylindrical	Oven Heating	Roll Forging	Pre-Forging Oven	Radial+,Radial-	Rear Outer Casing:0.7
Rear Outer Casing:0	Condition of Supply - Cylindrical	Roll Forging	Roll Forging	Roll Forger	Radial+,Radial-	Rear Outer Casing:0.8
Mounting Flange:0	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial-	Mounting Flange:0.7
Slots Inner:1	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial+,Axial-	Slots Inner:1.7
Slots Outer:1	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial-	Slots Outer:1.7
Mounting Flange:0	Slot Group Set Rotated	Turning Slot Group Set Roughing		Turning Machine	Radial-	Mounting Flange:0.8
Mounting Flange:1	Slot Group Set Rotated	Turning Slot Group Set Roughing		Turning Machine	Radial-,Axial+,Radial+	Mounting Flange:1.8
Bolt and Dowel Holes:0	Through Holes	Drilling		Mill/Drill,Drilling	Axial+	Bolt and Dowel Holes:0.7
Bolt and Dowel Holes:0	Through Holes	Drilling Reaming		Drilling,Mill/Drill	Axial+	Bolt and Dowel Holes:0.8
Bolt and Dowel Holes:0	Through Holes	Deburring		Manual Deburring	Axial+	Bolt and Dowel Holes:0.9
Bolt and Dowel Holes:1	Blind Holes	Drilling		Mill/Drill,Drilling	Axial+	Bolt and Dowel Holes:1.7
Bolt and Dowel Holes:0	Through Holes	Biological Clean		Dishwasher	Axial+	Bolt and Dowel Holes:0.10
Bolt and Dowel Holes:1	Blind Holes	Drilling Reaming		Drilling,Mill/Drill	Axial+	Bolt and Dowel Holes:1.8
Bolt and Dowel Holes:1	Blind Holes	Deburring		Manual Deburring	Axial+	Bolt and Dowel Holes:1.9
Bolt and Dowel Holes:1	Blind Holes	Biological Clean		Dishwasher	Axial+	Bolt and Dowel Holes:1.10
Slots Outer:0	Slot Group Set Rotated	Turning Slot Group Set Finishing		Turning Machine	Radial-	Slots Outer:0.8
Slots Outer:0	Slot Group Set Rotated	Biological Clean		Dishwasher	Radial-	Slots Outer:0.9
Slots Outer:0	Slot Group Set Rotated	Dimensional Inspection		Robotic Inspection	Radial-	Slots Outer:0.10
Slots Outer:1	Slot Group Set Rotated	Turning Slot Group Set Finishing		Turning Machine	Radial-	Slots Outer:1.8
Slots Outer:1	Slot Group Set Rotated	Biological Clean		Dishwasher	Radial-	Slots Outer:1.9
Slots Outer:1	Slot Group Set Rotated	Dimensional Inspection		Robotic Inspection	Radial-	Slots Outer:1.10
Guide Vane Bosses:0	Slot Group Set Rotated	Turning Slot Group Set Roughing		Turning Machine	Radial-	Guide Vane Bosses:0.7
Guide Vane Bosses:0	Slot Group Set Rotated	Turning Slot Group Set Finishing		Turning Machine	Radial-	Guide Vane Bosses:0.8
Guide Vane Bosses:1	Slot Group Set	Milling Slot Group Set Finishing		Milling,Mill/Drill	Radial-	Guide Vane Bosses:1.8
Guide Vane Bosses:1	Slot Group Set	Biological Clean		Dishwasher	Radial-	Guide Vane Bosses:1.9

Figure 5.19: A tabulated representation of the Route Sequencing Structure (RSS) before time modelling, DTMS and cost modelling

proceed to the time and cost modelling stages, otherwise it returns to re-sequencing. This step could be encompassed within the repair algorithm but remains separate as no significant advantage is derived from doing so and it remains a secondary check.

5.9.5 Equation Structure

M is a matrix specific to a manufacturing process for a defined feature type and cost category or time. The actual equations structure which stores all the matrices has multiple arrays for each **M** to allow faster computation:

- A Boolean array identifying whether a particular cell is empty
- A Boolean array identifying whether a particular cell contains a double
- A Double array only populated for values which can be stored as doubles
- An Integer array where reference to a specific dimension, tolerance or machine attribute is required.
- A String array for values which cannot be stored in other, faster formats
- A List array for easy reference to the encoded functions.
- A List array for L-Values

Available functions include sum, mean, difference, rounding and power functions, these are expandable only by modification of the C# source code.

For example a function might consist of the text *Fdiff,D1,L1:CC* which would read D1-L1:CC where:

1. D1 would read the D1 value of the elemental feature to which the operation relates
2. L1:CC would read the L1 value of the elemental feature corresponding to centrifugal casting.

The equations are taken from an access database, the reason being that this aids updating of the equations by non programmers, and network deployment. There are three feasible methods of calculation of costs and times, each with unique advantages and limitations:

Non-compiled code Using an interpreted language such as VBA, Matlab or even Excel spreadsheets would be possible, but slow, hindering the optimisation process particularly if calls to external programs are required.

Compiled equations within source code This is the fastest method, requiring hard coded equations within the base software. However, a new manufacturing process or modification to an existing process requires redeployment of the complete software, probably across a company's network. In addition this approach would need the users to be proficient coders which experience shows is a rare skill combination.

Hybrid Using hard coded equations but pulling in co-efficients from a centralised database limits flexibility but does not require software expertise to update the equations.

Compile at run time A language such as python could be used which compiles at first run time, combines the advantages of: speed at run time and flexibility but retains the disadvantage of requiring people of moderate programming skill to update, also raises potential deployment issues for large organisations. Probably the best option but would carry little advantage in terms of the proof of concept so is left as future work.

The hybrid approach is followed, flexibility is provided by allowing functions to be nested within the co-efficients of the equation. These functions are hard-coded within the main program and expansion of these in an industrial setting would cause disruption but should be infrequently.

5.9.6 Volume Modelling

Volume Modelling calculates the volume removed or added in a manufacturing process. It is calculated using an equation of the form:

$$\text{Volume} = f(D, L) = \sum_{m=0}^{4} \left(\prod_{n=0}^{7} (\mathbf{M}_{m,n}) \right) \tag{5.11}$$

The values four and seven have been found to be sufficient to allow enough flexibility in the equations.

Where $\mathbf{M}_{m,n}$ can be: a coefficient, a dimension or tolerance of the feature (D, L, or T), a dimension or tolerance of the component (G), a characteristic of the machine (M), or a function taking inputs of the aforementioned types. The same equation is used to calculate the time and remaining cost categories (Sections 5.9.7 5.9.11).

An example of **M** for the volume calculation for process *Milling Slot Group Finishing* is:

$$\mathbf{M} = \begin{pmatrix} \text{D1} & \text{D2} & \text{D3} & \text{D4} & & & & \\ \text{D5} & \text{D6} & \text{D7} & \text{D8} & & & & \\ \text{D9} & \text{D10} & \text{D11} & \text{D12} & & & & \\ \text{D13} & \text{D14} & \text{D15} & \text{D16} & & & & \\ \text{D17} & \text{D18} & \text{D19} & \text{D20} & & & & \end{pmatrix}$$

A blank entry in $\mathbf{M_{0,0}}$ ensures that the entire equation is assumed to be zero.

A blank entry in $\mathbf{M_{m,0}}$ ensures that the row m is assumed to be zero.

A blank entry anywhere else is treated as a one.

Thus this calculation is equivalent to:

$$\begin{gathered} (D_1 \times D_2 \times D_3 \times D_4 \times 1 \times 1 \times 1 \times 1) \\ + (D_5 \times D_6 \times D_7 \times D_8 \times 1 \times 1 \times 1 \times 1) \\ + (D_9 \times D_{10} \times D_{11} \times D_{12} \times 1 \times 1 \times 1 \times 1) \\ + (D_{13} \times D_{14} \times D_{15} \times D_{16} \times 1 \times 1 \times 1 \times 1) \\ + (D_{17} \times D_{18} \times D_{19} \times D_{20} \times 1 \times 1 \times 1 \times 1) \end{gathered}$$

5.9.7 Time Modelling

Time modelling calculates the baseline time of an operation. It is calculated using the same equation as the volume calculation in Section 5.9.6 but with the added option to reuse the calculated volume:

$$\text{Time} = f(D, G, L, M, v) = \sum_{m=0}^{4} \left(\prod_{n=0}^{7} (\mathbf{M}_{m,n}) \right) \tag{5.12}$$

Where $\mathbf{M}_{m,n}$ can be: a coefficient, a dimension or tolerance of the feature (D, L, or T), a dimension or tolerance of the component (G), a characteristic of the machine (M), the added or removed volume, or a function taking inputs of the aforementioned types. The same equation is used to calculate the remaining cost categories (5.9.11).

An example of $\mathbf{M}$ for the time calculation for process *Milling Slot Group Finishing* is:

$$\text{Time} = \text{TimePerCubicMM} \cdot v$$

Where:

- v is the added or removed volume (previously calculated)
- TimePerCubicMM is extracted from a lookup table in the KnowledgeBase and is itself a function of the hardness of the material and the type of manufacturing process.

5.9.8 Feature Modification

Conducted during the same calculation loop as the volume and baseline time calculations are feature modification actions.

If the process modifies the feature, for example reducing the depth of it if a forming/cutting operation is used, then this occurs at the same time as the time calculated. The state of all features after each stage of the manufacturing process is stored so when the cost modelling is conducted the change in features need not be recalculated. In addition if the feature affects the T.A.D.s available to other features then their parameters are modified at this point. This is done by identifying the modification (m) to the TAD and feature (x) for which the modification takes place. MCOST then searches through the operations list for all operations on x and modifies their allowed TADs to include m.

Finally it is noted that the cost of scrapping a component will only be known once a defect is discovered, thus during this stage of modelling the probability of failure is moved to the next operation on the feature IFF the current machine is incapable of discovering the defect. If there are no more operations on the feature an error is thrown. Further improvements could be to define a variable capability to inspect for defects i.e. all machines can detect defects at some pre-defined level of resolution.

5.9.9 Stored Time Values

Multiple times are derived from the baseline time and stored in a vector for the cost analysis later, they are:

Baseline time The time of the operation without considering the effect of tool access direction (TAD). Calculated using formula Equation 5.12 Section 5.9.7.

Operation time The time of the operation after considering the effect of TAD. Calculated using a lookup table for manufacturing operation and TAD and applying a factor to Baseline time.

Setup time The setup time of the machine, either a full setup, change in operation or change in TAD (all machines have values for these categories).

Total time Setup time + Operation time.

Rework time Rework probability · Total time.

It is assumed that the rework is not discovered until the component has been removed from the machine.

Time Inc Rework Total time + Rework time.

These time values are calculated during the cost modelling stage (after DTMS, Section 5.9.10) rather than the time modelling because they may depend on the choice of machine and TAD where a choice exists.

5.9.10 DTMS (Dynamic Tool access direction and Machine Selection)

DTMS is an iterative method designed to identify the optimal machine and TAD (Tool Access Direction) to use from the options available as shown in Figure 5.19.

It uses exhaustive search the possibilities within a segment of the process sequencing structure to minimise the cost of an operation, taking into account the previous operation and the possibilities for the next operations, it relies upon knowing the times of past, current and future operations and hence must be run after the time modelling. The search space of each iteration of DTMS is shown in Figure 5.20, and it is run for each manufacturing operation in the MoM. In essence each iteration of DTMS 6d constructs a six dimensional array considering all possible combinations of machine choice and TAD for the current operation and two future operations. The equation used to populate each cell of the array is shown in Figure 5.20, which takes into account the machine and TAD choice from operation $n-1$ to $n+2$ inclusive. It then searches the array to find the minimum cost combination and returns the optimal choice of machine and TAD for operation n. It was stated in Section 5.4 that a logic based approach such as this one would not find the optimal in a select group of test cases and this is correct, however the further into the successive operations this approach looks, the less likely this sub optimisation becomes, hence the trade-off becomes a choice between a better but more costly evaluation of the current method of manufacture, or allowing more evaluations to explore more methods of manufacture, assuming a given availability of computational resource. It is left to future work to determine the optimal size of this evaluation, but it will of course depend on the specific test case.

DTMS uses an amalgamated cost rate considering depreciation, labour and power for more rapid evaluation than if they were considered separately although this comes at the expense of some accuracy in costs such as material costs and other which could depend on machine choice.

5.9.11 Cost Modelling

$$\text{Cost} = f\left(D, G, L, T, M, v\right) = \sum_{m=0}^{x} \left(\prod_{n=0}^{7} \left(M_{m,n}\right) \right) \tag{5.13}$$

Cost modelling for each operation is run immediately after running DTMS for that operation. For many of the cost categories the same equation format as the time modelling (Section 5.9.7) is used but with the added option of using the time values (OpTime, setup time, total time). The costs are calculated using this method for the following categories:

- Labour
- Tooling

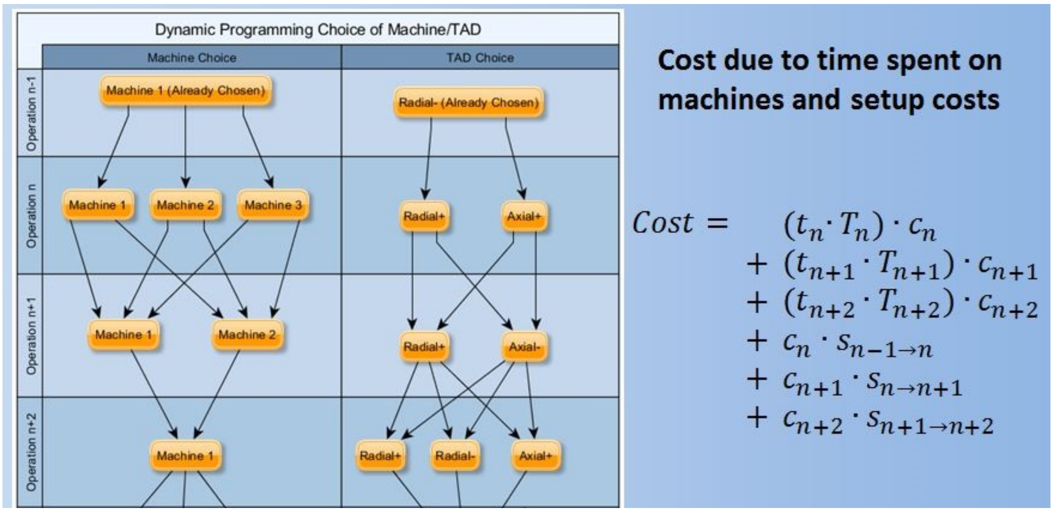


Figure 5.20: DTMS 6d

- Material
- Overheads
- Energy

For the following categories of cost an alternative method is used:

Building Building costs are calculated by multiplying a factory cost rate by machine area and machine time.

Equipment Equipment costs are the cost rate of the machine multiplied by the total time spent on the machine.

Failure Failure costs are the multiple of the failure probability and the current sum total of costs incurred up until that point in the production process.

Rework Rework is applied to the time value vector prior to cost modelling.

Transport Transport costs are determined by the formula if the machine used is not the same as that used for the last operation. This equation reflects the resource cost (due to mass) and risk incurred (due to value) during the movement of items.

$$C = vi + wm \tag{5.14}$$

Where:

v the current value

i is the insurance cost of moving components within or between factories per £ of current value (value being total invested cost up to that point, i.e. it will increase with every operation).

	CHENGDU	DERBY	HUCKNALL	KHI	SCAM
CHENGDU	0.01, 0.001	2, 0.05	2, 0.05	2, 0.05	2, 0.05
DERBY	2, 0.05	0.01, 0.001	0.5, 0.01	0.5, 0.01	0.5, 0.01
HUCKNALL	2, 0.05	0.5, 0.01	0.01, 0.001	0.5, 0.01	0.5, 0.01
KHI	2, 0.05	0.5, 0.01	0.5, 0.01	0.01, 0.001	0.5, 0.01
SCAM	2, 0.05	0.5, 0.01	0.5, 0.01	0.5, 0.01	0.01, 0.001

Table 5.2: Co-efficients of cost of transport between selected factory pairs - fictitious values only

w is the current mass

m is the resource cost of moving components within or between factories per kg

i and m are defined in the table below for selected factory pairs

It is true that i and m this equation will not necessarily be valid for certain (particularly urgent) transport requirements but it is a reasonable surrogate.

Feature	FeatureType	Process	ProcessParent	Machine	TADs	OperationID	FailureProbability	MTimePlusRework	LabourCost	ToolingCost	MaterialCost	OverheadsCost	EnergyCost	BuildingCost	EquipmentCost	FailureCost	TotalCost
Rear Outer Casing:0	Condition of Supply - Cylindrical	Oven Heating	Roll Forging	Pre-Forging Oven	Radial+	Rear Outer Casing:0.7	0	80	0	0	0	0	1200	0	80	0	1280
Rear Outer Casing:0	Condition of Supply - Cylindrical	Roll Forging	Roll Forging	Roll Forger	Radial-	Rear Outer Casing:0.8	0	160	53	2570	3460	0	0	1	479	0	7844
Mounting Flange:0	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial-	Mounting Flange:0.7	0	37	12	0	-399	0	0	0	111	0	7569
Mounting Flange:1	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial-	Mounting Flange:1.7	0	21	7	0	-225	0	0	0	63	0	7413
Slots Inner:1	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial+	Slots Inner:1.7	0	38	12	0	-406	0	0	0	113	0	7133
Slots Outer:0	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial-	Slots Outer:0.6	0	62	20	0	-667	0	0	0	186	0	6673
Slots Outer:1	Slot Group Set Rotated	Roll Forging - Slot Group Rotated Set	Roll Forging	Roll Forger	Radial-	Slots Outer:1.7	0	8	3	0	-84	0	0	0	24	0	6615
Mounting Flange:0	Slot Group Set Rotated	Turning Slot Group Set Finishing		Turning Machine	Radial-	Mounting Flange:0.8	0	145	48	0	0	0	0	1	145	0	6808
Mounting Flange:1	Slot Group Set Rotated	Turning Slot Group Set Roughing		Turning Machine	Radial-	Mounting Flange:1.8	0	12	4	1	0	1	0	0	11	0	6824
Bolt and Dowel Holes:0	Through Holes	Drilling		Mill/Drill	Axial+	Bolt and Dowel Holes:0.8	0	122	40	13	0	0	0	0	114	0	6991
Bolt and Dowel Holes:0	Through Holes	Deburring		Manual Deburring	Axial+	Bolt and Dowel Holes:0.9	0	50	16	20	0	0	0	0	50	0	7078
Bolt and Dowel Holes:0	Through Holes	Biological Clean		Dishwasher	Axial+	Bolt and Dowel Holes:0.10	0	30	10	3	0	0	0	0	30	0	7121
Bolt and Dowel Holes:1	Blind Holes	Drilling		Mill/Drill	Axial+	Bolt and Dowel Holes:1.8	0	36	12	1	0	0	0	0	34	0	7168
Bolt and Dowel Holes:1	Blind Holes	Deburring		Manual Deburring	Axial+	Bolt and Dowel Holes:1.9	0	50	16	25	0	0	0	0	50	0	7260

Figure 5.21: The RSS after time modelling, DTMS and cost modelling

5.9.12 Acceptance Criterion and elite tracking - Inner loop

Nomenclature

Value	Explanation
Current Value	The cost of the current evaluation of the objective function
Stored Value	The most recently accepted (by acceptance criterion) cost evaluation
Elite Value	The lowest cost evaluation for this inner loop

Table 5.3: Nomenclature for acceptance criterion - Inner loop

Generally speaking a RMHC will only move towards a new solution when it is superior to the current best solution. This means that if an area of low fitness must be traversed to reach a superior local optimum (or the global optimum), a fortunate mutation must be randomly selected. Simulated annealing is a technique which allows (with some probability) an inferior solution to be selected. As the optimisation progresses the temperature is lowered and lower fitness solutions are less likely to be accepted. This technique allows a more global search at the beginning, but becoming a very local search at the end of the optimisation. In this section the words fitness is measured in GBP and is equivalent to the cost of the component being modelled.

The simulated annealing test depends on the current baseline, i.e. the latest acceptable evaluation which this individual was mutated from, also known as the stored value. The acceptance criterion is shown below:

Accept if

$$e^{\left(-\frac{\triangle F}{T}\right)} > R$$

Where

R =A random number between zero and one

T =Simulated temperature

$\triangle F$ =Delta fitness, the fitness of the current evaluation minus the fitness of the stored value

The propability of accepting a new value, given a previous value and current simulated temperature is shown in Figure 5.22.

Elite tracking refers to the fact that when simulated annealing is used there is the possibility that the current solution can move away from the best to date solution (Figure 5.25), at a high temperature, it is then unable return to the best solution as the temperature drops. Elite tracking simply returns to the best solution so far in the inner loop (the Elite Value, Figure 5.26) thus allowing further improvement. The criterion to

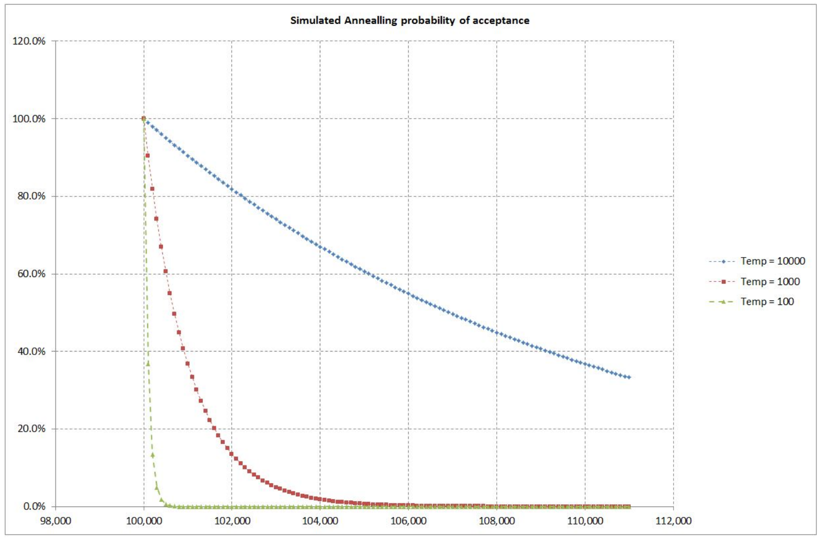


Figure 5.22: Simulated Annealling probability of acceptance of new solution - Stored value 100,000

return to the elite is simply a deviation ($\triangle F_E$) of three times the temperature T of the SA algorithm.

Where

$\triangle F_E$ =Delta fitness, the fitness of the current evaluation minus the fitness of the elite value

5.10 Calibration of MCOST KnowledgeBase

There are two key points determining the success or failure of MCOST as a methodology.

1. Is it finding the optimal manufacturing route or is it better than existing comparable methods.
2. Is the cost that it calculates for the optimal route correct or better than existing comparable methods.
3. Is it finding it as rapidly as possible or at least better than existing comparable methods.

Clearly if the cost calculations are incorrect MCOST's deductions for the optimal route is highly likely to be incorrect, and even if it is correct it will not be believed. Calibration of the time and cost equations is the most important aspect of the cost modelling. The most important aspects of the cost modelling are:

Time calculation times of processes

Equipment cost caused by depreciation and time spent on machine

Labour Cost Proportional to time and related to skill level of operator(s)

Bought in cost from suppliers, difficult to break down

Thus it can be seen that the biggest issue will be a miscalculation of the process times and because of this, this was the most concentrated on aspect. These equations were imported from the Rolls-Royce VCMS (Vanguard Cost Modelling System) lookup table shown in Figure 1.11, after inverting the values to give a time per unit volume value.

Initial verification was conducted by comparing the times extracted from MCOST for Test Case One: ROC against the SAP manufacturing data for the same part (Table 5.8).

5.11 Performance of the optimisation algorithm

The convergence graphs shown in this section all relate to Test Case One. Some graphs relate to using a RMHC for both the inner and outer optimisation loops. Later simulated annealing was implemented for the inner loop and the effects of this are shown.

5.11.1 Outer loop optimisation convergence graphs - Process set selection

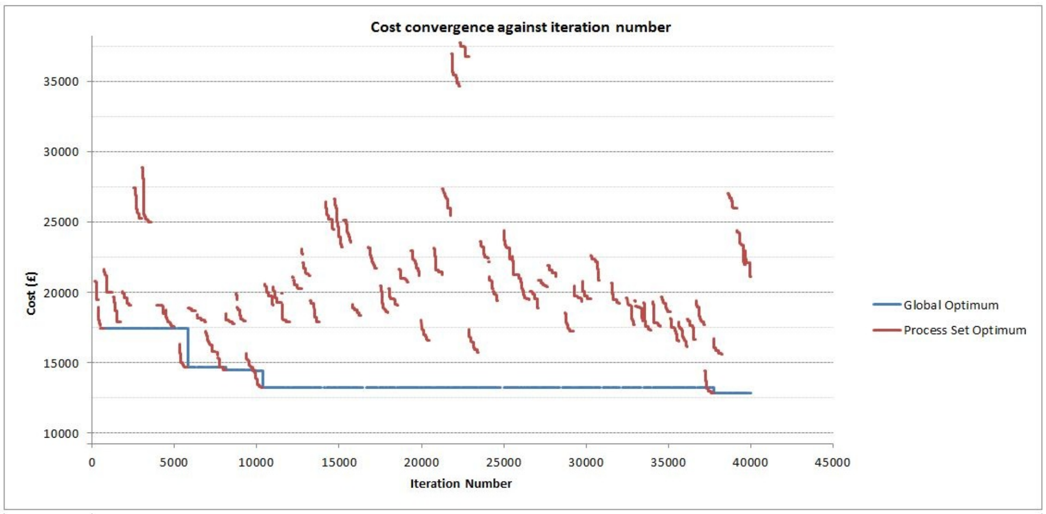


Figure 5.23: The entirety of the RMHC optimisation for centrifugal casting, each red curve is one inner loop

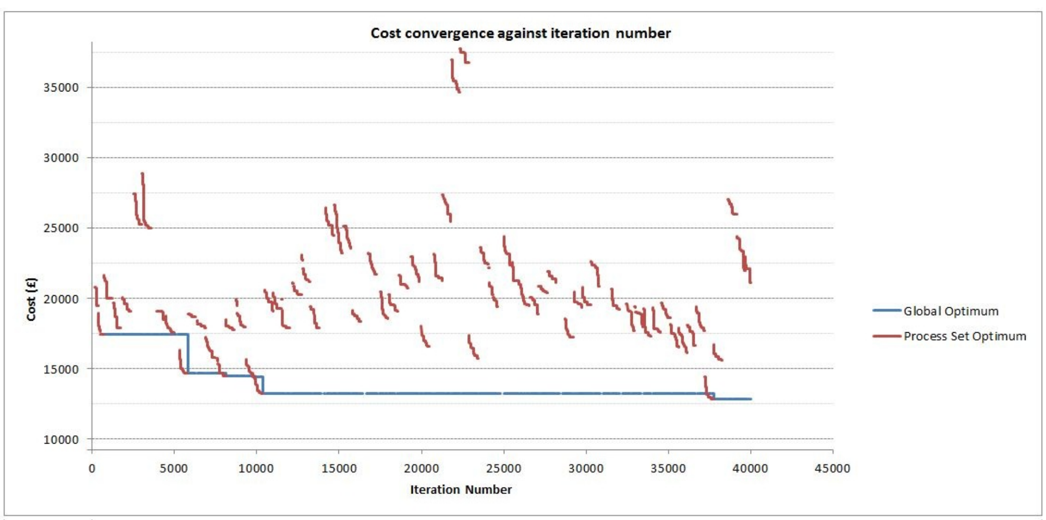


Figure 5.24: A portion of the total RMHC optimisation from objective function evaluation 0 to 45000

These graphs (Figure 5.23, Figure 5.24) show the convergence of the optimiser. Clearly visible are the global, inner loop best values and the current cost evaluation.

One of the biggest issues with optimisation of this complexity of design search space is identification of when or if the global optimum is reached, this is a common issue with optimisers exploring complex design search spaces with high levels of epistasis

5.11.2 Inner loop convergence graphs - Process sequencing

It is suggested in the literature review, that using simulated annealing in conjunction with an RMHC improves the convergence of the optimisation. The simulated annealing[4] algorithm has been implemented on the inner loop of the optimisation and the results are shown in Figure 5.25. As can be seen from the figure, simulated annealing can drift away from the optimum as improvement becomes more difficult in the later stages so a return-to-best algorithm called Elite Tracking (ET) was implemented and its effect is shown in Figure 5.26. If the algorithm detects a cost increase between current cost and elite cost of more than three[5] times current 'temperature', it returns to the elite value for further improvement. In tabular form the results for with and without ET are shown in Table 5.27, assuming all else remaining equal apart from the random number generation for inner loop mutation. The marginal benefit of implementing elite tracking is around one percent, which could make a substantial difference to a manufacturers profitability all else remaining equal.

[4]It should be noted that a pure RMHC is a specific case of SA (i.e. where the ambient temperature is zero).

[5]This value can be explored to find an optimal value or on what the optimal value depends for the most rapid convergence.

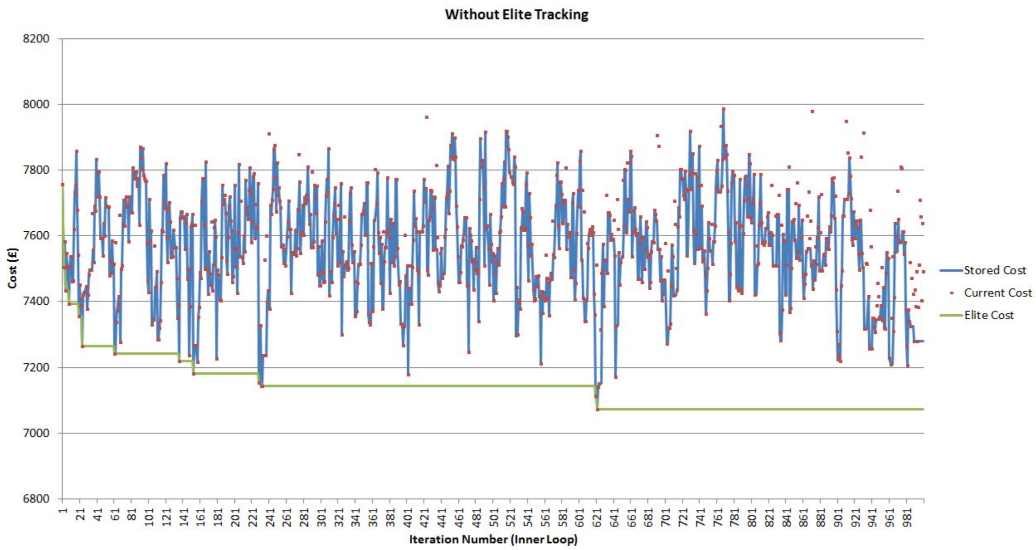


Figure 5.25: One inner optimisation loop with SA

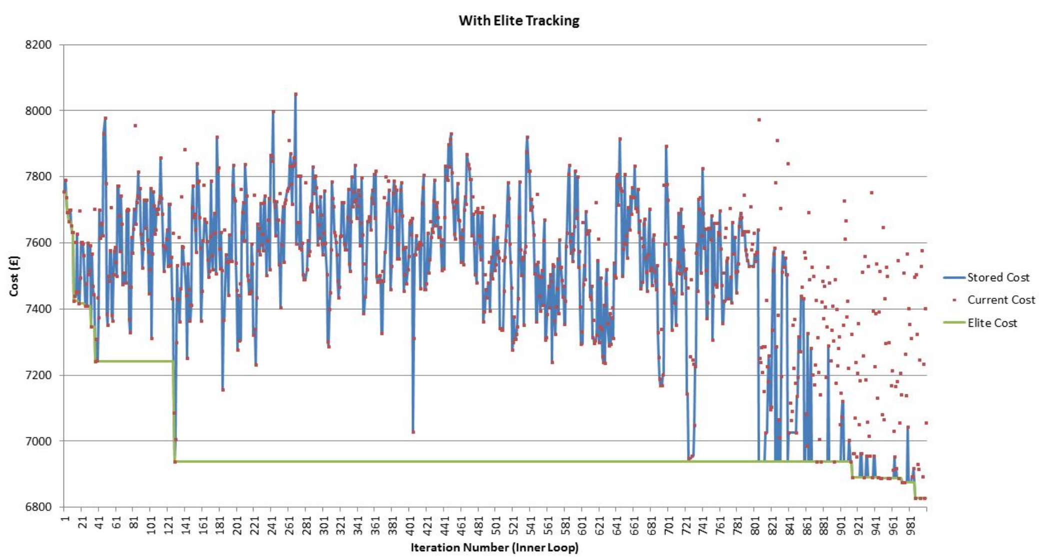


Figure 5.26: One inner optimisation loop with SA and ET

Optimisation parameters	Test Case one
	1000 generations
	50% chance of random sequence
	50% chance of machine clustering
	Temperature 1100 to 50

Inner loop start value (i.e. repair only)	Result of Inner Loop optimisation With Elite Tracking	Without Elite Tracking	Improvement by using elite tracking
£7,756	£6,829	£7,072	3.44%
£7,916	£7,227	£7,082	-2.05%
£8,510	£7,694	£7,727	0.43%
£8,446	£7,516	£7,573	0.75%
£8,518	£7,510	£7,706	2.54%
£8,676	£7,734	£7,866	1.68%
£8,924	£7,933	£8,056	1.54%
£10,836	£10,341	£10,482	1.35%
£11,256	£10,524	£10,587	0.60%
£11,256	£10,577	£10,621	0.41%
£10,485	£9,856	£9,871	0.15%
£10,524	£9,934	£9,983	0.49%
£10,640	£9,979	£10,119	1.39%
		Average improvement	0.98%

Optimisation parameters	Test Case zero
	1000 generations
	50% chance of random sequence
	50% chance of machine clustering
	Temperature 1100 to 50

Inner loop start value (i.e. repair only)	Result of Inner Loop optimisation With Elite Tracking	Without Elite Tracking	Improvement by using elite tracking
£6,144	£5,758	£5,803	0.78%
£5,975	£5,545	£5,594	0.87%
£5,975	£5,521	£5,577	1.01%
		Average improvement	0.89%

Figure 5.27: A tabular comparison of the same manufacturing processes optimised with or without elite tracking

5.12 Results

It is clear that given the type of optimiser used, assuming more than one possible permutation of the objective function, an infinite number of iterations (in both the inner and outer loops) is necessary to *guarantee* that the global solution will be found. However if the solution is desired in less than infinite time then an answer which is likely to be the global optimum, or is at least good enough, depending on the size of the design search space, must be accepted. It should be mentioned that only for test case zero was convergence actually reached.

Given this optimisation problem's type as a discrete rather than continuous optimisation problem, its quality can be determined by the frequency with which it can achieve the global optimum, if known, or how skewed the distribution of final results is. For example if the final result of exactly £8559 is achieved on 100 independent optimisation runs then it can reasonably be assumed that either that value is the global optimum.

If the optimisation is run with insufficient iterations to consistently find the optimal solution then a skewed normal distribution of final results (best result from the objective function) to be achieved, truncated at the global optimum.

Where iteration count is specified, it refers to the number of objective function evaluations (i.e. $G_0 \cdot G_1$ except in exceptional test cases with only small numbers of options). In these special cases, when the mutation operator failes to construct unique solutions repeatedly, it will assume no more unique solutions exist and moves to the next loop.

5.12.1 Test Case Zero

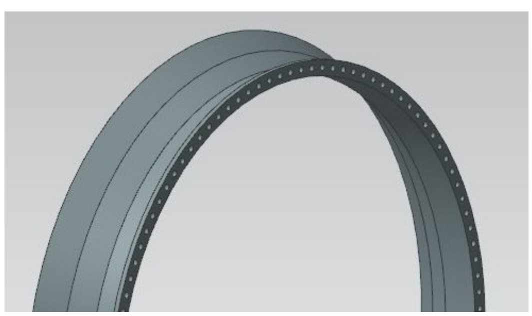


Figure 5.28: Test Case Zero: Simplified Rear Outer Casing (ROC)

The total optimisation MCOST described in this section is run multiple times with different (random) start points. Testing was conducted using differing numbers of generations of the optimiser to show improved convergence to the global optimum if more generations are allowed. By simplifying the geometry the feature set, G&T data and precedence list was contrived such that the global optimum was known to be £8559.

Each of the results generated by MCOST consists of the total cost of the manufacturing route, and a list of manufacturing operations with a complete cost breakdown for each operation, only the total cost is plotted in this graph (i.e. the lowest cost solution to the objective function). If large numbers of optimisations are conducted it is expected that a skewed normal distribution decay curve, truncated at the global optimum, will be shown if all end results are plotted as a histogram as in Figure 5.29. Clearly at higher iteration counts such at 10000 it is likely that all optimisation runs would find the global optimum of £8559.

Figure 5.29 shows the improvement towards more probability of finding the global optimum as the generation count is increased. If insufficient generations are used, the truncation at the global optimum becomes less evident, as is shown for the test case one results in Figure5.31.

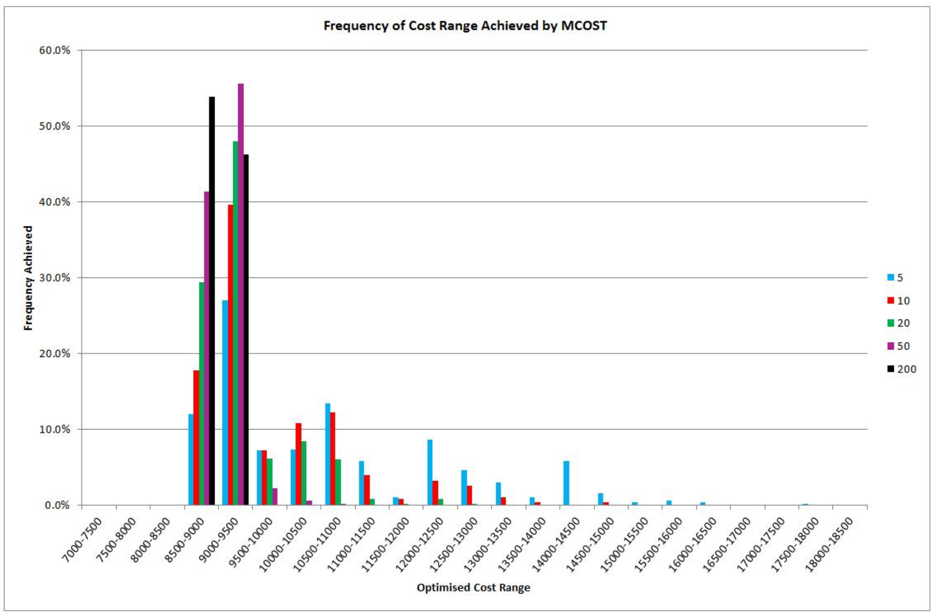


Figure 5.29: The spread of optimisation results varying with allowed generation count (Stock Machining only)

5.12.2 Test Case One

Figure 5.30: Test Case One: Trent 900 Rear Outer Casing (ROC)

A set of results for test case one is shown below. Three manufacturing strategies have so far been modelled. For this test case ring roll forging, centrifugal casting and machined from stock are shown.

The main output is a list of results from the optimisation (Table 5.4), each of these results contains a method of manufacture with a complete breakdown in costs, operation by operation, an abridged version is shown in figure 5.32 and a breakdown of time spent on machines for calibration with SAP data (Table 5.5). Table 5.4 can also be represented as a graph, as previously discussed in Section 5.12.1. Because insufficient time was allowed for convergence of the results the truncation of the Gaussian distribution (at the global optimum) is not evident. We can however, see that on average Roll Forging has the lowest cost, followed by machined from stock and finally casting is the most expensive production method for this component as specified.

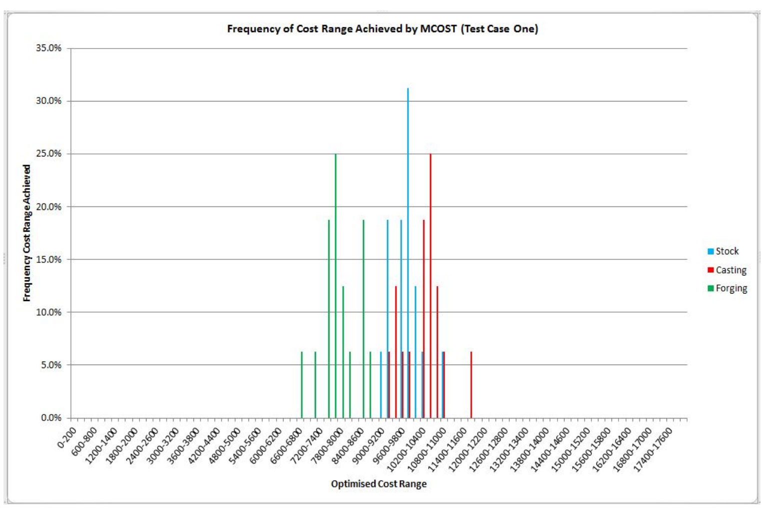


Figure 5.31: Test Case One Results: A graph showing the end result of 16 optimisation runs for each manufacturing strategy, raw data shown in Table 5.4

Cost	MS	Cost	MS	Cost	MS
6574	Ring Roll Forging	8957	Stock Tube	9290	Centrifugal Casting
7025	Ring Roll Forging	9138	Stock Tube	9422	Centrifugal Casting
7319	Ring Roll Forging	9210	Stock Tube	9428	Centrifugal Casting
7446	Ring Roll Forging	9254	Stock Tube	9691	Centrifugal Casting
7466	Ring Roll Forging	9574	Stock Tube	9708	Centrifugal Casting
7524	Ring Roll Forging	9592	Stock Tube	10210	Centrifugal Casting
7585	Ring Roll Forging	9672	Stock Tube	10212	Centrifugal Casting
7617	Ring Roll Forging	9785	Stock Tube	10291	Centrifugal Casting
7685	Ring Roll Forging	9794	Stock Tube	10391	Centrifugal Casting
7791	Ring Roll Forging	9822	Stock Tube	10398	Centrifugal Casting
7834	Ring Roll Forging	9859	Stock Tube	10399	Centrifugal Casting
7945	Ring Roll Forging	9897	Stock Tube	10421	Centrifugal Casting
8410	Ring Roll Forging	9921	Stock Tube	10513	Centrifugal Casting
8413	Ring Roll Forging	9937	Stock Tube	10664	Centrifugal Casting
8422	Ring Roll Forging	10118	Stock Tube	10870	Centrifugal Casting
8689	Ring Roll Forging	10782	Stock Tube	11531	Centrifugal Casting

Table 5.4: Set of cost estimates for test case one for each Manufacturing Strategy (MS)

Machine	TAD	Time
Pre-Forging Oven	Radial+	84
Hydraulic Press	Radial+	15
Roll Forger	Radial+	140
Roll Forger	Radial-	2
Roll Forger	Radial+	6
Roll Forger	Radial-	10
Mill/Drill/Turn	Radial-	582
Mill/Drill/Turn	Axial+	274
Mill/Drill/Turn	Radial-	82
Mill/Drill/Turn	Axial-	252
Manual Deburring	Radial+	45
Manual Deburring	Axial+	43
Laser Driller	Axial+	51
Laser Driller	Radial+	12
Biological Clean	Radial-	30
Robotic Inspection	Axial+	62
Robotic Inspection	Radial+	39
Robotic Inspection	Radial-	93

Table 5.5: Time breakdown by machine and TAD of the lowest cost MoM (for Ring Roll Forging) shown in Table 5.4

Machine	TAD	Time
Stock Supplier	Radial+	30
Mill/Drill/Turn	Radial-	416
Mill/Drill/Turn	Axial+	173
Mill/Drill/Turn	Radial-	323
Mill/Drill/Turn	Axial+	79
Mill/Drill/Turn	Radial-	30
Mill/Drill/Turn	Axial-	84
Mill/Drill/Turn	Radial+	22
Mill/Drill/Turn	Axial-	105
Mill/Drill/Turn	Radial+	41
Mill/Drill/Turn	Axial-	17
Mill/Drill/Turn	Radial+	205
Mill/Drill/Turn	Radial-	3
Manual Deburring	Radial+	58
Manual Deburring	Axial+	35
Laser Driller	Axial+	45
Biological Clean	Radial+	30
Biological Clean	Radial-	0
Biological Clean	Radial+	0
Biological Clean	Radial-	0
Biological Clean	Axial+	0
Biological Clean	Radial+	0
Robotic Inspection	Radial+	53
Robotic Inspection	Axial+	40
Robotic Inspection	Radial-	99

Table 5.6: Time breakdown by machine and TAD of the lowest cost MoM (for machined from stock) shown in Table 5.4

Machine	TAD	Time
Pre-Casting Oven	Radial+	91.81
Centrifugal Caster	Radial+	185
Centrifugal Caster	Radial-	0
Centrifugal Caster	Radial+	0
Centrifugal Caster	Radial-	0
Mill/Drill/Turn	Radial-	176.61
Mill/Drill/Turn	Axial-	9.92
Mill/Drill/Turn	Radial+	78.49
Mill/Drill/Turn	Radial-	251.27
Mill/Drill/Turn	Axial+	188.41
Mill/Drill/Turn	Radial-	138.07
Mill/Drill/Turn	Axial-	29.93
Mill/Drill/Turn	Radial-	2.93
Mill/Drill/Turn	Radial+	174.31
Mill/Drill/Turn	Axial+	71.57
Manual Deburring	Radial+	57.59
Manual Deburring	Axial+	35.1
Biological Clean	Radial+	30
Biological Clean	Radial-	0
Biological Clean	Radial+	0
Biological Clean	Radial-	0
Biological Clean	Radial+	0
Biological Clean	Radial-	0
Biological Clean	Axial+	0
Laser Driller	Axial+	44.82
Robotic Inspection	Radial-	75.17
Robotic Inspection	Axial+	11.3
Robotic Inspection	Radial-	27.21
Robotic Inspection	Radial+	16.35
Robotic Inspection	Axial+	58.72
Robotic Inspection	Radial+	15.96

Table 5.7: Time breakdown by machine and TAD of the lowest cost MoM (for Centrifugal Casting) shown in Table 5.4

Feature	Process	Machine	TADs	FailureProbability	ReworkProbability	BaseLineTime	TotalCostIncFailure	OpNumber
Rear Outer Casing:0	Oven Heating Casting	Pre-Casting Oven	Radial+	0	0	85.3	4673	0
Rear Outer Casing:0	Centrifugal Casting	Centrifugal Caster	Radial+	0	0	65	7246.6	1
Mounting Flange:1	Centrifugal Casting - Slot Group Rotated Set	Centrifugal Caster	Radial-	0	0	0	7207.1	2
Mounting Flange:2	Centrifugal Casting - Slot Group Rotated Set	Centrifugal Caster	Radial-	0	0	0	7082.5	3
Slots Inner:0	Centrifugal Casting - Slot Group Rotated Set	Centrifugal Caster	Radial+	0	0	0	6929.5	4
Slots Outer:0	Centrifugal Casting - Slot Group Rotated Set	Centrifugal Caster	Radial-	0	0	0	6614.2	5
Slots Outer:1	Centrifugal Casting - Slot Group Rotated Set	Centrifugal Caster	Radial-	0	0	0	6543.9	6
Mounting Flange:1	Turning Slot Group Set Finishing	Mill/Drill/Turn	Axial+	0	0	14.7	6729.4	7
Slots Outer:1	Turning Slot Group Set Roughing	Mill/Drill/Turn	Radial-	0.027	0.069	10.2	6921.2	8
Mounting Flange:2	Turning Slot Group Set Roughing	Mill/Drill/Turn	Radial-	0	0	37.4	6949.1	9
Slots Outer:0	Turning Slot Group Set Roughing	Mill/Drill/Turn	Radial-	0.027	0.069	41.2	7169.9	10
Pads:1	Milling Slot Group Set Finishing	Mill/Drill/Turn	Radial-	0.003	0.003	29.8	7229.1	11
Pads:0	Turning Slot Group Set Finishing	Mill/Drill/Turn	Radial-	0	0	165.7	7401.7	12
Pads:0	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Radial-	0.003	0.003	11.2	7481.3	13
Slots Inner:0	Turning Slot Group Set Roughing	Mill/Drill/Turn	Axial-	0	0	6.1	7486.5	14
Slots Inner:0	Turning Slot Group Set Finishing	Mill/Drill/Turn	Axial-	0	0	20.3	7503.2	15
Mounting Flange:2	Turning Slot Group Set Finishing	Mill/Drill/Turn	Radial-	0	0	7.5	7512.2	16
Mounting Flange:2	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Radial-	0.003	0.003	46.8	7774.4	17
Mounting Flange:1	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Radial-	0.003	0.003	31.2	7957.6	18
Guide Vane Holes:2	Drilling	Mill/Drill/Turn	Radial-	0	0	10.8	7971.3	19
Guide Vane Holes:2	Drilling Reaming	Mill/Drill/Turn	Radial-	0	0	7.2	7984.2	20
Guide Vane Holes:1	Drilling	Mill/Drill/Turn	Radial-	0	0	14.4	8002.3	21
Guide Vane Holes:0	Drilling	Mill/Drill/Turn	Radial-	0	0	42.8	8056.1	22
Guide Vane Holes:0	Drilling Reaming	Mill/Drill/Turn	Radial-	0	0	28.6	8107	23
Guide Vane Holes:1	Drilling Reaming	Mill/Drill/Turn	Radial-	0	0	9.6	8124.2	24
Mounting Flange:3	Turning Slot Group Set Roughing	Mill/Drill/Turn	Axial+	0	0	15.4	8133.9	25
Mounting Flange:3	Turning Slot Group Set Finishing	Mill/Drill/Turn	Axial+	0	0	7.5	8140.2	26
Mounting Flange:0	Turning Slot Group Set Finishing	Mill/Drill/Turn	Axial+	0	0	119	8237.8	27
Mounting Flange:0	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Axial+	0.003	0.003	61.6	8578.1	28
Mounting Flange:3	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Axial+	0.003	0.003	46.8	8843.6	29
Bolt and Dowel Holes:1	Drilling	Mill/Drill/Turn	Axial+	0	0	5.1	8850.1	30
Weight Relief Slots:0	Milling Slot Group Set Finishing	Mill/Drill/Turn	Axial+	0.003	0.003	140.9	9008.1	31
Bolt and Dowel Holes:1	Drilling Reaming	Mill/Drill/Turn	Axial+	0	0	3.4	9014.3	32
Slots Inner:0	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Axial-	0	0	126.3	9659.9	33
Slots Inner:1	Turning Slot Group Set Finishing	Mill/Drill/Turn	Axial-	0	0	608.9	10158.9	34
Slots Inner:1	Slot Group Set Rotated Polishing	Mill/Drill/Turn	Axial-	0	0	50.4	10416.3	35
Guide Vane Holes:1	Deburring	Manual Deburring	Radial+	0	0	15	10464.8	36
Bolt and Dowel Holes:0	Laser Drilling	Laser Driller	Axial+	0	0	20.8	10492.9	37
Bolt and Dowel Holes:1	Deburring	Manual Deburring	Axial+	0	0	42.7	10589.8	38
Guide Vane Holes:0	Deburring	Manual Deburring	Radial+	0	0	15	10617.3	39

Figure 5.32: Test case one: Rear Outer Casing, abridged lowest cost partial manufacturing route for centrifugal casting

5.12.3 Test Case One - Comparison with Rolls-Royce data

Results show that for test case one, the rear outer casing the delta cost between different manufacturing strategies is small, however the material properties which can be achieved by using ring roll forging are superior and the technology is also legacy, not requiring change from Rolls-Royce or their suppliers. Thus it is unsurprising that the technology is retained and the production route is unchanged in recent history. MCOST has been able to recommend changes on the manufacturing route such as laser drilling of the holes but these equations have not been confirmed as accurate by RR and would also require capital expenditure or further inter factory transport which would negate the gains.

SCAM NQF002241			1 NPT	CASING,REAR OUTER
	3005	3005FM07 LABOUR	0.083 H	INSPECT ON RECEIPT
	SCAM NQF001604		1 NPT	FORGING,CASE REAR OUTER
	3005	3005FM04 LABOUR	2.667 H	PROFILE TURN
	3005	3005FM04 LABOUR	4.167 H	PROFILE TURN
	3005	3005FM04 LABOUR	2.667 H	PROFILE TURN
	3005	3005FM04 LABOUR	4.533 H	PROFILE TURN
	3005	3005FM12 LABOUR	0.500 H	BIO CLEAN
	3005	3005FM07 LABOUR	1.367 H	INSPECT DIMENSIONAL
	3005	3005FM05 LABOUR	7.000 H	PROFILE MILL
	3005	3005FM12 LABOUR	0.500 H	BIO CLEAN
	3005	3005FM09 LABOUR	1.800 H	DEBURR
	3005	3005FM12 LABOUR	0.500 H	BIO CLEAN
	3005	3005FM07 LABOUR	1.983 H	FINAL INSPECTION
	ZB06	802002		All Parts Burden

Figure 5.33: Rolls-Royce record of manufacturing times (for test case one)

Machine	RR Time	RR Cost
Inspection	5	IP removed
Forging	-	IP removed
Turning	842	IP removed
Cleaning	30	IP removed
Dimensional Inspection	82	IP removed
Milling	420	IP removed
Biological Clean	30	IP removed
Deburring	108	IP removed
Biological Clean	30	IP removed2
Final Inspection	119	IP removed
Amortised Scrap Costs		IP removed

Table 5.8: T900 FBH SAP data, test case one, times combined when operations conducted on the same machine

Machine	TAD	MCOST time	MCOST time	RR Time	RR Cost	Delta
Pre-Forging Oven	r+	78	78	-		
Hydraulic Press	r+	15	15	-		
Roll Forger	r+	140		-		
Roll Forger	r-	4		-		
Roll Forger	r+	3		-		
Roll Forger	r-	9		-		
Roll Forger	r+	3	159	-	IP removed	
Lathe	a+	150				
Lathe	a-	20				
Lathe	r+	231				
Lathe	r-	428	829	842		-1.5%
Mill/Drill	r+	384				
Mill/Drill	r-	36	420	420		0%
Manual Deburring	r+	107.5	107.5	108		-0.5%
Biological Clean	r-	30				
Biological Clean	r+	0	30	30		0%
Robotic Inspection	r-	78				
Robotic Inspection	r+	46	124	119		+4.2%

Table 5.9: Comparison of MCOST estimates for machining time (including setup) and Rolls-Royce data (Figure 5.8) for test case one

5.12.4 Test Case Two

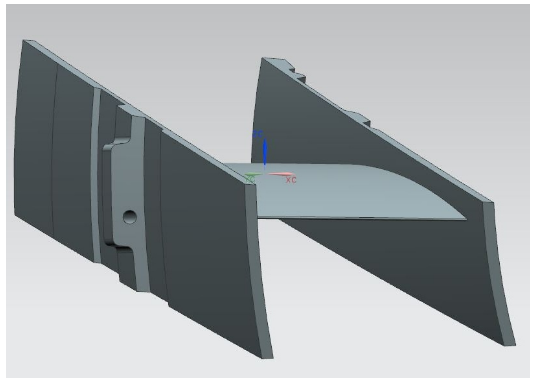


Figure 5.34: Test Case Two: Trent 700 Engine Section Stator (ESS)

Test case two (Figure 5.34) is usually die forged and machined to the finished shape. Figure 5.35 shows the a range of optimisation results using die forging as a manufacturing strategy while Figure 5.37 shows the machine and TAD breakdown for the lowest cost die forged method. Figures 5.38, 5.39 show an abridged breakdown of the data created by MCOST relating to Figure 5.37.

In addition the contrast between die forging and stock machining was explored. As can be seen, the best die forging method of manufacture is less costly due to the lower material removal volumes necessary, and lower input mass, although the cost of the input materials is higher. However the optimisation pertaining to stock machining is more consistent due to the smaller design search space it is exploring (there are fewer possible combinations).

The supplied cost for this component was £521 per vane in (2014). compared to MCOST's best prediction of £4212. The reason for the huge inaccuracy is that FoSCo 3d cannot currently represent (near) net shape manufacture (and indeed there is no point in it being able to do so). For this component therefore both the predicted forging cost is too high and the machining cost is also too high. On the aerodynamic surfaces only a polish is required, not milling operations as predicted by MCOST due to the incorrect FoSCo prediction.

Fitness	MinimumC	FormMethod	RelevantDS
0	4212	Die Forging	4
0	4357	Die Forging	1
0	4364	Die Forging	7
0	4609	Die Forging	5
0	4710	Die Forging	3
0	4777	Die Forging	6
0	4894	Die Forging	2
0	5034	Die Forging	0

Figure 5.35: Test Case Two: Range of optimisation results, all for Die Forging

Fitness	MinimumC	FormMethod
0	4227	Die Forging
0	4373	Die Forging
0	4608	Stock Billet
0	4737	Die Forging
0	4744	Stock Billet
0	4772	Stock Billet
0	4808	Stock Billet
0	4814	Die Forging

Figure 5.36: Test Case Two: Range of optimisation results including Die Forging and Stock Machining

Total Cost is: £4211.8		
Pre-Forging Oven	x+	41
Hydraulic Press	x+	15
Die Forger	x+	60
Die Forger	y+	0
Die Forger	y-	0
Die Forger	z-	0
Mill/Drill	z+	45
Mill/Drill	x+	168
Mill/Drill	x-	113
Mill/Drill	z-	14
Mill/Drill	y-	185
Mill/Drill	y+	222
Mill/Drill	x+	105
Mill/Drill	y+	236
Biological Cleaner	x+	30
Biological Cleaner	z+	0
Biological Cleaner	y+	0
Biological Cleaner	x+	0
Biological Cleaner	x-	0
Biological Cleaner	z-	0
Biological Cleaner	y-	0
Biological Cleaner	y+	0
Biological Cleaner	x-	0
Robotic Inspection	x+	32
Robotic Inspection	z-	5
Robotic Inspection	z+	5
Robotic Inspection	x-	7
Robotic Inspection	y-	8
Robotic Inspection	y+	8

Figure 5.37: Test Case Two: Optimal MoM from Figure 5.35

Feature	FeatureType	Process	ProcessParent	Machine	TADs	OperationID
ESS:0	Condition of Supply - Billet	Oven Heating Forging	Die Forging	Pre-Forging Oven	x+	ESS:0.6
ESS:0	Condition of Supply - Billet	Hydraulic Pressing	Die Forging	Hydraulic Press	x+	ESS:0.7
ESS:0	Condition of Supply - Billet	Die Forging	Die Forging	Die Forger	x+	ESS:0.8
Faces Inner:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	x+	Faces Inner:0.7
Faces Front:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	z-	Faces Front:0.6
Lower Vane Surface:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	y-	Lower Vane Surface:0.6
Faces Bottom:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	y-	Faces Bottom:0.6
Faces Back:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	z+	Faces Back:0.7
Faces Top:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	y+	Faces Top:0.7
Faces Outer:0	Slot Group Set	Die Forging - Slot Group Set	Die Forging	Die Forger	x-	Faces Outer:0.6
Faces Top:0	Slot Group Set	Milling Slot Group Set Roughing	All	Milling	y+	Faces Top:0.8
Faces Top:0	Slot Group Set	Biological Clean	All	Biological Cleaner	y+	Faces Top:0.9
Faces Top:0	Slot Group Set	Dimensional Inspection	All	Robotic Inspection	y+	Faces Top:0.10
Faces Front:0	Slot Group Set	Milling Slot Group Set Roughing	All	Mill/Drill	z-	Faces Front:0.7
Faces Front:0	Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	z-	Faces Front:0.8
Faces Back:0	Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	z+	Faces Back:0.8
Lower Vane Surface:0	Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	y-	Lower Vane Surface:0.7
Faces Inner:0	Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	x+	Faces Inner:0.8
Slots Inner:0	Tapered Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	x+	Slots Inner:0.7
Slots Inner:0	Tapered Slot Group Set	Slot Group Set Polishing	All	Mill/Drill	x+	Slots Inner:0.8
Faces Bottom:0	Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	y-	Faces Bottom:0.7
Lower Vane Surface:0	Slot Group Set	Slot Group Set Polishing	All	Mill/Drill	y-	Lower Vane Surface:0.8
Slots Outer:0	Tapered Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	x-	Slots Outer:0.7
Slots Outer:0	Tapered Slot Group Set	Slot Group Set Polishing	All	Mill/Drill	x-	Slots Outer:0.8
Faces Outer:0	Slot Group Set	Milling Slot Group Set Roughing	All	Mill/Drill	x-	Faces Outer:0.7
Faces Outer:0	Slot Group Set	Milling Slot Group Set Finishing	All	Mill/Drill	x-	Faces Outer:0.8
Upper Vane Surface:0	Slot Group Set	Milling Slot Group Set Roughing	All	Mill/Drill	y+	Upper Vane Surface:0.7
Faces Bottom:0	Slot Group Set	Slot Group Set Polishing	All	Mill/Drill	y-	Faces Bottom:0.8
Upper Vane Surface:0	Slot Group Set	Slot Group Set Polishing	All	Mill/Drill	y+	Upper Vane Surface:0.8
Faces Inner:0	Slot Group Set	Biological Clean	All	Biological Cleaner	x+	Faces Inner:0.9

Figure 5.38: Test Case Two: Abridged optimal MoM from Figure 5.37 (Columns 1-7)

FailureProbability	ReworkProbability	BaseLineTime	OpTime	SetupTime	ReworkTime	MTime	MTimePlusRework	LabourCost
0	0	21.1	21.1, 21.1, 21.1	20	0	41.1	41.11	24.7, 24.7, 24.7
0	0	10	10, 10, 10	5	0	15	15	9, 9, 9
0	0	0.5	0.5, 0.5, 0.5	60	0	60.5	60.5	0, 0, 0
0	0	0	0, 0, 0	0	0	0	0	0, 0, 0
0	0	0	0, 0, 0	0.1	0	0.1	0.1	0, 0, 0
0	0	0	0, 0, 0	0.1	0	0.1	0.1	0, 0, 0
0	0	0	0, 0, 0	0	0	0	0	0, 0, 0
0	0	0	0, 0, 0	0.1	0	0.1	0.1	0, 0, 0
0	0	0	0, 0, 0	0.1	0	0.1	0.1	0, 0, 0
0	0	0	0, 0, 0	0.1	0	0.1	0.1	0, 0, 0
0	0.292	5.1	5.1, 5.1, 5.1	30	10.24	35.1	45.31	27.2, 27.2, 27.2
0	0	0	0, 0, 0	30	0	30	30	18, 18, 18
0.292	0	1	1, 1, 1	30	0	31	31	18.6, 18.6, 18.6
0	0	7.3	7.3, 7.3, 7.3	30	0	37.3	37.33	22.4, 22.4, 22.4
0	0.023	1.5	1.5, 1.5, 1.5	0.1	0.04	1.6	1.63	1, 1, 1
0	0.023	159	159, 159, 159	5	3.77	164	167.76	100.7, 100.7, 100.7
0	0	129.9	129.9, 129.9, 129.9	5	0	134.9	134.89	80.9, 80.9, 80.9
0	0.023	47.4	47.4, 47.4, 47.4	5	1.21	52.4	53.64	32.2, 32.2, 32.2
0	0	96.5	96.5, 96.5, 96.5	0.1	0	96.6	96.61	58, 58, 58
0	0	3.2	3.2, 3.2, 3.2	0.1	0	3.3	3.33	2, 2, 2
0	0	118	118, 118, 118	5	0	123	123.05	73.8, 73.8, 73.8
0.001	0.001	25.7	25.7, 25.7, 25.7	0.1	0.03	25.8	25.8	15.5, 15.5, 15.5
0	0	157.7	157.7, 157.7, 157.7	5	0	162.7	162.67	97.6, 97.6, 97.6
0.001	0.001	3.1	3.1, 3.1, 3.1	0.1	0	3.2	3.22	1.9, 1.9, 1.9
0	0	50.6	50.6, 50.6, 50.6	0.1	0	50.7	50.66	30.4, 30.4, 30.4
0	0.023	50.7	50.7, 50.7, 50.7	0.1	1.17	50.8	51.94	31.2, 31.2, 31.2
0	0	217	217, 217, 217	5	0	222	221.96	133.2, 133.2, 133.2
0	0	3.9	3.9, 3.9, 3.9	5	0	8.9	8.9	5.3, 5.3, 5.3
0	0.2	25.7	25.7, 25.7, 25.7	5	6.14	30.7	36.81	22.1, 22.1, 22.1
0	0	0	0, 0, 0	30	0	30	30	18, 18, 18

Figure 5.39: Test Case Two: Abridged MoM from Figure 5.37 (Columns 8-16)

5.12.5 Minor Test Cases

MCOST was run on the minor test cases shown in Tables 1.4, 1.5 to demonstrate its flexibility. It did not fail to give a response but was unable to accurately cost these parts. This was due to the fact that the manufacturing equations used were based on Rolls-Royce's experience of manufacturing large components. Smaller, more commoditised components such as these minor test cases would be procured from suppliers who would have more experience in manufacturing smaller components and would be more experienced with optimal order quantities to minimise the cost of setup per component. This is an aspect of batch production and the suggestion to consider its inclusion is made in the conclusions.

6 Conclusions

The previous sections in this thesis deal with:

Background Information A review of academic literature and the current state of commercial technology, the source data obtained from the industrial sponsor used in this research.

Methodology Sections on the methodology and toolset created as a whole, then separate sections for the tool which creates the precursor shape estimate, and the tool for manufacturing optimisation.

6.1 Review of aims and objectives

The question to be investigated was:

> *Conducting manufacturing process selection prior to manufactuing sequencing in CAPP can lead to suboptimisation. Dynamic process selection, conducted within an optimisation will improve this, and thus allow superior decision making by component designers and manufacturing experts.*

It became apparent through the course of the research, that the optimal manufacturing processes depended to a significant extent on the precursor shape. Larger precursor shapes (such as those created by roll forging of high temperature materials) create a tendency to use processes with less smooth finishes and inferior accuracy, such as VIPER (Very Impressive Performance Extreme Removal) grinding, but faster material removal rates. These may then require more precise finishing operations with improved surface finish.

The choice of manufacturing processes was a completely different type of question than the shape which would be generated by the precursor technology. Thus the research was divided into two sections, and two distinct, though complementary, tools were created: the Forming Shape Creator (FoSCo), and the Manufacturing Choice Optimisation and Sequencing Tool (MCOST). The purpose of FoSCo was to deduce potential precursor shapes. MCOST was developed to choose the manufacturing operations and their sequence.

The tasks required to satisfy the objective were covered by the two tools.

Task 1 To deduce the shape created using additive or forming processes (by which it is meant casting, forging processes).
FoSCo Section 4.

Task 2 To determine what possible manufacturing processes can be used to make the remaining features (i.e. any not already created by the additive or forming process.

MCOST Section Y. Implicit within the FMP, Predecessors and Successors DataTables. Graphically shown in Section 5.8.1.

Task 3 To derive which possible manufacturing processes should be used to make the remaining features.
Determined by the MCOST outer loop optimiser, this is a random mutation hill climber which simply mutates the numbers in a matrix which determines the path followed through the predecessor's and successor's tree illustrated in Figures 5.9, 5.10, building on and expanding on the work of Zhou et al. (2007). This is explained in far more detail in Section 5.8.

Task 4 To deduce the optimal order in which these processes should be conducted.
Drawing on the work of many authors presented in Section 2.2. A random mutation hill climber augmented with simulated annealing and customised mutation functions is used to optimise the process sequence, Section 5.9.

Task 5 To make any remaining decisions such as machine, tooling used and access direction DTMS (Section 5.9.10) is used.
It relies on an exhaustive evaluation of all possibilities for n future operations within the cost modelling stage of the inner loop of optimisation, to iteratively calculate the optimal decisions (machine choice and TAD) for a given operation, taking into account setup cost, transport and operation cost.

Clearly tasks three, four and five which require optimiser made decisions all rely on the cost model generation and evaluation, in order to make their decision. Cost model construction and evaluation is shown in Section 5.9.11.

This thesis proves that automated deduction of the cost-optimal manufacturing process is possible and practical using dynamic process seletion, thus avoiding the suboptimisation present in existing methodologies.

It allows design engineers access to the effect on cost, of their decisions or changes. It can deduce sensible manufacturing routes when compared with expert knowledge, can respond appropriately to changes in geometry and tolerance data and achieve reasonable accuracy in cost modelling given its limited calibration data.

The following section details the specific conclusions and further work for FoSCo and MCOST separately.

6.2 FoSCo

It is known that the choice of manufacturing processes, and the time taken by these operations, depended in large part on the shape of the material which is created initially, for example by casting, forging, as supplied in the form of stock etc. Clearly the closer to the finished part geometry the initial shape is created, the less time must be spent removing extraneous material. Sometimes the shape is limited by the capabilities of the shaping process, or the economics of it, whilst sometimes it is constrained by other factors, such as the ability to check for defects using X-ray or ultrasonic technology. In any case, it is necessary to predict the form that this shape could take to estimate an accurate unit cost, and to determine which of these processes, e.g. casting, forging etc. to use, where more than one is feasible. An example of the shape generated for test case two is shown in Figure 6.1.

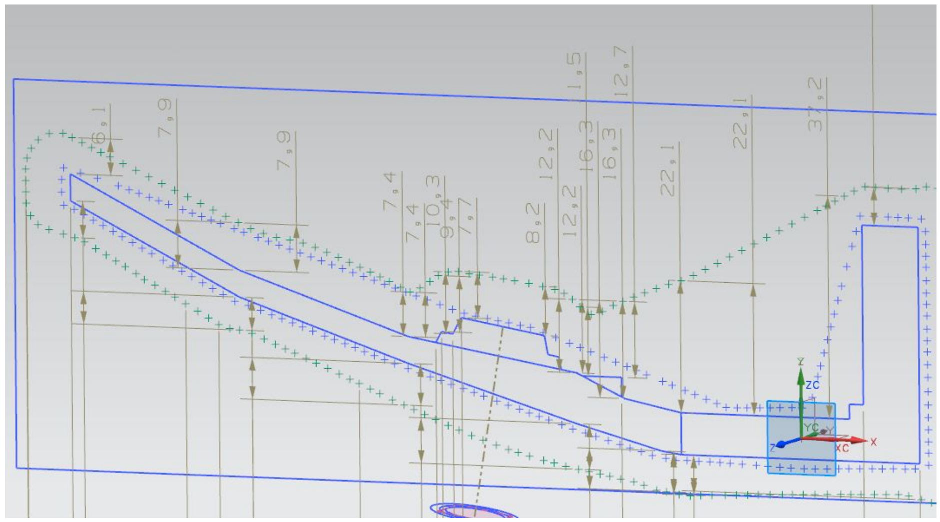


Figure 6.1: Diagram of the 2d imported point ring plus automatically created reference dimensions

FoSCo was the tool designed to predict the shape generated by the primary shaping process. It was identified earlier, in Section 2.4, that there are pre-existing methodologies which can predict this shape but they are limited in aforementioned ways, Section 2.4. To summarise however, methods consist of rules based methods, face and edge offset and other geometric methods, detailed in Section 4.6. Full rule based methods will be more accurate if correctly coded. However past research detailed in Section 2.4 shows extensive effort must be invested to create rule sets, and the rule sets created are not generic across components or manufacturing strategies. That is to say, if a rule based black forging shape generator for a disk is created, it may not be applicable for a blisk (bladed disk), or a bling (bladed ring), certainly not for a casing. This component specificity reduces the value for components which are not high value, or sufficiently generic, as it reduces the number of components the cost of rule finding and coding can be amortised over.

The objective of FoSCo was thus to provide a simple, effective and consistent method, with which to construct an estimation of the shape created by the potential primary shaping processes; a paramount determinant of the amount of material requiring removal. The primary shaping process is the process which creates a precursor shape, potential processes include: casting, forming and moulding processes.

It is appreciated that this methodology will rarely be as accurate as a full rules based approach, however its simplicity and thus maintainability is key to successful use. It is not designed as an exact tool to calculate the shape of a black forging, cast or moulded shape, but to provide a reasonable idea of the volumes to be removed for a cost estimate, in this it is successful.

FoSCo with co-efficients configured for ring roll forgings has been shown to be applicable, with modification to encompass a solid centre, to die forgings for blisks and disks.

Where the primary shaping process does not require significant aspects of vertex or edge smoothing, offset methods can be used instead. FoSCo should therefore be considered a modular program where the chosen module for modelling could be varied, according to the desired accuracy and the primary shaping process to be explored.

FoSCo was developed for primary shaping processes which result in smoothed surfaces, and this methodology in particular does not require extensive configuration of a rules base for new primary shaping processes, or the same ones applied to different components. In addition, the three dimensional version is simply an expansion of the two dimensional version, allowing the maintenance of similar rather than dissimilar models.

It has been found that from discussions with the industrial sponsor, and some testing with data made available by them, that the shapes FoSCo2d produces are reasonable estimations of the forged shape. Validation checks of FoSCo against the rules-based approach ACE (a proprietary software tool), created and used by the industrial sponsor showed an underestimate (by FoSCo) of forging volume by 8%. However it should be appreciated that this was using the co-efficients estimated for use of ring roll forging, and not the die forging process for blisks. Furthermore FoSCo's assumptions were a 6mm black forging stand off, wheras ACE is programmed with a 4mm COS stand off + 4mm black forging stand off distance. Ergo it can be stated that FoSCo is a reasonable estimate of shape, even without significant calibration and thus time expended in generation of rules.

6.3 MCOST

The objective of MCOST was to deduce the least costly method of manufacture for a component, given its geometric and tolerance information, and potential precursor shapes for different forming methods. In this it is successful. The major contribution made is towards the use of a random mutation hill climber with simulated annealing, to solve the combinatorial optimisation problem of process planning and sequencing, to obtain a cost optimised method of manufacture from appropriately parameterised CAD geometry. To date, most research has focussed only on the operation sequencing problem, little research has been published on the construction of the set of manufacturing processes. No other research has investigated genetic optimisation to construct the route, sequence it, and combine with cost modelling and scrap/rework calculations.

The reason this was required is because the optimal manufacturing route for a component is an amalgamation of the manufacturing route for each feature. However, the optimal manufacturing route for a feature depends not only on characteristics of the feature, but external factors, as proved in Section (5.4). These include characteristics of the component as a whole, but also characteristics of other features, and finally the placement of the manufacturing operation within the chain of operations. This last influence is the hardest to encompass, but necessary for the global optimum to be found. This research provides a method for allowing the holistic view required to find the least costly manufacturing route automatically and consistently, without dependency on the experience of the operator.

The creation and sequencing of the manufacturing has required, in order of use within the developed methodology:

1. The construction of a manufacturing route creator algorithm, which identifies which manufacturing process to use, using unweighted backward and forward propagation. This stage of analysis also uses multivariant normal distributions for process variability, to calculate the scrap and rework probabilities. It is accepted that this is an approximation, as previously stated this could be replaced by more advanced methodologies developed by, among others, Dodd et al. (2015).
2. The creation of an optimisation tool:
   (a) for choosing (Figure 6.2) the processes to be used.
   (b) and sequencing (Figure 6.3) the operations.
3. A deterministic methodology (DTMS) to choose the machine and tool access direction for each operation. DTMS can be easily expanded to cover other choices such as tooling, fixturing or choice of coolant etc. Its only limitation is the amount of computational resource available. Clearly the more choices introduced, and/or the more forward looking it is made, the more evaluation time is necessary.
4. A cost modelling methodology adapted from various papers detailed in the literature survey, but showing its industrial connections with its ability to be modified without access to source code. It is the authors opinion that a hard coded solution, to cost modelling, would be superior in speed and flexibility, however it would be impractical in some industrial organisations which have more limited agility than universities and small companies. A solution such as developed commercially by aPriori (a software company located in Concord, Massachusetts, USA), using code presumed to be compiled on run time would be equally good and could have been achieved through the use of python, however time was too limited to develop this at the time of the author's realisation.

The reasoning behind the multi-loop method of optimisation is to reduce the number of design variables which are considered at any one time, and the division between

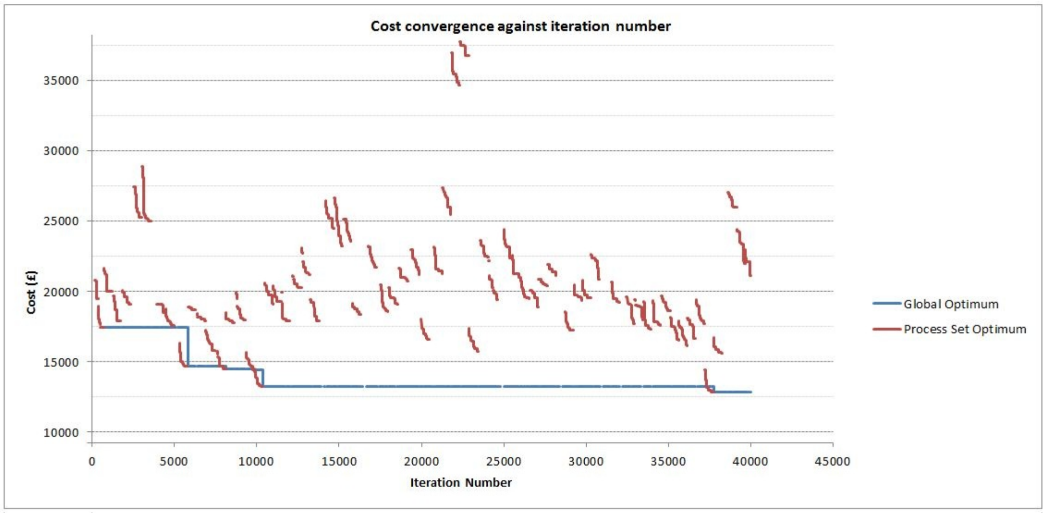


Figure 6.2: The entirety of the RMHC optimisation for test case one centrifugal casting, each red curve is one inner loop of the optimisation

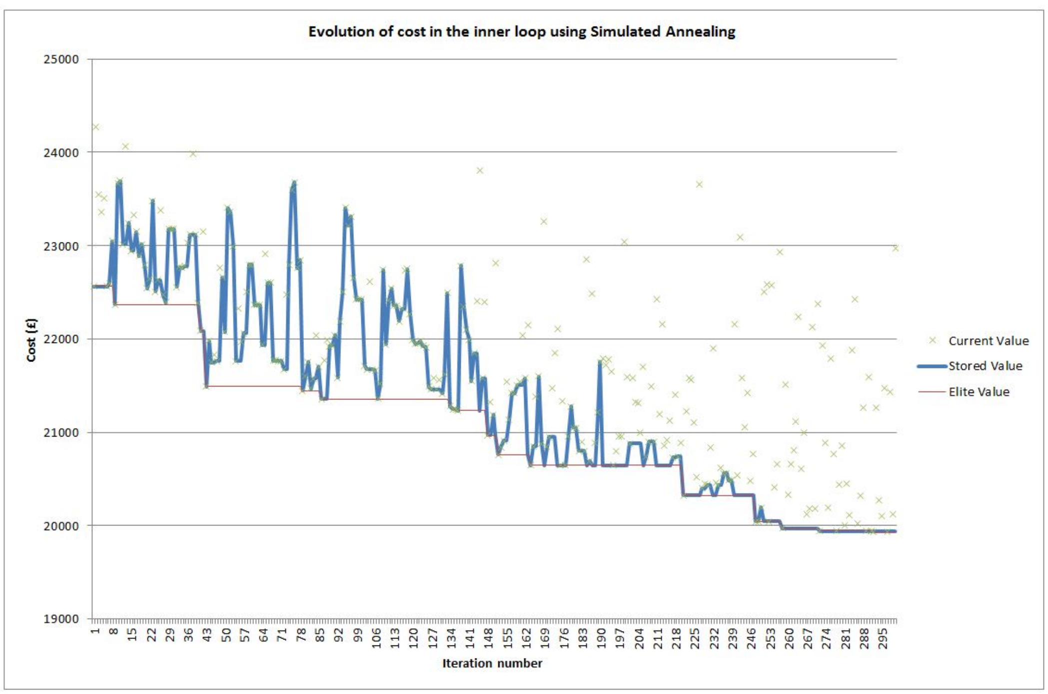


Figure 6.3: One inner loop optimisation, with simulated annealing and elite tracking

manufacturing process choice and sequence is a natural separation point. It would be undesirable to eliminate good manufacturing process sets, simply because an inferior order is applied to them. The methodology applied gives all manufacturing process sets an equal number of iterations to prove their superiority.

For the selection of manufacturing operations, the use of weighted neural nets was considered but felt undesirable because of the conditionality that results from exactly where the process will be in the sequence of operations. In addition this approach would require a set of training data to establish probabilities, thus detracting from the usability of the system.

For the selection of machines and the choice of TAD, the original version used random number selection to choose. However the developed DTMS methodology was found to produce superior results in a given number of iterations.

This research demonstrates that a system which is capable of both constructing and cost modelling a manufacturing route is feasible. The construction does not need to be rules based, which causes maintainability issues to keep it up to date, and is also unnecessarily limiting. The system constructed is not limited, as many previous systems are, in considering pre-coded manufacturing routes, nor does it require coding experts to update the manufacturing database.

The major advantage of this methodology over previous methodologies, is that it will change the manufacturing process for a specific elemental feature in response to changes of multiple attributes of the component. This is what has been termed dynamic process selection.

In test case one the global optimum may have been found, but this is unlikely as no results were repeated even on 12hr optimisation runs. It should therefore be accepted that, until research is available to the contrary, components of a level of complexity equivalent to test case two cannot be reliably optimised in a reasonable time, without recourse to high performance computing if the size of the design space is not reduced. Despite its lack of convergence however, the results shown in Figure 6.4 do clearly show that ring roll forging is the least costly method of manufacture for this component. Therefore design for manufacture activities should focus on how to best design for a ring roll forging. In further optimisations other forming processes can be eliminated, allowing more time and computational expense to be brought to bear on ring roll forging to further improve the manufacturing route.

6.4 Future Work FoSCo

For the two dimensional version, a method of automatically creating the convergence line, or better a polygon, would be beneficial. More testing is required to establish the

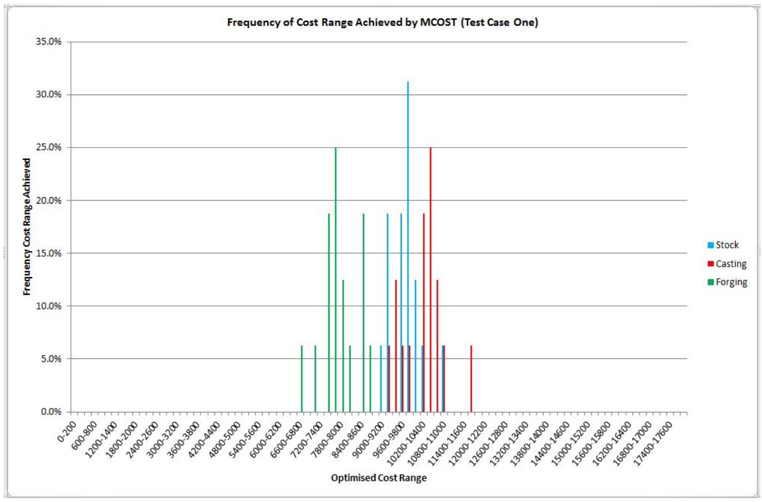


Figure 6.4: Test Case One Results: A graph showing the end result of 16 optimisation runs for each manufacturing strategy.

boundaries of its capabilities, in terms of the shapes and forming processes for which it is sufficiently accurate. Automatic detection of where non-hollow shapes are required, and the automatic creation of fixed points (Section 4.4.10) would be beneficial as well.

At present the convergence line is manually created and defined simply as a sequence of co-ordinate points, which are converted into a continuous line, it could be easily created and defined in a CAD package, although this was not done, which would allow it to move automatically with modifications to the basic outline if appropriately constrained. However, future work could focus on automatic creation of the convergence line as well. One possibility could be the use of cubic splines or similar, the line need not be straight, and attempting to maximise distance from the closest n points perhaps.

Future work for FoSCo mostly applies to the three-dimensional version, implementing the changes that were, and could be, made to the two-dimensional version such as: moving the smoothing algorithm to the final stages, convergence to a line, polygon, or (3d only) polyhedron, fixed points. Finally, consideration of the smoothing effect between the different circles which comprise the surface, this was clearly not an issue in two dimensional space, but becomes relevant for the 3d version. Implementation of a polygon or polyhedron convergence line would be essential for near net shape manufacturing processes, since as can be seen in Figure 6.5 the methodology does not converge to an accurate casting representation. This results in MCOST overestimating the times and manufacturing operations required and is the cause of the incorrect result in Section 5.12.4.

Given that at there are an infinite range of forming shapes which could be produced, there will be many sensible ones worthy of consideration. Even if only forging is considered there are a range of shapes with differing curvatures around various features, where the optimal forging shape may not be the minimum achievable. This depends on the interplay of: additional forging time (to achieve more complex shapes), material cost, offcut value (from

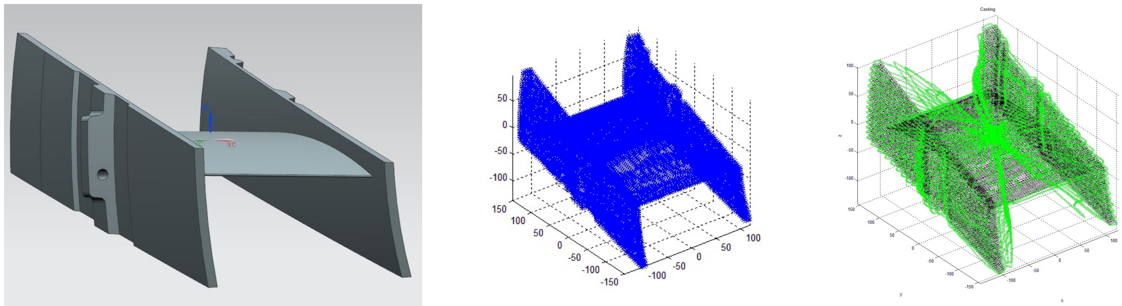


Figure 6.5: Test Case Two, ESS (Engine Section Stator) and its forging shape plot

machining), machining time, and cost. Thus in theory it would be best if the shape generated by FoSCo were to be part of the optimisation loop, given an appropriate parameterisation. This is thought too computationally expensive to achieve at the current time, but may be a consideration for future work. It would also require detailed and accurate models for forging, casting and other forming methods.

Because of the way the convergence point is found, there is the potential for shading issues in near net shape forming processes. One possible solution is to allow non equal spacing of the points on the convergence line in the same manner as adaptive grid control in CFD analysis. However for near net shape processes it may also be beneficial to use a dedicated tool and import (as in Section 4.4.11) the results of the shape to be generated in a similar manner to FoSCo simply from a point cloud/ring.

6.5 Future Work MCOST

A key area for research is to allow the machining of certain elemental features to affect others, their dimensions, and/or the tool access directions. This is because after certain features are machined, a new access direction may be accessible due to the removal of material which was previously an obstruction. Given that in certain cases the access direction of a material removal tool will affect the maximum material removal rate, allowing different access directions may materially reduce the time of the operation thus reducing the costs. There is the functionality in the method to modify tool access directions for features, but the design to elemental feature conversion needs to take advantage of this capability. This is considered a feature recognition type opportunity for improvement, and was thus left out of scope at this stage. In addition to this subtractive effect there could also be the opposite effect, where an additive process increases access difficulties. In this case, if possible, it would be beneficial if the optimiser recognised this, and ensured that the additive process was conducted after the subtractive.

The calculation used for the probability of scrap/rework created (Section 5.8.2) needs improvement. It is assumed that the probability of scrap/rework is dependent only on the tolerances assigned to the feature and the variability of the process set. There is no allowance for the number of elements (for example holes), or size of the feature (e.g. the machined area). This is clearly inaccurate since if the probability of successfully drilling a hole to within a specified tolerance is 99.9%, then the probability of accurately drilling 100 of them correctly is 90.47%, i.e. a scrap rate of 9.53%.

Further work could also be to combine with a factory model, either static or dynamic, which would enable consideration of the dynamic aspects of cost (for example machine/labour utilisation). This could be incorporated into the optimisation loop, or only executed on the 10 (or more) best results. Dynamic modelling inside the MCOST optimisation would be impractical because of the time taken. Currently MCOST can give a cost for a particular operation sequence within approximately 100 µs (on the test configuration computer). A dynamic simulation would take seconds to stabilise (on a similar specification machine) and give reasonable results. Furthermore the current methodology works for the insertion of a single new component into the supply network (which is assumed to have infinite capacity, and cost rates are unaffected by the new component). Dynamic simulation, on the other hand, should consider all new components to be produced, introducing multiple more dimensions to the optimisation problem. In addition there may be an opportunity to modify the manufacturing routes for all existing components, for example, if a new component introduced the requirement a new machine or process, other components could benefit. A factory model was constructed early in the research, but was not progressed to allow focus on the areas presented.

DTMS (the selection of machines and TADs) could also be improved, either by extending the number of future operations considered, or by considering more than just machine and TAD choices. Fixturing or different options on feed rates, for example, could also be considered. In addition, machine calibration time factors could be applied during DTMS outside of the feature based time calculation (Section 5.9.7).

Application of the repair algorithm (forming grouping and constraint based rearrangement) after each iteration of the route sequencing algorithm, followed by the check to see if a sequence has already been evaluated, may result in less time spent converging to target cost. It certainly results in fewer objective function evaluations, but possibly not less time due to computational time of repair. It remains to be investigated whether the repair function is beneficial or deleterious to computational time, but it would clearly become more beneficial if the cost model were to encompass more functionality. This functionality would be intended to increase accuracy and/or flexibility, and could be as outlined above with DTMS expansion, or extension of cost modelling equations.

There is currently no value assigned to scrapped components, which for a titanium or other valuable material may be substantial, reducing the impact of an operation creating non-conforming components. This research also assumes that a single stage of rework creates a conforming feature, that is to say no further rework or additional scrap is created by the rework operation. Clearly this is an approximation and the incorporation of work by DODD (2015) would improve accuracy. Waste material created by machining processes may well also have value, and again this is currently ignored in the cost equations constructed, however this is not an intrinsic failure of this methodology, it is simply that data has not been found and populated.

The use of ontological based approaches such as by ZHANG ET AL. (2014) could allow more complex, and useful choice of machine and tool, but potentially at the expense of computational time or usability. Considerations of industrial application at the time of research eliminated this approach at the time. Similarly the use of interpreted or compiled at run-time software languages could allow more complex cost calculations, such as exemplifed by the commercial program aPriori (Section 3.2). In the chosen language this is possible but it is not simple or fast.

Finally MUSHARAVATI/HAMOUDA (2015) showed a method of resolving the recombination problem of variable length chromosomes, this could be applied to resolving the recombination difficulties which prevented use of a genetic algorithm in this research, specifically within the outer loop of the optimisation.

6.5.1 Cost modelling

References to other features inside the cost modelling equations would be useful. For example if the component requires surface treatment (perhaps for wear or heat resistance), holes may require masking. This may require the data from multiple features, and it's possible that feature recognition is essential for this. Equally if a set of holes is to be drilled at a low angle of incidence it may be beneficial, if possible, to drill them earlier before the angled face is created; resulting in a deeper hole but benefitting from the higher angle of incidence.

aPriori has far more flexible/powerful cost modelling capability, allowing complete freedom of programming of equations and compiles on run-time. This type of approach allows time and cost equations more complex than the standard equation format used in this research. aPriori however is limited by its inability to cope with non geometric inputs, such as operating temperature. For example, the component requires a surface coating if $T > x$. It can also not cope with operation sequencing adequately, and thus cannot manage compound scrap effects.

6.5.2 Batch production

Batch production can be used on parts significantly smaller than machines to amortise setup costs and reduce wastage. It is thus essential to consider for small parts where otherwise the setup costs would unduly distort the total cost. This was discovered as the cause of major inaccuracies in cost for the 20 additional test cases. It is thought that the most accurate method to amortise the costs would be a three dimensional layout planner, to calculate the maximum number of parts which can be batched. An easier, and therefore faster method, would be for machines to have a volume or area associated with them, the part would need the corresponding dimension, which could then be used to calculate batch sizes. Accuracy could be compromised if the methodology did not take into account the changing dimensions throughout the design process, but this is not insurmountable and would be trivial in some cases.

6.5.3 Assemblies

The assembly process was considered for incorporation in this thesis, and cost/time equations for the Electron Beam Welding process were obtained. It is thought that the cost of an assembly could be modelled by taking previously derived costs for the component parts, which may be assemblies themselves, and calculating the costs of joining them. The costs of joining could be calculated by MCOST in the same way as the operation costs are derived, by specifying the parameters of the joining feature and access directions, and letting MCOST determine the process set (including preparation, clean up, and inspection operation if necessary) and calculate the costs of each operation in the way described in Section 5.9.7.

6.6 Future work Component Family Definition

At present, creation of a new component family - which results from a change in the type of features on a component, is a manual process. An appropriately parameterised (i.e. parameterised to allow the flexibility requried of it by the designers/design system) CAD model must be constructed and linked to the Component Interface Spreadsheet (CIS) (Section 7.3). However, the reason for illustrating the design feature automation work in Section 1.7.2 was to highlight the fact that an automatically inserted library feature could populate an additional line/lines in the CIS.

The CIS could also potentially become a database, or part of a database, which would enable more flexibility. This would be recommended, but would need to be carefully thought out through the use of structures,dictionaries etc., so as not to impact execution speed of the cost model.

6.7 Concluding remarks

In the author's view, the ultimate objective of design optimisation research is to achieve full computational value based design optimisation of a component/assembly. This type of value based approach requires information including, but not limited to, potential design options, and for each of these: original manufacturing cost and replacement cost for scheduled and unscheduled repairs, in addition to the disruption cost of repairs. A model of value to the customer is also required, which takes account of factors such as take off thrust, specific fuel consumption, lifetime and others.

The deduction of manufacturing cost depends on knowledge of the optimum manufacturing routes, and hence an approach such as outlined in this research is a necessary pre-requisite.

It has been shown that in order to deduce the optimal manufacturing route for a component, from a cost perspective, it is necessary to consider a holistic view of the method of manufacture, including detailed knowledge of the forming process. This research provides this holistic view computationally and automatically, although, as highlighted in the future work section, there are many improvements which can be made.

References

Agyapong-Kodua, K./Brown, R./Darlington, R./Ratchev, S. (2012): An integrated product–process design methodology for cost-effective product realisation, in: International Journal of Computer Integrated Manufacturing, 25(9), S. 814–828.

Agyapong-Kodua, K./Darlington, R./Ratchev, S. (2013): Towards the derivation of an integrated design and manufacturing methodology, in: International Journal of Computer Integrated Manufacturing, 26(6), S. 527 – 539, ISSN 0951192X.

Amin-Naseri, M./Afshari, A. J. (2012): A hybrid genetic algorithm for integrated process planning and scheduling problem with precedence constraints, in: International Journal of Advanced Manufacturing Technology, 59(1-4), S. 273 – 287, ISSN 02683768.

Anjum, N./Harding, J. A./Young, R. I. M./Case, K. (2012): Manufacturability verification through feature-based ontological product models, in: Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 226(6), S. 1086 – 1098, ISSN 09544054.

Ayyappan, S./Kalaimathi, M./Venkatachalam, G. (2015): Cost-tolerance prediction models for electrochemical machining of metal matrix composites, in: Engineering Review, 35(3), S. 299 – 307, ISSN 13309587.

Azadi Moghaddam, M./Kolahan, F. (2015): Optimization of EDM process parameters using statistical analysis and simulated annealing algorithm, in: International Journal of Engineering, Transactions A: Basics, 28(1), S. 157 – 166, ISSN 17281431.

Babic, B./Nesic, N./Miljkovic, Z. (2008): A review of automated feature recognition with rule-based pattern recognition, in: Computers in Industry, 59(4), S. 321 – 337, ISSN 01663615.

Bao, H. P./Samareh, J. (2000): Affordable design: A methodology to implement process-based manufacturing cost models into the traditional performance-focused multidisciplinary design optimization, Long Beach, CA, United states.

Behrens, B.-A./Denkena, B./Charlin, F./Dannenberg, M. (2011): Model based optimization of forging process chains by the use of a genetic algorithm, S. 154 – 158, Aachen, Germany.

Bettwy, M./Deschinkel, K./Gomes, S. (2014): An Optimization Tool for Process Planning and Scheduling, Bd. 438, S. 443 – 450, Ajaccio, France, ISSN 18684238.

Biba, N./Stebounov, S./Lishiny, A. (2001): Cost effective implementation of forging simulation, in: Journal of Materials Processing Technology, 113(1), S. 34–39.

BISWAS, S. K./KNIGHT, W. A. (1976): Towards an integrated design and production system for hot forging dies, in: International Journal of Production Research, 14(1), S. 23–49, doi:10.1080/00207547608956578.

Blose, R./Walker, B./Walker, R./Froes, S. (2006): New opportunities to use cold spray process for applying additive features to titanium alloys, in: Metal Powder Report, 61(9), S. 30–37.

Boothroyd, G. (1982): Design for producibility–the road to higher productivity, in: Assembly Engineering, S. 42–45.

Bramley, A. N./Mynors, D. (2000): The use of forging simulation tools, in: Materials & Design, 21(4), S. 279–286.

Brunetti, G./Golob, B. (2000): Feature-based approach towards an integrated product model including conceptual design information, in: CAD Computer Aided Design, 32(Compendex), S. 877–887.

Caporalli, A./Gileno, L. A./Button, S. T. (1998): Expert system for hot forging design, in: Journal of Materials Processing Technology, 80,81, S. 131–135.

Caputo, A. C./Pelagagge, P. M./Salini, P. (2016): Manufacturing cost model for heat exchangers optimization, in: Applied Thermal Engineering, 94, S. 513 – 533, ISSN 13594311.

Chen, J./Zhang, L.-W./Luo, J.-Q. (2009): Study on reconfiguration cost modelling of Reconfigurable Manufacturing System, in: International Technology and Innovation Conference 2009, ITIC 2009, October 12, 2009 - October 14, 2009, Bd. 2009vonIET Conference Publications, Institution of Engineering and Technology.

Cheng, K. (2011): Study of CAPP decision-making system based on experience, in: Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on, S. 4338–4341, IEEE.

Cheung, J./Scanlan, J./Wiseall, S. (2008): Value driven design: an initial study applied to novel aerospace components in Rolls-Royce plc, S. 241–248, Springer.

Cheung, J./Scanlan, J./Wong, J./Forrester, J./Eres, H./Collopy, P./Hollingsworth, P./Wiseall, S./Briceno, S. (2010): Application of value-driven design to commercial aero-engine systems, in: Journal of Aircraft, S. 1–31.

Chiu, N. H./Weigh, G. T. (2012): Optimization of Micro-Milling Process with Genetic Algorithm, in: Advanced Materials Research, 383, S. 7111–7116.

C/O Boyce, R. (2011): Standard Features Ontology, Rolls-Royce plc.

C/O Johnson, C. (2010): Data on the manufacturing route: SAP Export, Rolls-Royce plc.

C/O Johnson, C. (2011a): Data pack for ESS Anit-Icing, Rolls-Royce plc.

C/O Johnson, C. (2011b): Features for Front Bearing Structure, Rolls-Royce plc.

C/O Johnson, C. (2012): Commodity information on the Front Bearing Housing, Rolls-Royce plc.

C/O Reuss, D. (2010): Process rates for machining operations, Rolls-Royce plc.

C/O Wiseall, S. (2010): Process rates for drilling type operations, Rolls-Royce plc.

Culler, D. E./Burd, W. (2007): A framework for extending computer aided process planning to include business activities and computer aided design and manufacturing (CAD/CAM) data retrieval, in: Robotics and Computer-Integrated Manufacturing, 23(3), S. 339 – 350, ISSN 07365845.

Davis, L. (1985): Applying adaptive algorithms to epistatic domains, in: Proceedings of the 9th international joint conference on Artificial intelligence - Volume 1, S. 162–164, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN 0-934613-02-8, 978-0-934-61302-6.

Denkena, B./Shpitalni, M./Kowalski, P./Molcho, G./Zipori, Y. (2007): Knowledge Management in Process Planning, in: CIRP Annals - Manufacturing Technology, 56(1), S. 175 – 180, ISSN 0007-8506, doi:DOI:10.1016/j.cirp.2007.05.042.

Di Stefano, P./Bianconi, F./Di Angelo, L. (2004): An approach for feature semantics recognition in geometric models, in: CAD Computer Aided Design, 36(10), S. 993 – 1009, ISSN 00104485.

Dodd, C. (2015): A paradigm to maximise performance and profitability of engineering products in the presence of manufacturing uncertainty.

Dodd, C./Scanlan, J./Marsh, R./Wiseall, S. (2015): Improving profitability of optimal mean setting with multiple feature means for dual quality characteristics, in: The International Journal of Advanced Manufacturing Technology, S. 1–14.

Dong, J./Vijayan, S. (1997a): Manufacturing feature determination and extraction - Part I: Optimal volume segmentation, in: CAD Computer Aided Design, 29(6), S. 427 – 440, ISSN 00104485.

Dong, J./Vijayan, S. (1997b): Manufacturing feature determination and extraction - Part II: A heuristic approach, in: CAD Computer Aided Design, 29(7), S. 475 – 484, ISSN 00104485.

El-Mounayri, H. A. (1997): Generic solid modelling based machining process simulation, in: ProQuest Dissertations and Theses Global.

Esterman Jr, M./Kamath, K. (2010): Design for Assembly Line Performance: The Link Between DFA Metrics and Assembly Line Performance Metrics, ASME.

Feng, S. C. (2003): A machining process planning activity model for systems integration, in: Journal of Intelligent Manufacturing, 14(6), S. 527 – 539, ISSN 09565515.

Franca, P. M./Mendes, A./Moscato, P. (2001): A memetic algorithm for the total tardiness single machine scheduling problem, in: European Journal of Operational Research, 132(1), S. 224 – 242, ISSN 03772217.

Gao, S./Shah, J. J. (1998): Automatic recognition of interacting machining features based on minimal condition subgraph, in: Computer-Aided Design, 30(9), S. 727 – 739, ISSN 0010-4485, doi:DOI:10.1016/S0010-4485(98)00033-5.

Gao, X./Li, Z./Li, L. (2008): A process model for concurrent design in manufacturing enterprise information systems, in: Enterprise Information Systems, 2(Compendex), S. 33–46.

Ghadge, A./Dani, S./Kalawsky, R. (2010): A framework for managing risks in the aerospace supply chain using systems thinking, in: 2010 5th International Conference on System of Systems Engineering, SoSE 2010, June 22, 2010 - June 24, 2010, 2010 5th International Conference on System of Systems Engineering, SoSE 2010, IEEE Computer Society.

Goldberg, D. E. (1989): Genetic Algorithms in Search, Optimization and Machine Learning, 1. Aufl., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, ISBN 0201157675.

Gologlu, C. (2004): A constraint-based operation sequencing for a knowledge-based process planning, in: Journal of Intelligent Manufacturing, 15, S. 463–470, ISSN 0956-5515.

Gorges-Schleuter, M. (1989): ASPARAGOS An Asynchronous Parallel Genetic Optimization Strategy, in: Proceedings of the 3rd International Conference on Genetic Algorithms, S. 422–427, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN 1-55860-066-3.

Grabowik, C./Kalinowski, K./Monica, Z. (2005): Integration of the CAD/CAPP/PPC systems, in: Journal of Materials Processing Technology, 164, S. 1358–1368.

Grabowik, C./Knosala, R. (2003): The method of knowledge representation for a CAPP system, in: Journal of Materials Processing Technology, 133(1), S. 90–98.

Gronostajski, Z./Hawryluk, M./Kaszuba, M./Misiun, G./Niechajowicz, A./Polak, S./PaweLczyk, M. (2015): An analysis of the industrial forging process of flange in order to reduce the weight of the input material, in: Archives of Metallurgy and Materials, 60(2A), S. 849 – 853, ISSN 17333490.

Han, J./Pratt, M./Regli, W. C. (2000): Manufacturing feature recognition from solid models: A status report, in: IEEE Transactions on Robotics and Automation, 16(6), S. 782 – 796, ISSN 1042296X.

Han, J. H./Han, I./Lee, E./Yi, J. (2001): Manufacturing feature recognition toward integration with process planning, in: IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 31(3), S. 373 – 380, ISSN 10834419.

Hiller, E. (2014): SEER Composites Brings Cost Estimating to CATIA.

Hoque, A. S. M./Szecsi, T. (2008): Designing using manufacturing feature library, in: Journal of Materials Processing Technology, 201(Compendex), S. 204–208.

Hu, D./Cao, W./Liu, Y. (2009): Study on CNC/CAD/CAPP/CAM integration system modeling for rotational parts, in: International Technology and Innovation Conference 2009, ITIC 2009, October 12, 2009 - October 14, 2009, Bd. 2009vonIET Conference Publications, Institution of Engineering and Technology.

Huang, S. J. (2013): Optimisation of process routing in CAPP using genetic algorithm and graph theory, Bd. 385-386, S. 1747 – 1751, Tianjin, China, ISSN 16609336.

Huang, W./Hu, Y./Cai, L. (2012): An effective hybrid graph and genetic algorithm approach to process planning optimization for prismatic parts, in: The International Journal of Advanced Manufacturing Technology, 62(9-12), S. 1219–1232.

Huang, W.-j./Hu, Y.-j. (2011): Hybrid evolutionary multi-objective approaches to process planning global optimization for complex parts, in: Industrial Engineering and Engineering Management (IE&EM), 2011 IEEE 18Th International Conference on, S. 709–712, IEEE.

Isnaini, M. M./Shirase, K. (2014): Review of computer-aided process planning systems for machining operation - Future development of a computer-aided process planning system, in: International Journal of Automation Technology, 8(3), S. 317 – 332, ISSN 18817629.

Iwakowicz, R. R. (2016): An efficient evolutionary method of assembly sequence planning for shipbuilding industry, in: Assembly Automation, 36(1), S. 60 – 71, ISSN 01445154.

Jiang, T./Guo, S./Wang, S./Wei, R. (2007): Universal study on parametric models in equipment system cost estimation, in: 2007 International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM 2007, September 21, 2007 - September 25, 2007, 2007 International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM 2007, S. 5223–5226, Inst. of Elec. and Elec. Eng. Computer Society.

Jinks, S./Scanlan, J./Reed, P./Wiseall, S. (2010): Utilising dynamic factory simulation to improve unit cost estimation and aid design decisions, in: 2010 43rd Winter Simulation Conference, WSC'10, December 5, 2010 - December 8, 2010, Proceedings - Winter Simulation Conference, S. 1758–1766, Institute of Electrical and Electronics Engineers Inc.

Joshi, S./Chang, T. C. (1988): Graph-based heuristics for recognition of machined features from a 3D solid model, in: Comput. Aided Des., 20, S. 58–66, ISSN 0010-4485, doi:10.1016/0010-4485(88)90050-4.

Kafashi, S./Shakeri, M. (2011): Application of genetic algorithm in integrated setup planning and operation sequencing, in: AIP Conference Proceedings, 1315(1), S. 1413–1418, doi:10.1063/1.3552384.

Kang, M./Han, J./Moon, J. (2003): An approach for interlinking design and process planning, in: Journal of Materials Processing Technology, 139(1-3 SPEC), S. 589 – 595, ISSN 09240136.

Karunakaran, K. P./Suryakumar, S./Pushpa, V./Akula, S. (2010): Low cost integration of additive and subtractive processes for hybrid layered manufacturing, in: , (Compendex).

Kass, M./Witkin, A./Terzopoulos, D. (1988): Snakes: Active contour models, in: International journal of computer vision, 1(4), S. 321–331.

Khalifa, R. B./Yahia, N. B./Zghal, A. (2006): New neural networks approach for optimal machines tools selection, in: Information Control Problems in Manufacturing, Bd. 12, S. 461–466.

Khoshnevis, B./Park, J. Y./Sormaz, D. (1994): Cost based system for concurrent part and process design, in: Engineering Economist, 40(1), S. 101 – 124, ISSN 0013791X.

Kim, Y. (1992): Recognition of form features using convex decomposition, in: Computer-Aided Design, 24(9), S. 461 – 476, ISSN 0010-4485, doi:DOI:10.1016/0010-4485(92)90027-8.

Kulon, J./Mynors, D./Broomhead, P. (2006): A knowledge-based engineering design tool for metal forging, in: Journal of Materials Processing Technology, 177(1), S. 331–335.

Kumar, M./Rajotia, S. (2005): Development of a generative CAPP system for axisymmetric components for a job shop environment, in: International Journal of Advanced Manufacturing Technology, 27(1-2), S. 136 – 144, ISSN 02683768.

Lee, H./Lee, J. (2011): The Optimal Cost Estimating for Improvement of Product Design, in: International Journal of Industrial Engineering: Theory, Applications and Practice, 18(5).

Lee, H. C./Jhee, W. C./Park, H.-S. (2007): Generative CAPP through projective feature recognition, in: Computers and Industrial Engineering, 53(2), S. 241 – 246, ISSN 03608352.

Leo Kumar, S./Jerald, J./Kumanan, S. (2015): Feature-based modelling and process parameters selection in a CAPP system for prismatic micro parts, in: International Journal of Computer Integrated Manufacturing, 28(10), S. 1046 – 1062, ISSN 0951192X.

Leo Kumar, S./Jerald, J./Kumanan, S./Aniket, N. (2014): Process parameters optimization for micro end-milling operation for CAPP applications, in: Neural Computing and Applications, 25(7-8), S. 1941 – 1950, ISSN 09410643.

Leus, R./Zhang, L. L./Kowalczyk, D. (2014): Process family planning: An optimization-based approach, Bd. 2015-January, S. 501 – 505, Selangor, Malaysia, ISSN 21573611.

Li, J./Wang, A./Tang, C. (2014a): Production planning in virtual cell of reconfiguration manufacturing system using genetic algorithm, in: International Journal of Advanced Manufacturing Technology, 74(1-4), S. 47 – 64, ISSN 02683768.

Li, J./Yang, X./Ren, C./Chen, G./Wang, Y. (2014b): Multiobjective optimization of cutting parameters in Ti-6Al-4V milling process using nondominated sorting genetic algorithm-II, in: International Journal of Advanced Manufacturing Technology, 76(5-8), S. 941 – 953, ISSN 02683768.

Li, S./Liu, Y./Landers, R. G. (2015): An adaptive evolutionary algorithm for optimizing process planning of parallel drilling operations, in: International Journal of Advanced Manufacturing Technology, 79(5-8), S. 1247 – 1263, ISSN 02683768.

Li, W./Ong, S./Lu, Y./Nee, A. (2003): A Tabu search-based optimization approach for process planning, Bd. 2774, S. 1000 – 1007, Oxford, United kingdom, ISSN 03029743.

Lian, K./Zhang, C./Li, X./Gao, L. (2009): An Effective Hybrid Genetic Simulated Annealing Algorithm for Process Planning Problem, in: Natural Computation, 2009. ICNC ’09. Fifth International Conference on, Bd. 5, S. 367–373, doi:10.1109/ICNC.2009.689.

Liu, H. J./Kou, L. F./San, L./Cao, N. J. (2014): Rapid NC machining technology for hole and groove parts based on UG feature recognition, Bd. 635-637, S. 579 – 583, Hangzhou, China, ISSN 16609336.

Lv, S./Qiao, L. (2013): A cross-entropy-based approach for the optimization of flexible process planning, in: The International Journal of Advanced Manufacturing Technology, S. 1–12.

Manogharan, G./Wysk, R. A./Harrysson, O. L. (2016): Additive manufacturing-integrated hybrid manufacturing and subtractive processes: Economic model and analysis, in: International Journal of Computer Integrated Manufacturing, 29(5), S. 473 – 488, ISSN 0951192X.

Marion, T. J. (2008): On the use of global cost modeling early in new product development, in: IEMC-Europe 2008 - 2008 IEEE International Engineering Management Conference, Europe: Managing Engineering, Technology and Innovation for Growth, June 28, 2008 - June 30, 2008, IEMC-Europe 2008 - 2008 IEEE International Engineering Management Conference, Europe: Managing Engineering, Technology and

Innovation for Growth, S. IEEE Technology Management Council; IEEE Portugal Section, Inst. of Elec. and Elec. Eng. Computer Society.

Marri, H./Gunasekaran, A./Grieve, R. (1998): Computer-aided process planning: A state of art, in: International Journal of Advanced Manufacturing Technology, 14(4), S. 261 – 268, ISSN 02683768.

Masel, D. T./Young II, W. A./Judd, R. P. (2010): A rule-based approach to predict forging volume for cost estimation during product design, in: The International Journal of Advanced Manufacturing Technology, 46(1-4), S. 31–41.

Meas, J. (2015): Ultrasonic analysis: a key driver in manufacturing design cost optimisation for high pressure turbine discs.

Morgan, J. M./Liker, J. K. (c2006.): The Toyota product development system : integrating people, process, and technology, Productivity Press„ New York :, ISBN ISBN: 1563272822 (alk. paper).

Morinaga, E./Joko, H./Wakamatsu, H./Arai, E. (2015): A computer-aided process planning method considering production scheduling, Bd. 459, S. 348 – 355, Tokyo, Japan, ISSN 18684238.

Musharavati, F./Hamouda, A. (2011): Modified genetic algorithms for manufacturing process planning in multiple parts manufacturing lines, in: Expert Systems with Applications, 38(9), S. 10.770 – 10.779, ISSN 0957-4174, doi:http://dx.doi.org/10.1016/j.eswa.2011.01.129.

Musharavati, F./Hamouda, A. M. S. (2015): Multiple parts process planning in serialparallel flexible flow lines: part IIsolution method based on genetic algorithms with fixed- and variable-length chromosomes, in: International Journal of Advanced Manufacturing Technology, 77(5-8), S. 1105 – 1143, ISSN 02683768.

Nasr, E. A./Al-Ahmari, A./Khan, A. A./Kamrani, A. (2015): An automatic fixture modeling system using search strategy, in: Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 229(12), S. 2165 – 2179, ISSN 09544054.

Ouyang, H.-B. (2015): A STEP-compliant intelligent process planning system for milling, in: Open Automation and Control Systems Journal, 7(1), S. 510 – 520, ISSN 18744443.

Papadimitriou, C. H./Steiglitz, K. (1998): Combinatorial optimization: algorithms and complexity, Courier Dover Publications.

Park, C. H./Saouab, A./Breard, J./Han, W. S./Vautrin, A. (2006): Integrated optimization for weight, performance and cost of composite stuructures, in: Information Control Problems in Manufacturing, Bd. 12, S. 807–812.

Park, S. C. (2003): Knowledge capturing methodology in process planning, in: CAD Computer Aided Design, 35(12), S. 1109 – 1117, ISSN 00104485.

Perez, H./Rios, J./Diez, E./Vizan, A. (2008): Increase of material removal rate in peripheral milling by varying feed rate, in: Journal of Materials Processing Technology, 201(Compendex), S. 486–490.

Petrovi, M./Miti, M./Vukovi, N./Miljkovi, Z. (2015): Chaotic particle swarm optimization algorithm for flexible process planning, in: International Journal of Advanced Manufacturing Technology, S. 1 – 21, ISSN 02683768.

Pierreval, H./Bruniaux, R./Caux, C. (2007): A continuous simulation approach for supply chains in the automotive industry, in: Simulation Modelling Practice and Theory, 15(Compendex), S. 185–198.

Prabhu, B./Biswas, S./Pande, S. (2001): Intelligent system for extraction of product data from CADD models, in: Computers in Industry, 44(1), S. 79 – 95, ISSN 01663615.

Prakash, A./Chan, F./Deshmukh, S. (2012): Application of knowledge-based artificial immune system (KBAIS) for computer aided process planning in CIM context, in: International Journal of Production Research, 50(18), S. 4937 – 4954, ISSN 00207543.

Qian, L./Ben-Arieh, D. (2008): Parametric cost estimation based on activity-based costing: A case study for design and development of rotational parts, in: International Journal of Production Economics, 113(Compendex), S. 805–818.

Rauch, M./Hascoet, J.-Y. (2012): A manufacturing system for advanced multi-process manufacturing based on STEP-NC, Bd. 4, S. 71 – 79, Nantes, France.

Razali, N. (2014): Process sequencing modeled as TSP with precedence constraints - A genetic algorithm approach, Bd. 575, S. 843 – 847, Tianjin, China, ISSN 16609336.

Ribeiro, I./Pecas, P./Henriques, E. (2012): Comprehensive model to evaluate the impact of tooling design decisions on the life cycle cost and environmental performance, S. 515 – 520, Berkeley, CA, United states.

Rocha, J./Ramos, C./Vale, Z. (1999): Process planning using a genetic algorithm approach, in: Assembly and Task Planning, 1999.(ISATP'99) Proceedings of the 1999 IEEE International Symposium on, S. 338–343, IEEE.

Rolls-Royce (2013): Rolls-Royce Interactive Games `http://www.rolls-royce.com/interactive_games/journey03`, Rolls-Royce plc.

Sadaiah, M./Yadav, D./Mohanram, P./Radhakrishnan, P. (2002): A generative computer-aided process planning system for prismatic components, in: International Journal of Advanced Manufacturing Technology, 20(10), S. 709 – 719, ISSN 02683768.

Salehi, M./Tavakkoli-Moghaddam, R. (2009): Application of genetic algorithm to computer-aided process planning in preliminary and detailed planning, in: Engineering Applications of Artificial Intelligence, 22(8), S. 1179 – 1187, ISSN 0952-1976, doi:DOI:10.1016/j.engappai.2009.04.005.

Sanz-Lobera, A./Gomez, E./Perez, J./Sevilla, L. (2015): A proposal of cost-tolerance models directly collected from the manufacturing process, in: International Journal of Production Research, ISSN 00207543.

Scanlan, J./Rao, A./Bru, C./Hale, P./Marsh, R. (2006): DATUM project: Cost estimating environment for support of aerospace design decision making, in: Journal of Aircraft, 43(Compendex), S. 1022–1028.

Semiatin, S. (2005): ASM Handbook, Volume 14A: Metalworking: Bulk Forming, ASM Handbook Series, ASM International, ISBN 9780871707086.

Shah, J. J./Yan, Y./Zhang, B.-C. (1998): Dimension and tolerance modeling and transformations in feature based design and manufacturing, in: Journal of Intelligent Manufacturing, 9, S. 475–488, ISSN 0956-5515.

Singh, M./Khan, I./Grover, S. (2011): Selection of manufacturing process using graph theoretic approach, in: International Journal of System Assurance Engineering and Management, 2(4), S. 301–311.

Skiba, T./Baufeld, B./Van Der Biest, O. (2009): Microstructure and mechanical properties of stainless steel component manufactured by shaped metal deposition, in: ISIJ International, 49(Compendex), S. 1588–1591.

Sommerville, M. G. L./Clark, D. E. R./Corney, J. R. (2001): Viewer-centered geometric feature recognition, in: Journal of Intelligent Manufacturing, 12, S. 359–375, ISSN 0956-5515.

Sormaz, D. N./Arumugam, J./Harihara, R. S./Patel, C./Neerukonda, N. (2010): Integration of product design, process planning, scheduling, and FMS control using XML data representation, Bd. 26vonRobotics and Computer-Integrated Manufacturing, S. 583–595, Elsevier Ltd.

Sormaz, D. N./Khoshnevis, B. (1997): Process planning knowledge representation using an object-oriented data model, in: International Journal of Computer Integrated Manufacturing, 10(1-4), S. 92 – 92, ISSN 0951192X.

Sortino, M./Belfio, S./Totis, G. (2015): An innovative approach for automatic generation, verification and optimization of part programs in turning, in: Journal of Manufacturing Systems, 36, S. 168 – 181, ISSN 02786125.

Su, Y./Chu, X./Chen, D./Sun, X. (2015): A genetic algorithm for operation sequencing in CAPP using edge selection based encoding strategy, in: Journal of Intelligent Manufacturing, ISSN 09565515.

Tam, W. (2004): Improvement opportunities for aerospace design process, in: Space, S. 28–30.

Tammineni, S./Rao, A./Scanlan, J./Reed, P./Keane, A. (2009): A knowledge-based system for cost modelling of aircraft gas turbines, in: Journal of Engineering Design, 20(3), S. 289–305.

Tao, J./Zhang, T. C./Liu, P. T./Li, J./Mang, Y. (2008): Numerical computation of a linear friction welding process, Bd. 575-578, S. 811 – 815, Zhengzhou, China, ISSN 02555476.

Taufik, R. S./Sulaiman, S./Baharudin, B./Arifin, M./Hambali, A. A. (2011): Design and Simulation on Investment Casting Mold for Metal Matrix Composite Material, in: Applied Mechanics and Materials, 66, S. 1676–1681.

Tavcar, J./Slak, A./Duhovnik, J. (2013): Concurrent process planning and scheduling applied into production of turned parts, S. 837 – 848, Trier, Germany, ISSN 18655440.

Teich, T./Militzer, J./Jahn, F./Neumann, T./Kretz, D. (2010): Step standardized product data representation and exchange for optimized product development and automated process planning, Bd. 66 AISC, S. 41 – 56, Hong Kong, Hong kong, ISSN 18675662.

Thomas, M./Baxter, G. J./Todd, I. (2016): Normalised model-based processing diagrams for additive layer manufacture of engineering alloys, in: Acta Materialia, 108, S. 26 – 35, ISSN 13596454.

Thompson, S. C./Paredis, C. J. J. (2010): An investigation into the decision analysis of design process decisions, in: Journal of Mechanical Design, Transactions of the ASME, 132(Compendex).

Tikhon, M./Ko, T. J./Lee, S. H./Sool Kim, H. (2004): NURBS interpolator for constant material removal rate in open NC machine tools, in: International Journal of Machine Tools and Manufacture, 44(Compendex), S. 237–245.

Toal, D. (2012): personal communication, converation.

Vandenbrande, J. H./Requicha, A. A. (1993): Spatial reasoning for the automatic recognition of machinable features in solid models, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(12), S. 1269 – 1285, ISSN 01628828.

Wan, L./Li, X./Gao, L./Wen, X./Wang, W. (2013): A novel two-level genetic algorithm for integrated process planning and scheduling, S. 2790 – 2795, Manchester, United kingdom.

Wang, J./Du, B./Ding, H. (2011): A Modified Genetic Algorithm(GA) for Optimization of Process Planning, in: Journal of Computers, 6(7).

Wang, W./Li, X./Gao, L./Wen, X./Wan, L. (2013): Application of interval theory and genetic algorithm for uncertain integrated process planning and scheduling, S. 2796 – 2801, Manchester, United kingdom.

Wasim, A./Shehab, E./Abdalla, H./Al-Ashaab, A./Sulowski, R./Alam, R. (2013): An innovative cost modelling system to support lean product and process development, in: International Journal of Advanced Manufacturing Technology, 65(1-4), S. 165 – 181, ISSN 02683768.

Wen, X./Li, X./Gao, L./Wan, L./Wang, W. (2013): Multi-objective genetic algorithm for integrated process planning and scheduling with fuzzy processing time, S. 293 – 298, Hangzhou, Zhejiang, China.

Wiedenmann, R./Zaeh, M. F. (2015): Laser-assisted milling-Process modeling and experimental validation, in: CIRP Journal of Manufacturing Science and Technology, 8, S. 70 – 77, ISSN 17555817.

Williams, R. G. (1991): An intuitive parametric cost forecasting model for assembled products, in: Proceedings of the 1991 Portland International Conference on Management of Engineering and Technology - PICMET '91, October 27, 1991 - October 31, 1991, S. 304–308, Publ by IEEE.

Xu, L./Deng, W./Liu, W./Ma, S./Li, A./Matta, A. (2014): Optimization of process planning for cylinder block based on feature machining elements, Bd. 2014-January, S. 2575 – 2580, San Diego, CA, United states, ISSN 1062922X.

Xu, X./Fang, S./Gu, X. (2006): A model for manufacturing cost estimation based on machining feature, in: International Technology and Innovation Conference 2006, ITIC 2006, November 6, 2006 - November 7, 2006, IET Conference Publications, S. 273–278, Institution of Engineering and Technology.

Xu, X./He, Q. (2004): Striving for a total integration of CAD, CAPP, CAM and CNC, in: Robotics and Computer-Integrated Manufacturing, 20(2), S. 101 – 109, ISSN 07365845.

Xu, X./Wang, H./Mao, J./Newman, S./Kramer, T./Proctor, F./Michaloski, J. (2005): STEP-compliant NC research: The search for intelligent CAD/CAPP/CAM/CNC integration, in: International Journal of Production Research, 43(17), S. 3703 – 3743, ISSN 00207543.

Xu, X./Wang, L./Newman, S. T. (2011): Computer-aided process planning - A critical review of recent developments and future trends, in: International Journal of Computer Integrated Manufacturing, 24(1), S. 1 – 31, ISSN 0951192X.

Yang, Z./Shirwaiker, R. A./Wysk, R. A. (2011): Optimization of manufacturing operations using a new interactive computer-aided process planning program, S. Arena; Boeing; et al.; FedEx Ground; ISE – NC State University; The Hershey Company –, Reno, NV, United states.

Yu, J./Sun, S./Si, S./Yang, H./Wu, X. (2006): A study on the aviation manufacture cell scheduling based on adaptive ant colony algorithm, Bd. 532-533, S. 1060 – 1063, Xian, China, ISSN 02555476.

Yu, J./Wang, C. (2013): A max–min ant colony system for assembly sequence planning, in: The International Journal of Advanced Manufacturing Technology, S. 1–17.

Yusof, Y./Latif, K. (2014a): Survey on computer-aided process planning, in: The International Journal of Advanced Manufacturing Technology, ISSN 02683768.

Yusof, Y./Latif, K. (2014b): Survey on computer-aided process planning, in: International Journal of Advanced Manufacturing Technology, 75(1-4), S. 77 – 89, ISSN 02683768.

Zahid, T./Baqai, A. A. (2013): Multi-criteria optimization of process plans for reconfigurable manufacturing systems: An evolutionary approach, Bd. 2 B, S. ASME –, San Diego, CA, United states.

Zbiciak, R./Grabowik, C./Janik, W. (2015): An automation of design and modelling tasks in NX Siemens environment with original software - Cost module, Bd. 95, Mamaia, Romania, ISSN 17578981.

Zelen, M./Severo, N. C. (1964): Probability functions, in: Handbook of mathematical functions, 5, S. 925–995.

Zhang, F./Yan, X./Chen, Z./Yu, Z. (2010): A system of rapid process planning for machine features, Bd. 66 AISC, S. 3 – 14, Hong Kong, Hong kong, ISSN 18675662.

Zhang, L./Wong, T. (2015): An object-coding genetic algorithm for integrated process planning and scheduling, in: European Journal of Operational Research, 244(2), S. 434 – 444, ISSN 03772217.

Zhang, S./Yu, Z./Zhang, W./Yu, D./Zhang, D. (2015a): Distributed integration of process planning and scheduling using an enhanced genetic algorithm, in: International Journal of Innovative Computing, Information and Control, 11(5), S. 1587 – 1602, ISSN 13494198.

Zhang, W./Gen, M./Lin, L. (2008): An approach to computer-Aided process planning in distributed manufacturing using multiobjective genetic algorithm, Bd. 2, S. 1865 – 1874, Beijing, China.

Zhang, Y./Bernard, A./Valenzuela, J. M./Karunakaran, K. (2015b): Fast adaptive modeling method for build time estimation in Additive Manufacturing, in: CIRP Journal of Manufacturing Science and Technology, 10, S. 49 – 60, ISSN 17555817.

Zhang, Y./Luo, X./Zhang, H./Sutherland, J. W. (2014): A knowledge representation for unit manufacturing processes, in: International Journal of Advanced Manufacturing Technology, 73(5-8), S. 1011 – 1031, ISSN 02683768.

Zhao, X./Curran, R./Verhagen, W. J. C. (2014): Aircraft component multidisciplinary design optimization considering cost performance, S. Airbus; Boeing; Dunmore; et al; Georgia Innovation Aerospace; Lockheed Martin –, Atlanta, GA, United states.

Zheng, Y./Chen, M./Yao, X./Zhang, J. (2010): A manufacturing cost estimation method based on activity-based costing, in: 2010 International Conference on Mechanic Automation and Control Engineering, MACE2010, June 26, 2010 - June 28, 2010, 2010 International Conference on Mechanic Automation and Control Engineering, MACE2010, S. 3476–3479, IEEE Computer Society.

Zhou, X./Qiu, Y./Hua, G./Wang, H./Ruan, X. (2007): A feasible approach to the integration of CAD and CAPP, in: CAD Computer Aided Design, 39(4), S. 324 – 338, ISSN 00104485.

Zhu, D. F./Wang, H./Wang, S. B./Xu, J. X. (2012): Hybrid Genetic Algorithm-Based Production Planning for Steel-Making and Continuous Casting Process, in: Advanced Materials Research, 383, S. 1677–1683.

7 Appendix: MCOST and FoSCo additional information

7.1 Extracting dimensional data from NX: test case one example

All of the dimensions in the NX model which are required by MCOST must be named appropriately (appropriately meaning matching the names in the CIS Figure 7.3b) as shown in Figure 7.1.

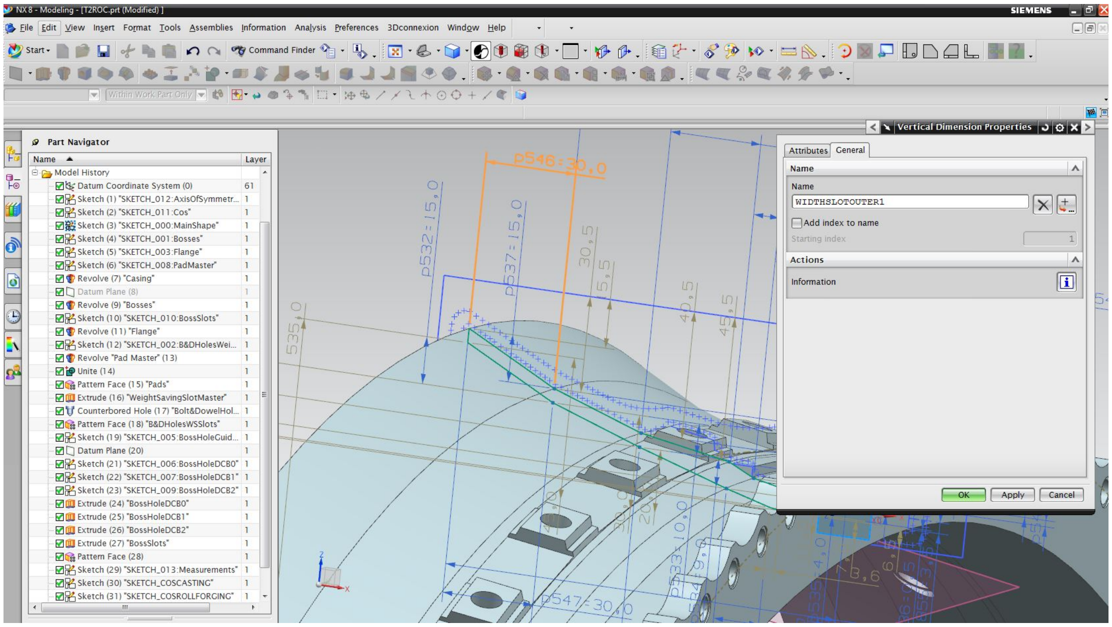


Figure 7.1: Exporting named NX dimensions from Test Case One

7.2 NX Output example test case one

An extract of the the output from NX (encoded in NX open - the NX API) is shown in Figure 7.2.

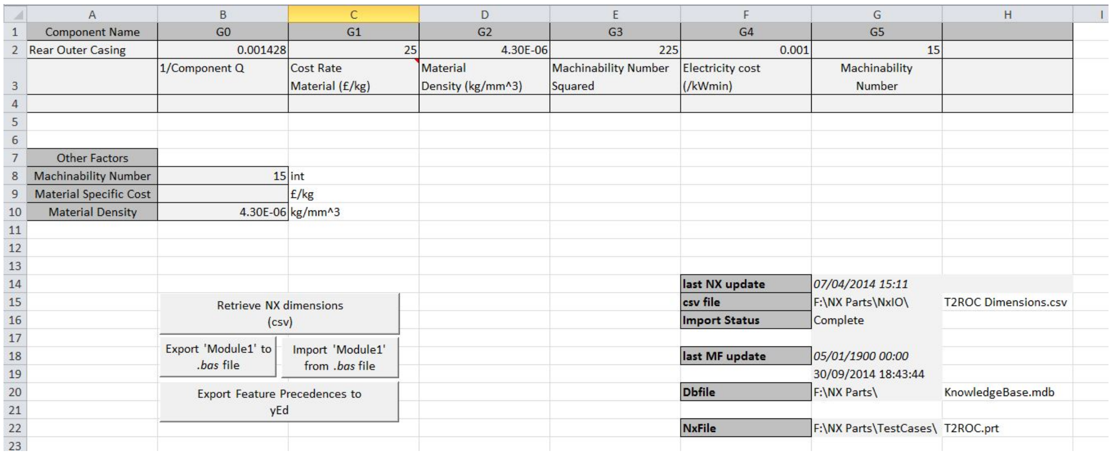
```
Expressions
Identifier|Expression|ComputedValue
NumberOfStatorVaneBosses|51|51
BossHolesCB1Depth|4|4
AngleOfMasterBoltHole|180/NumberOfBoltDowelHoles|1.42857142857143
NumberOfPads|10|10
BDHolesDiameter|8|8
NumberOfBoltDowelHoles|126|126
BossHolesCB2Depth|2|2
BDHolesCbDepth|0.5|0.5
BDHolesCbDiameter|10|10
PadAngularLength|7|7
Dimensions
Name|UniqueIdentifier|ComputedSize|MainTextLine|DualTextLine|ToleranceType|LowerTolerance|UpperTolerance
|HorizontalDimension 46875|15|p532=15,0|No secondary text lines|None||
 --- Removed lines here ---
|HorizontalDimension 46893|2|p541=2,0|No secondary text lines|None||
WIDTHSLOTOUTER5|VerticalDimension 35335|10|p542=10,0|No secondary text lines|None||
WIDTHSLOTOUTER4|VerticalDimension 35346|8.579|p543=8,6|No secondary text lines|None||
WIDTHSLOTOUTER3|VerticalDimension 35347|30|p544=30,0|No secondary text lines|None||
WIDTHSLOTOUTER2|VerticalDimension 35348|30|p545=30,0|No secondary text lines|None||
WIDTHSLOTOUTER1|VerticalDimension 35334|30|p546=30,0|No secondary text lines|None||
WIDTHSLOTINNER1|VerticalDimension 35349|30|p547=30,0|No secondary text lines|None||
WIDTHSLOTINNER2|VerticalDimension 35345|30|p548=30,0|No secondary text lines|None||
WIDTHSLOTINNER3|VerticalDimension 35344|30|p549=30,0|No secondary text lines|None||
WIDTHSLOTINNER4|VerticalDimension 35343|15|p550=15,0|No secondary text lines|None||
|PerpendicularDimension 47232|3.5|p855=3,5|No secondary text lines|None||
|PerpendicularDimension 35391|3.5|p554=3,5|No secondary text lines|None||
|PerpendicularDimension 35389|130|p553=130,0|No secondary text lines|None||
WIDTHSLOTINNER5|VerticalDimension 35332|3.579|3,6|No secondary text lines|None||
SLOTINNERRAD1|HorizontalDimension 46923|535|535,0|No secondary text lines|None||
SLOTINNERRAD2|HorizontalDimension 46925|520|520,0|No secondary text lines|None||
```

Figure 7.2: The export of the named dimensions from NX - this data can be read into the component interface spreadsheet

7.3 Component Interface Spreadsheet (Test case one)

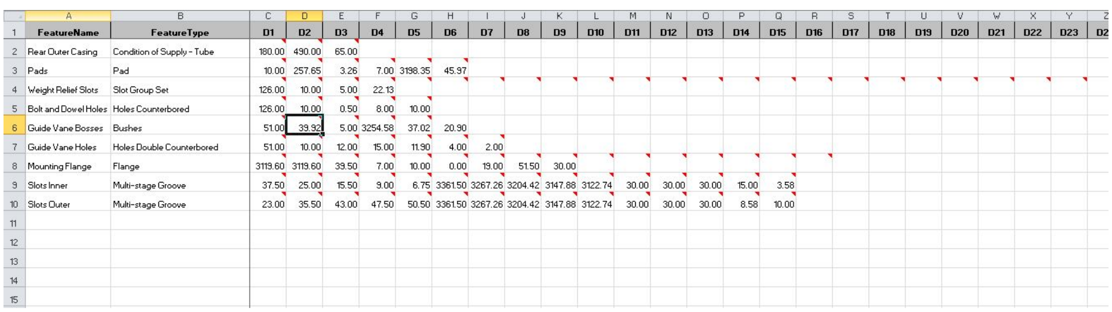
The component interface spreadsheet is the key part of the connection between NX and MCOST. It is comprised of multiple sheets, some related to the design features and general information about the component, which are pre-populated or populated during design of the component family. Other sheets are representations of the elemental features, they are populated by MCOST, and only used for debugging or informative purposes. There are VBA macros in the spreadsheet, which allow the import and synthesis of data contained in the Dimensions file (Figure 7.2).

- General information regarding the part is shown in Figure 7.3a.
- Design feature dimensions and tolerances are stored in the sheet shown in Figure 7.3b.
- Design feature precedences are shown populated in Figure 7.3c.



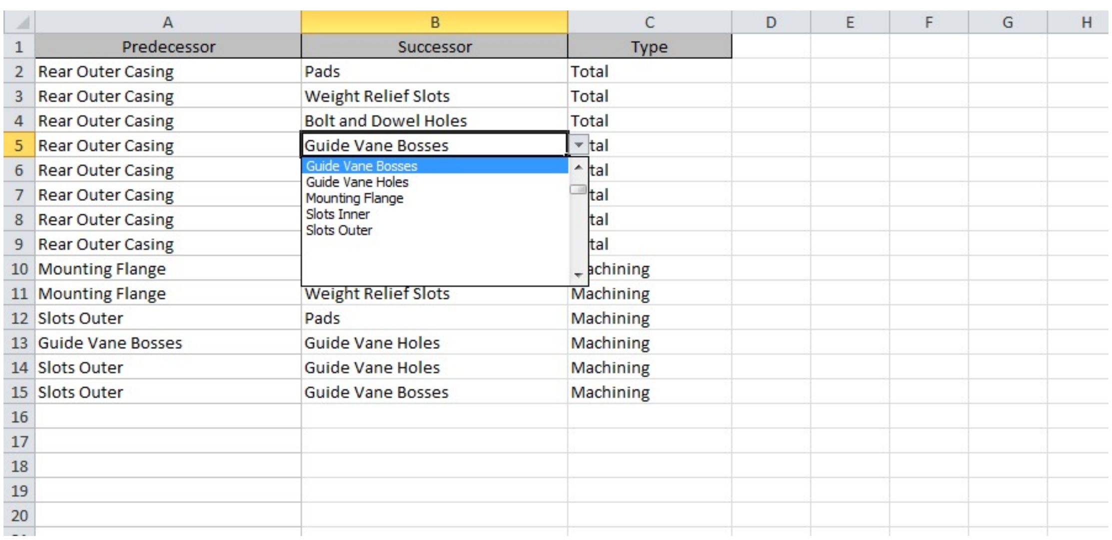
	A	B	C	D	E	F	G	H
1	Component Name	G0	G1	G2	G3	G4	G5	
2	Rear Outer Casing	0.001428	25	4.30E-06	225	0.001	15	
3		1/Component Q	Cost Rate Material (£/kg)	Material Density (kg/mm^3)	Machinability Number Squared	Electricity cost (/kWmin)	Machinability Number	
7	Other Factors							
8	Machinability Number	15	int					
9	Material Specific Cost		£/kg					
10	Material Density	4.30E-06	kg/mm^3					
14						last NX update	07/04/2014 15:11	
15		Retrieve NX dimensions (csv)				csv file	F:\NX Parts\NxIO\	T2ROC Dimensions.csv
16						Import Status	Complete	
18		Export 'Module1' to .bas file	Import 'Module1' from .bas file			last MF update	05/01/1900 00:00	
19							30/09/2014 18:43:44	
20		Export Feature Precedences to yEd				Dbfile	F:\NX Parts\	KnowledgeBase.mdb
22						NxFile	F:\NX Parts\TestCases\	T2ROC.prt

(a) Component data



	FeatureName	FeatureType	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D2
2	Rear Outer Casing	Condition of Supply - Tube	180.00	490.00	65.00																					
3	Pads	Pad	10.00	257.65	3.26	7.00	3198.35	45.97																		
4	Weight Relief Slots	Slot Group Set	126.00	10.00	5.00	22.13																				
5	Bolt and Dowel Holes	Holes Counterbored	126.00	10.00	0.50	8.00	10.00																			
6	Guide Vane Bosses	Bushes	51.00	39.92	5.00	3254.58	37.02	20.90																		
7	Guide Vane Holes	Holes Double Counterbored	51.00	10.00	12.00	15.00	11.90	4.00	2.00																	
8	Mounting Flange	Flange	3119.60	3119.60	39.50	7.00	10.00	0.00	19.00	51.50	30.00															
9	Slots Inner	Multi-stage Groove	37.50	25.00	15.50	9.00	6.75	3361.50	3267.26	3204.42	3147.88	3122.74	30.00	30.00	30.00	15.00	3.58									
10	Slots Outer	Multi-stage Groove	23.00	35.50	43.00	47.50	50.50	3361.50	3267.26	3204.42	3147.88	3122.74	30.00	30.00	30.00	8.58	10.00									

(b) Design Features



	Predecessor	Successor	Type
2	Rear Outer Casing	Pads	Total
3	Rear Outer Casing	Weight Relief Slots	Total
4	Rear Outer Casing	Bolt and Dowel Holes	Total
5	Rear Outer Casing	Guide Vane Bosses	tal
6	Rear Outer Casing		tal
7	Rear Outer Casing		tal
8	Rear Outer Casing		tal
9	Rear Outer Casing		tal
10	Mounting Flange		achining
11	Mounting Flange	Weight Relief Slots	Machining
12	Slots Outer	Pads	Machining
13	Guide Vane Bosses	Guide Vane Holes	Machining
14	Slots Outer	Guide Vane Holes	Machining
15	Slots Outer	Guide Vane Bosses	Machining

(c) Design Features Precedences

Figure 7.3: The component interface spreadsheet for test case one

7.4 Finding the convergence point

Figure 7.4 shows what will occur when the combination of FMP line mesh spacing, the cone within which FMPs can be considered, and the distance from the FMP to the FSPs interact to disqualify any FMP from the distance calculations and therefore the force calculations. The easiest solution is a finer mesh on the FMP lines.

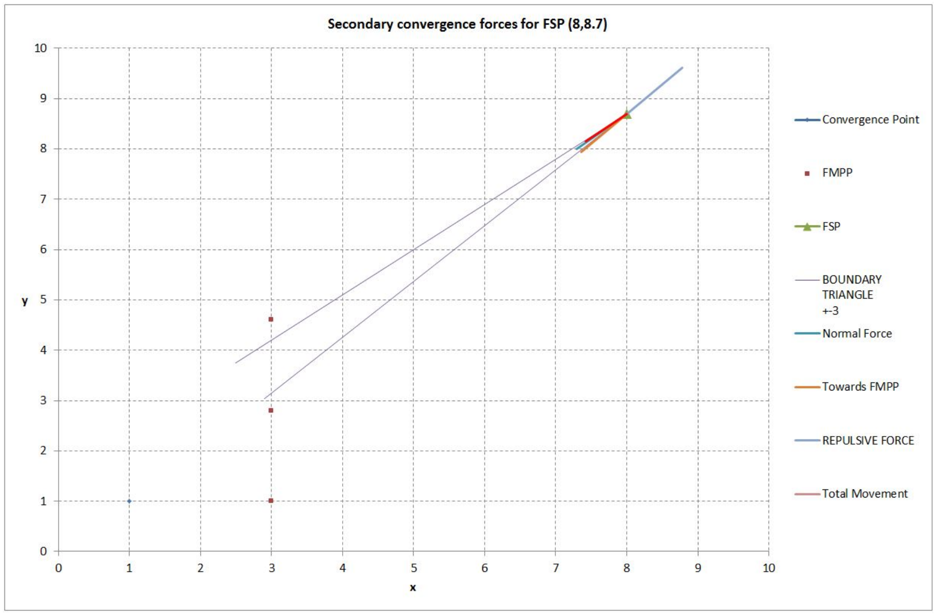
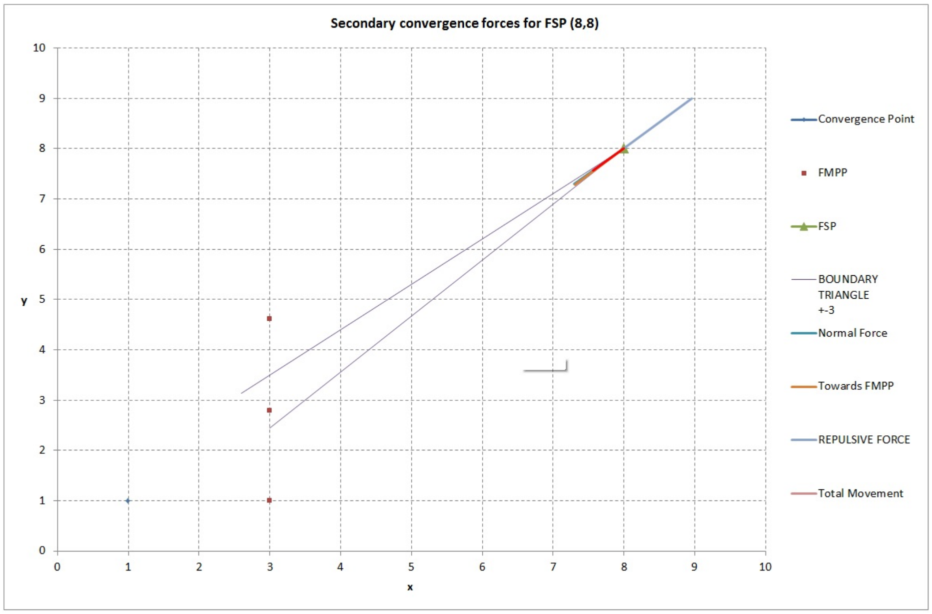


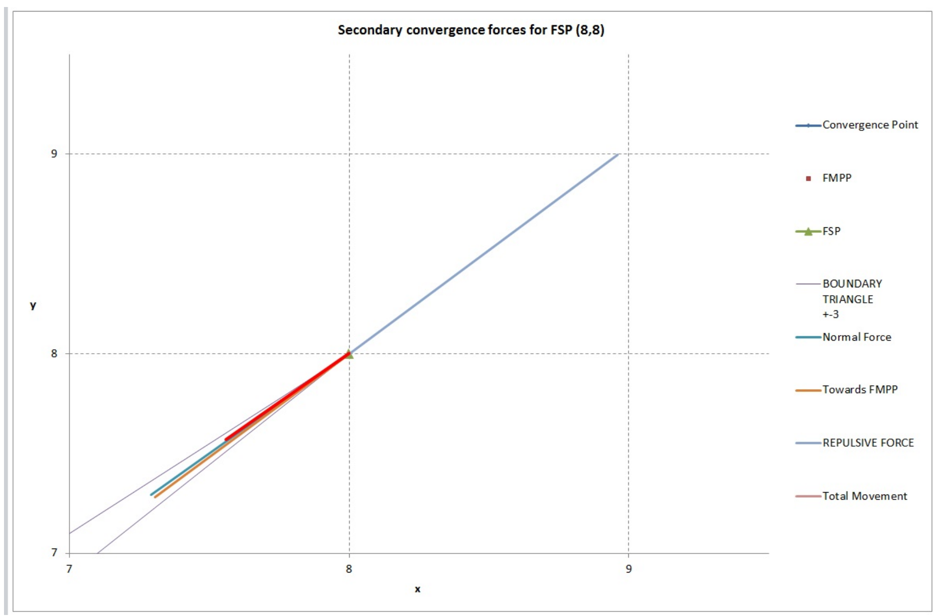
Figure 7.4: An insufficient number of FMPPs

7.5 Secondary convergence of FoSCo

The secondary convergence of FoSCo relies on a force balance, the four graphs below show (at two different scales) the force balance when the FSP is in two different positions (but the same theta value).

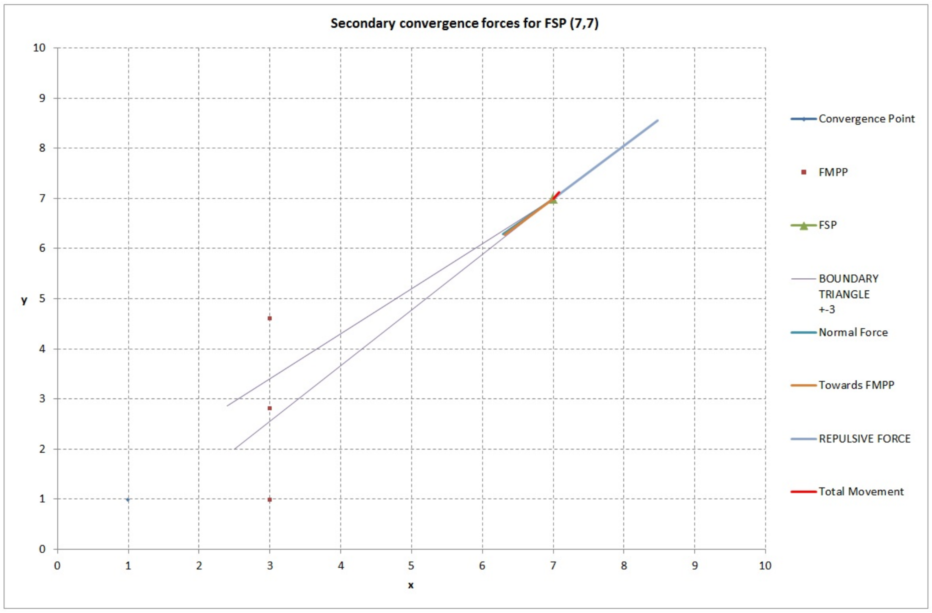


(a) Full graph

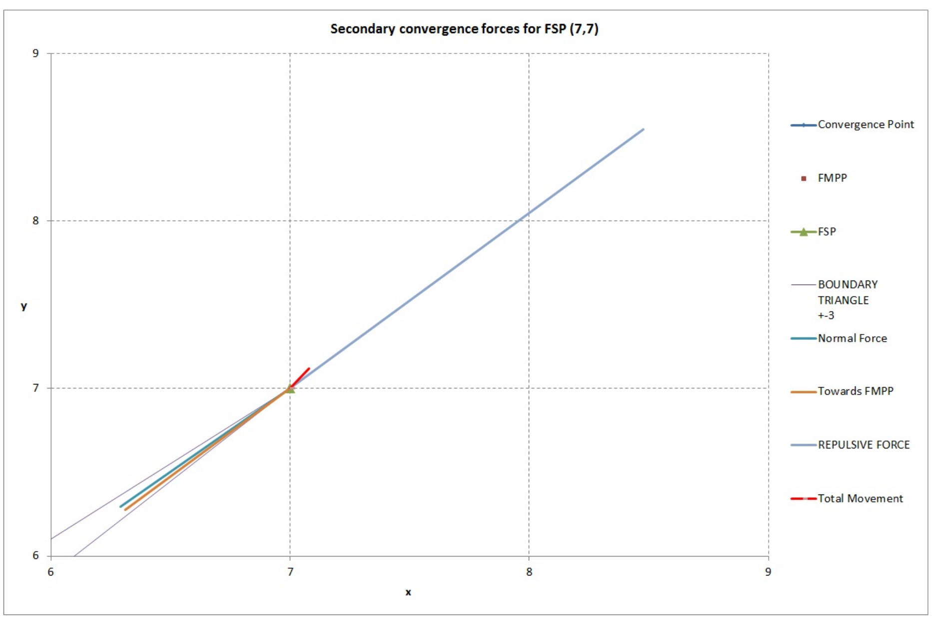


(b) Close view around FSP to show the three movement vectors

Figure 7.5: When FSP is at (8,8)



(a) Full graph



(b) Close view around FSP to show the three movement vectors

Figure 7.6: When FSP is at (7,7)

8 Appendix: FoSCo patent submission

This methodology, to be executed in a computer or similar device, is submitted for patent in the USA and the EU.

EU patent number 16180366.3 - 1954

US patent submission reference 5895 LMT

Note that references which are not ***figure*** references apply to the labled items on the figures in this appendix.

8.1 Abstract

Methods of estimating a precursor shape for a part There is disclosed a method of estimating a precursor shape (44, 45) for a part (2) suitable for manufacture via a process selected from a plurality of manufacturing processes. The method includes: generating a convergence line (20) within the confines of a final part shape (1); providing an array (18) of spaced-apart forming shape points (19) around the final part shape (1); converging said points (19) towards the convergence line (20) using a force-energy-balance technique via a movement algorithm; and applying a smoothing algorithm to the converged points (19) to create an estimated precursor shape (44, 45). The algorithms include coefficients specific to a predetermined one of said manufacturing processes. (To be accompanied by Figure 8.15 when published).

8.2 Claims

1. A method of estimating a precursor shape (44, 45) for a part (2) suitable for manufacture via a process selected from a plurality of manufacturing processes, the method including: generating a convergence line (20) within the confines of a final part shape (1); providing an array (18) of spaced-apart forming shape points (19) around the final part shape (1); converging said points (19) towards the convergence line (20) using a force-energy-balance technique via a movement algorithm; and applying a smoothing algorithm to the converged points (19) to create an estimated precursor shape (44, 45), wherein the algorithms include coefficients specific to a predetermined one of said manufacturing processes.

2. A method according to claim 1, further including selecting said predetermined manufacturing process from said plurality of manufacturing processes.

3. A method according to claim 1 or claim 2, further including defining said final part shape (1) prior to generation of the convergence line (20).

4. A method according to any preceding claim, wherein generation of the convergence line (20) is performed manually.

5. A method according to any one of claims 1 to 3, wherein generation of the convergence line (20) is performed automatically.

6. A method according to any preceding claim, wherein the ends (22, 24) of said convergence line (20) are proximate, but spaced inwardly of the final part shape (1).

7. A method according to any preceding claim, wherein said provision of said array of points (19) is one of: automated; manual; and partially automated.

8. A method according to any preceding claim, wherein each of said forming shape points (19) is assigned a respective angular value (ϑ) between 0° and 360°.

9. A method according to any preceding claim, wherein the angular value (ϑ) of each forming shape point (19) remains constant throughout said convergence and smoothing steps.

10. A method according to any preceding claim, wherein said final part shape (1) and said convergence line (20) are provided within a Cartesian coordinate system (16), such that the origin of said coordinate system falls within the final part shape (1).

11. A method according to any preceding claim, wherein at least some of said forming shape points (19) are moved relative to respective convergence points (24) during said convergence and smoothing, said convergence points (24) being located at discrete positions along said convergence line (24).

12. A method according to any preceding claim, wherein some of said forming shape points (19) are moved relative to the same, shared, convergence point (24), and wherein said shared convergence point (24) is coincident with an end (22, 23) of the convergence line (20).

13. A method according to claim 11 or claim 12, as dependent upon claim 8, wherein the convergence point (24), relative to which each forming shape point (19) is moved during convergence and smoothing, is determined solely in dependence on the angular value (ϑ) assigned to the respective forming shape point (19).

14. A method according to any preceding claim, further including application of an adaptive grid spacing to maintain said forming shape points (19) in approximately equi-spaced relation to one another during said convergence and smoothing.

15. A method according to any preceding claim, wherein convergence of said forming shape points (19) involves a primary convergence stage followed by a secondary convergence stage, wherein said primary convergence stage is performed independently of whichever one of said manufacturing processes is selected for manufacture of said part (2), and wherein

said secondary convergence stage involves application of said movement algorithm.

16. A method according to any preceding claim, wherein said movement algorithm includes a plurality of said coefficients (k1, k2, k3).

17. A method according to any preceding claim, wherein said smoothing algorithm includes a single said coefficient (k4).

18. A method according to any preceding claim, further including the creation of said final part shape (1) in a CAD package.

19. A method according to claim 18, further including the creation of said convergence line (20) in said CAD package.

20. A method according to any preceding claim, wherein the method is implemented on a computer.

21. An apparatus (47) comprising a processor (52) configured to perform the method of any one of claims 1 to 19.

22. A computer program (54) which, when read by a computer (47), causes performance of the method according to any one of claims 1 to 19.

23. A non-transistory computer readable storage medium (53) comprising computer readable instructions (54) which, when read by a computer (47), cause performance of the method according to any one of claims 1 to 19.

24. A method of estimating a precursor shape substantially as herein described with reference to the accompanying drawings.

25. An apparatus substantially as herein described with reference to the accompanying drawings.

26. A computer program substantially as herein described with reference to the accompanying drawings.

27. A non-transistory computer readable storage medium substantially as herein described with reference to the accompanying drawings.

8.3 Figures

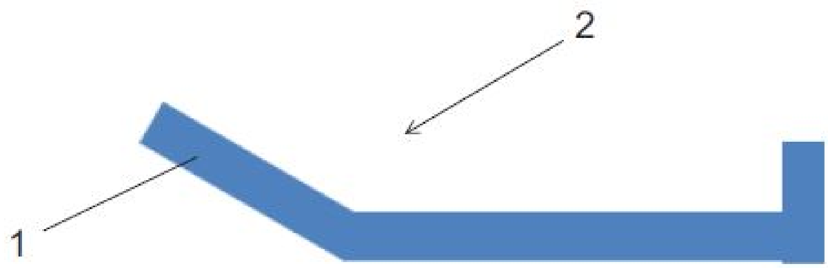


Figure 8.1: Schematic representation of the final two-dimensional shape of an exemplary part

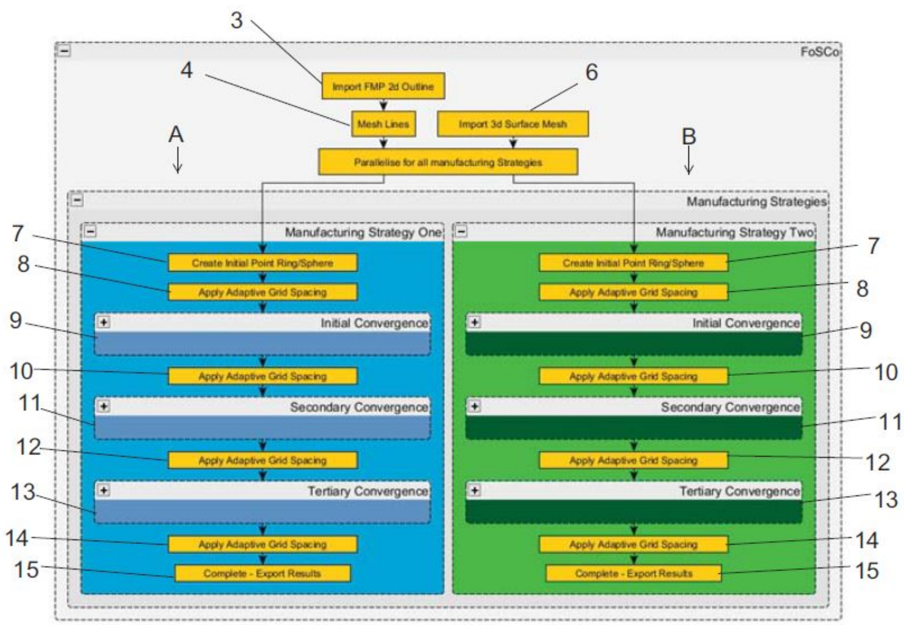


Figure 8.2: Flow diagram representing the overall architecture of the method

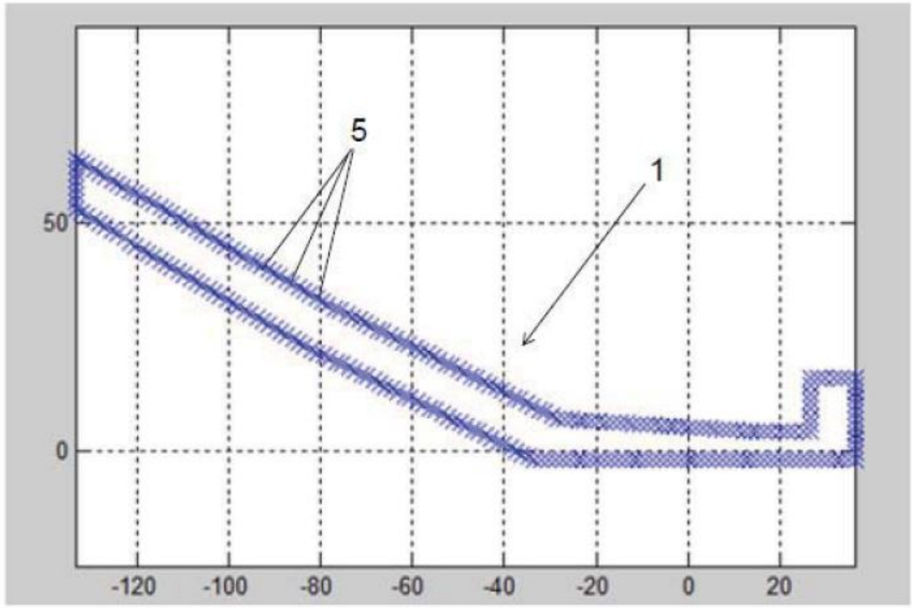


Figure 8.3: Linear mesh of the final part shape illustrated in figure 8.1

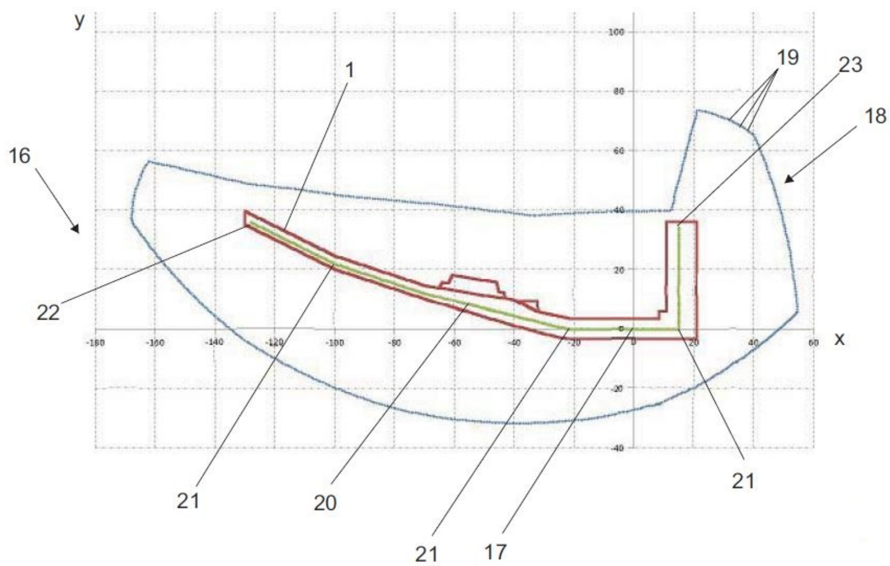


Figure 8.4: Schematic illustration showing a convergence line and starting positions of points (small radius at start)

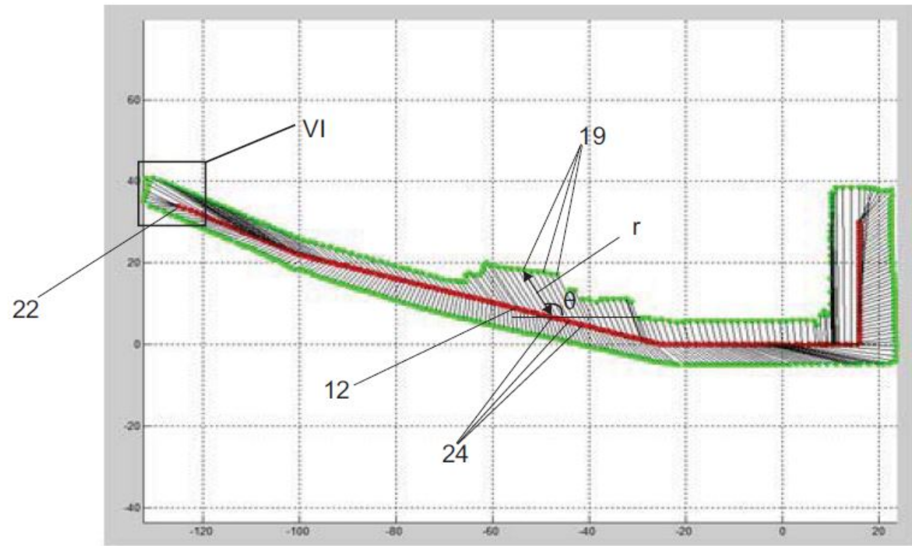


Figure 8.5: Schematic illustration showing the forming shape points at an intermediate position during convergence

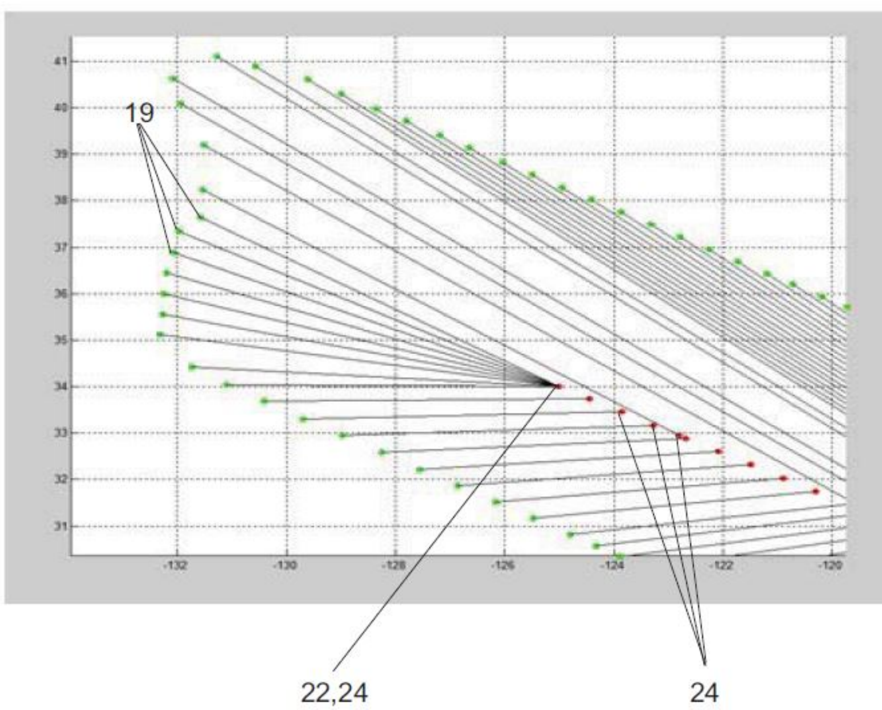


Figure 8.6: Enlarged view of the region VI denoted in figure 8.5

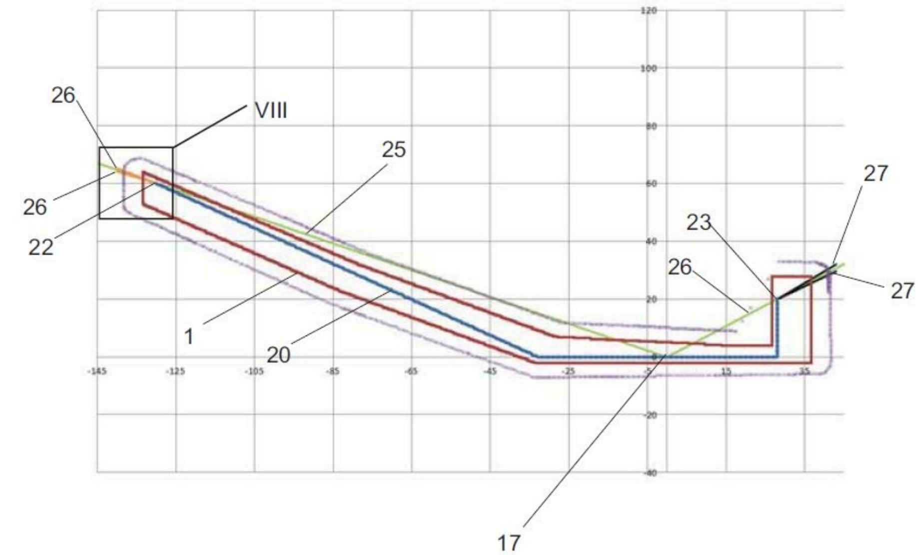


Figure 8.7: Schematic illustration showing principles involved in determining the convergence point

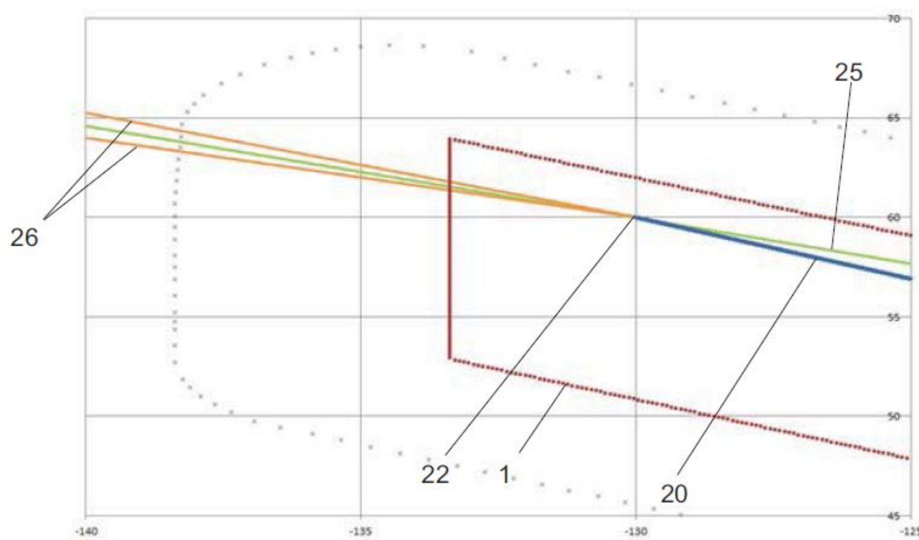


Figure 8.8: Enlarged view of the region VIII denoted in figure 8.7

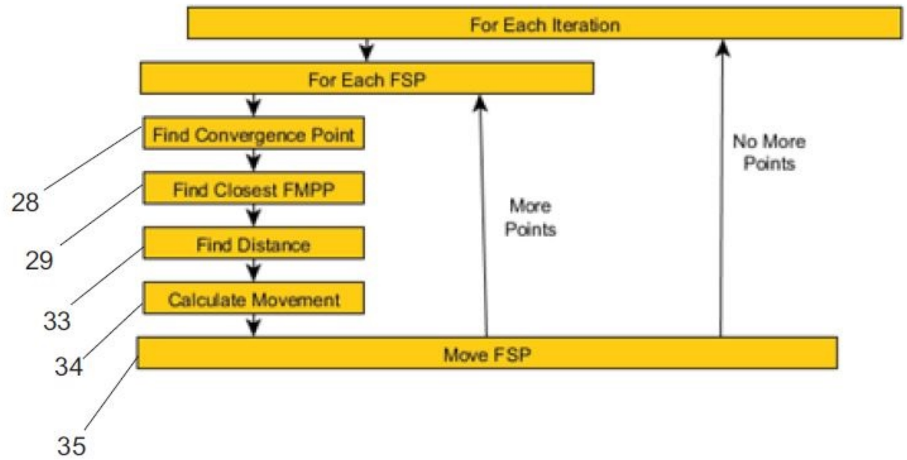


Figure 8.9: Flow diagram representing an iterative primary convergence process (optional)

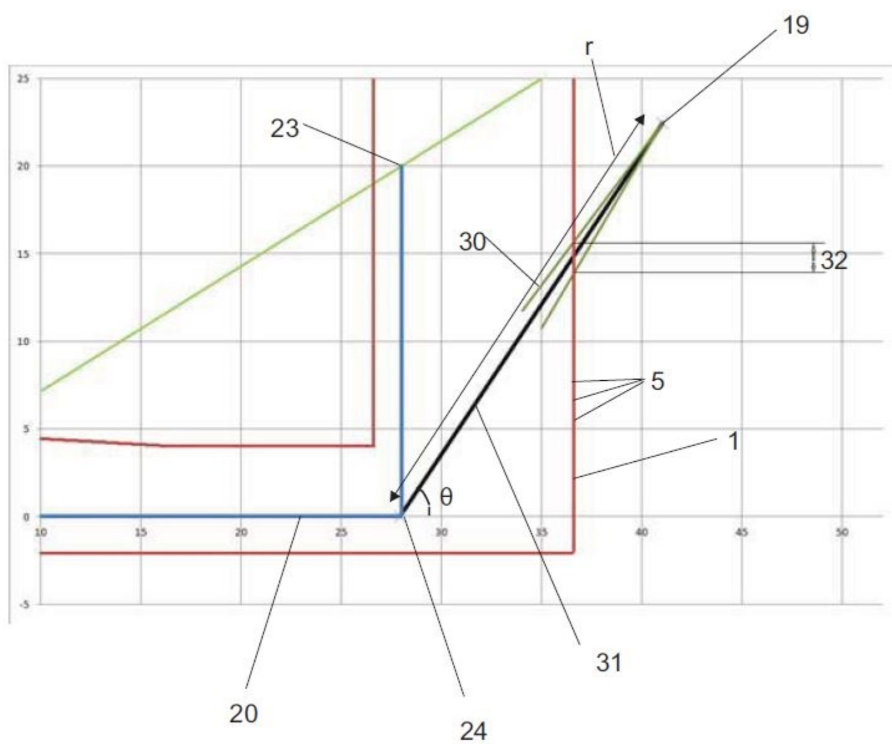


Figure 8.10: Schematic illustration showing principles involved in determining a distance to FMPP

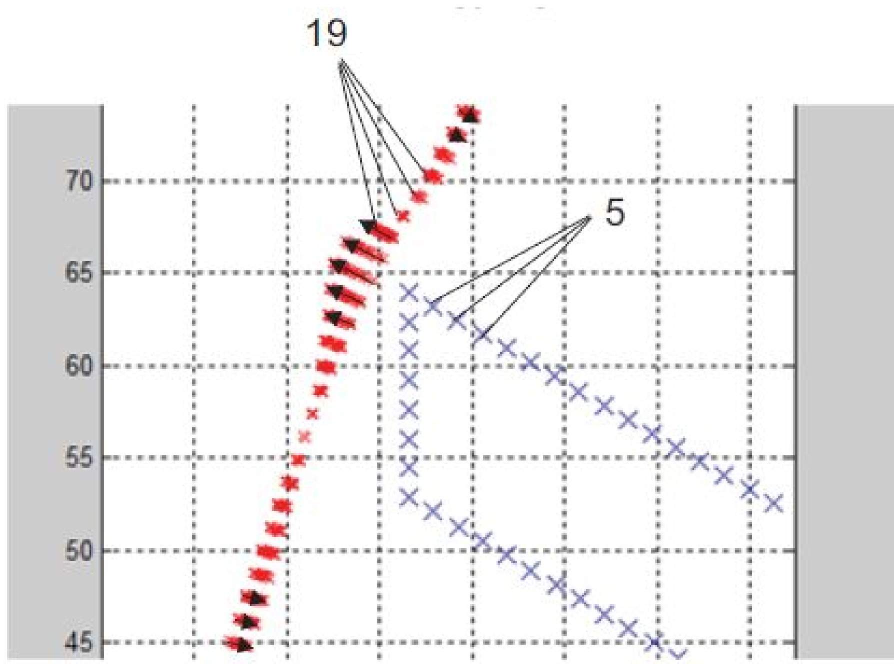


Figure 8.11: Schematic illustration, in enlarged view (secondary convergence)

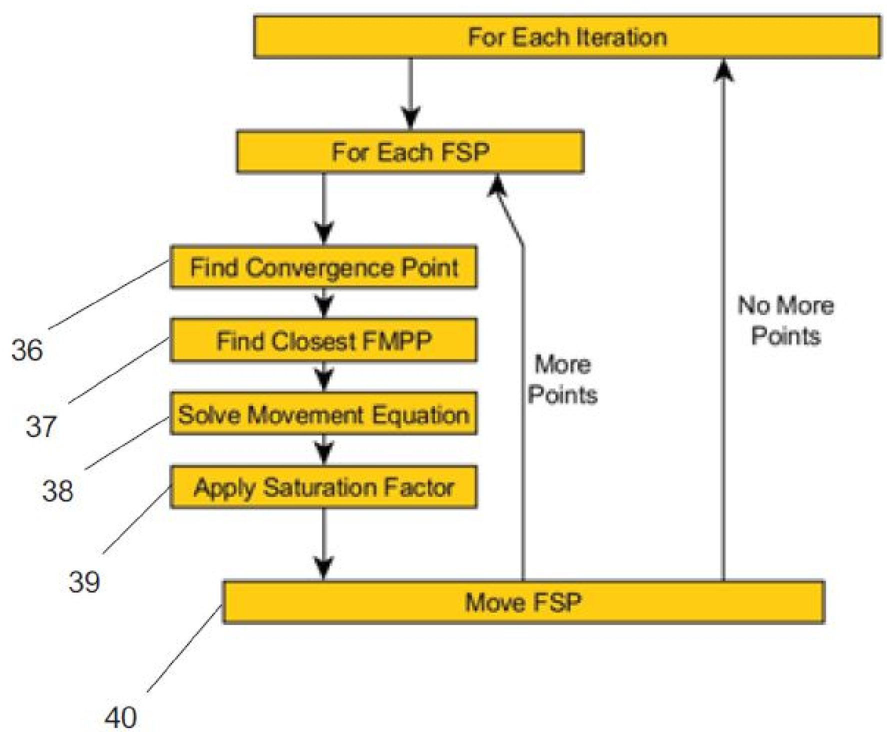


Figure 8.12: Flow diagram representing the secondary convergence process

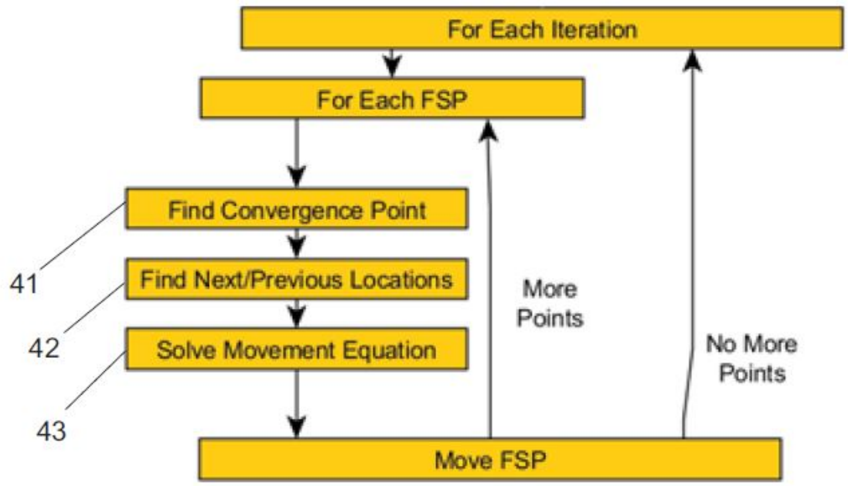


Figure 8.13: Flow diagram representing a tertiary convergence process

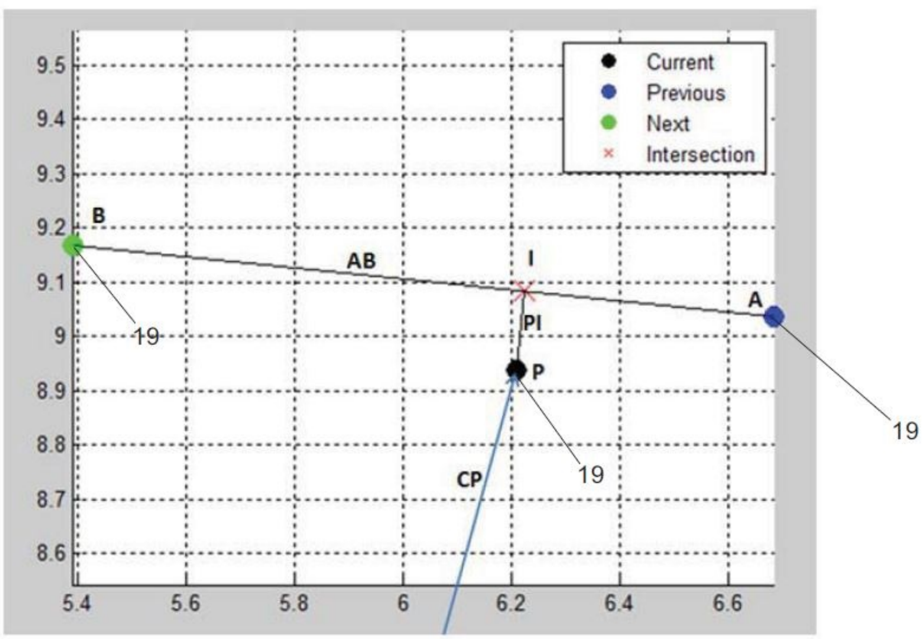


Figure 8.14: Tertiary convergence process in respect of an individual forming shape point

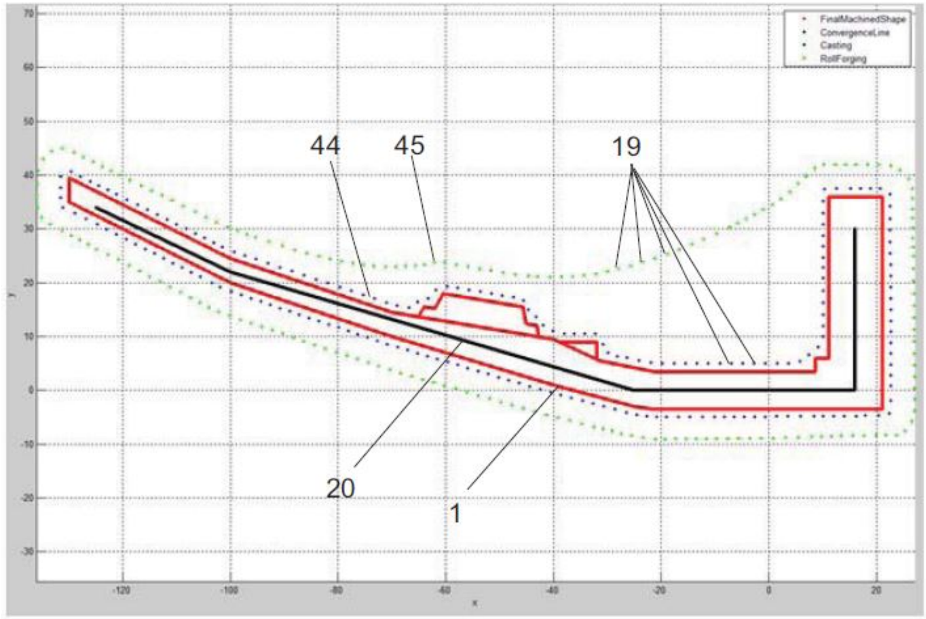


Figure 8.15: Schematic illustration showing alternative precursor shapes

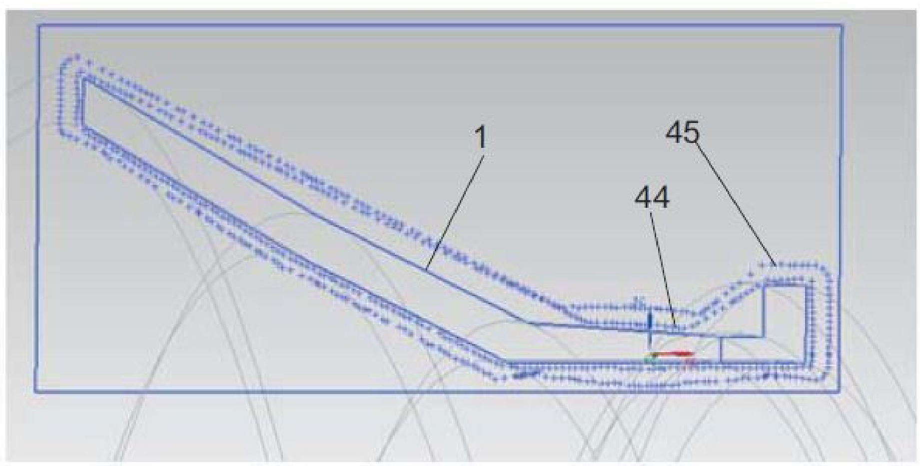


Figure 8.16: The resulting precursor shape imported into a CAD package

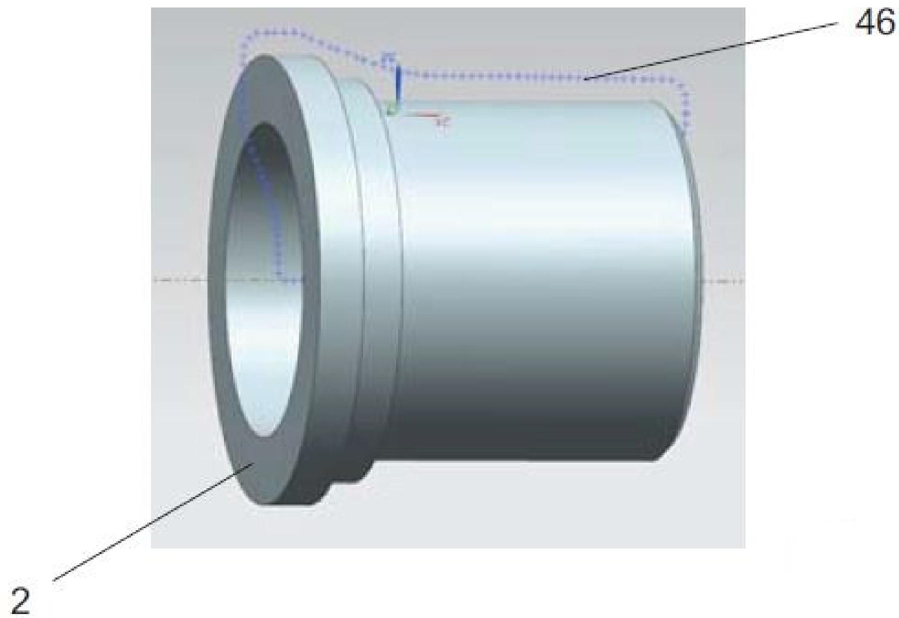


Figure 8.17: A component whose precursor shape may be estimated via this method

Figure 8.18: Schematic drawing showing a computer apparatus

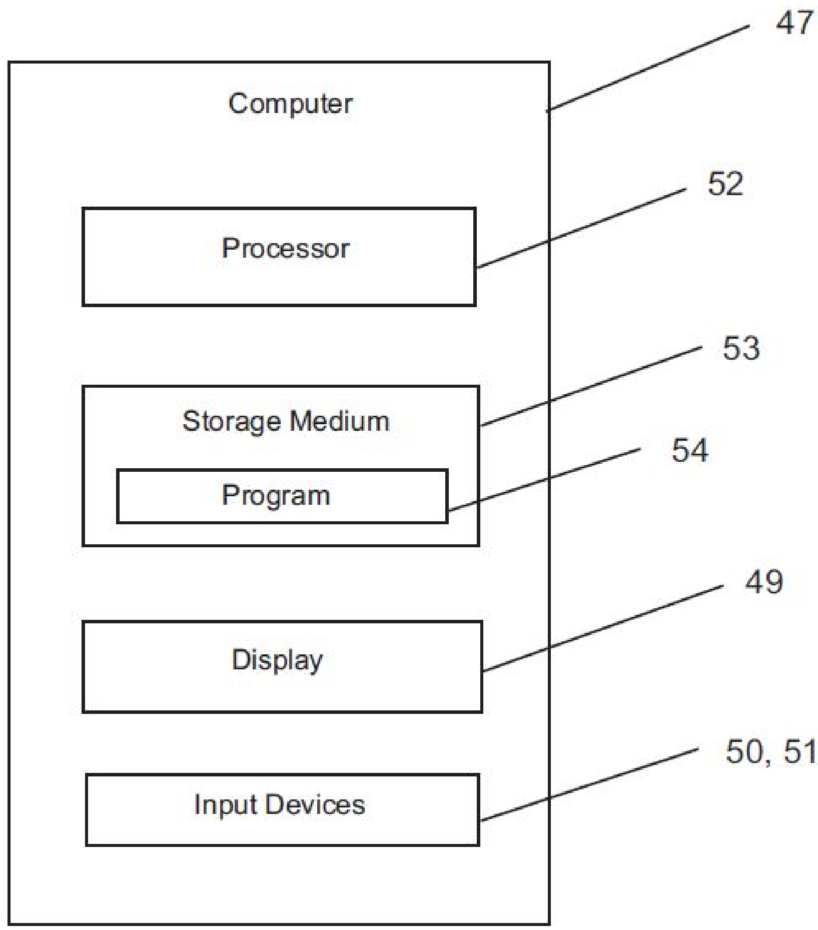


Figure 8.19: Schematic drawing of computer apparatus

8.4 Method

This specification relates to a method of estimating a precursor shape for a part to be manufactured, and more particularly relates to a method of estimating a precursor shape for a part suitable for manufacture via a process selected from a plurality of possible manufacturing processes.

Computer Aided Process Planning (CAPP) is a generic name for a diverse range of techniques which can be used to take design information for parts, such as engineering components, and create therefrom an appropriate manufacturing route in the form of a set of manufacturing operations which can be used to process raw material in order to produce a desired part. CAPP can be considered to represent an exercise in decision making. The level of human intervention, the extent of the freedom allowed to the computer, the accuracy of any evaluations or simulations the computer is required to run, the acceptable run-time, and the form and extent of the information required by the computer are matters for the CAPP programmer and other decision makers to decide.

The manufacturing methodology used for many intricate and/or high performance parts such as, for example, components of gas turbine engines or the like often involves the formation of a precursor volume via one of a number of possible forming manufacturing processes (e.g. casting and forging). The precursor part produced by the chosen manufacturing process will often be somewhat larger than the desired final shape of the part, and so further processing will then be required (e.g. chip forming operations) to achieve the final desired shape. As will be appreciated, the precursor shape which might be achievable by one manufacturing process will often be somewhat different to that which might be achievable by another manufacturing process.

As will therefore be appreciated, the manufacturing operations which are required to produce any given part will depend on the geometry of the finished part and the shape of the precursor part. It is therefore useful to be able to rapidly estimate the precursor shape which can be created by whatever forming process (e.g. casting or forging) is to be used. In order to optimise the process and cost modelling, it is important to be able to estimate a precursor shape automatically. It has been previously proposed to estimate a precursor shape via the use of a full rule-based technique, which has been found to be accurate and to require a mid-level of computational complexity. However, this technique is not generic to all part shapes and so requires the formation of a different set of rules for each different part. A different set of rules is also required for each manufacturing process.

It has also been previously proposed to create a precursor shape simply by adding an offset of a few millimetres around the final desired part shape, optionally including the use of a smoothing algorithm or taking account of sharp corners. However, these techniques are considered to be far from optimal.

According to a first aspect of the present proposal, there is provided a method of estimating a precursor shape for a part suitable for manufacture via a process selected from a plurality of manufacturing processes, the method including: generating a convergence line within the confines of a final part shape; providing an array of spaced-apart forming shape points around the final part shape; converging said points towards the convergence line using a force-energy-balance technique via a movement algorithm; and applying a smoothing algorithm to the converged points to create an estimated precursor shape, wherein the algorithms include coefficients specific to a predetermined one of said manufacturing processes.

The method may further include selecting said predetermined manufacturing process from said plurality of manufacturing processes.

The method may include defining said final part shape prior to generation of the convergence line.

Generation of the convergence line may be performed manually or automatically.

Optionally, the ends of said convergence line are proximate, but spaced inwardly of the final part shape.

Said provision of said array of points may be one of: automated; manual; and partially automated. Each of said forming shape points may be assigned a respective angular value (ϑ) between 0° and 360°. Optionally, the difference between the angular values (ϑ) of each neighbouring pair of points is equal.

The angular value (ϑ) of each forming shape point may remain constant throughout said convergence and smoothing steps.

Said final part shape and said convergence line may be provided within a Cartesian coordinate system, such that the origin of said coordinate system falls within the final part shape.

Optionally, at least some of said forming shape points are moved relative to respective convergence points during said convergence and smoothing, said convergence points being located at discrete positions along said convergence line.

Some of said forming shape points may be moved relative to the same, shared, convergence point, and wherein said shared convergence point is coincident with an end of the convergence line.

The convergence point relative to which each forming shape point is moved during convergence and smoothing may be determined solely in dependence on the angular value (ϑ) assigned to the respective forming shape point.

The method may further include application of an adaptive grid spacing to maintain said forming shape points in approximately equi-spaced relation to one another during said convergence and smoothing.

Convergence of said forming shape points may involve a primary convergence stage followed by a secondary convergence stage, wherein said primary convergence stage is performed independently of whichever one of said manufacturing processes is selected for manufacture of said part, and wherein said secondary convergence stage involves application of said movement algorithm.

Said movement algorithm may include a plurality of said coefficients.

Said smoothing algorithm may include a single said coefficient.

The method may further include the creation of said final part shape in a CAD package. The method may also further include the creation of said convergence line in said CAD package.

The method may be implemented on a computer.

According to a second aspect of the present proposal, there is provided an apparatus comprising a processor configured to perform the method of the first aspect.

According to a third aspect of the present proposal, there is provided a computer program which, when read by a computer, causes performance of the method according to the first aspect.

According to a fourth aspect of the present proposal, there is provided a non-transistory computer readable storage medium comprising computer readable instructions which, when read by a computer, cause performance of the method according to the first aspect.

So that the proposals may be more readily understood, and so that further features thereof may be appreciated, aspects of the proposals will now be described by way of example with reference to the accompanying drawings in which:

Figure 8.1 is a schematic representation of the final two-dimensional shape of an exemplary part whose precursor shape may be estimated via the proposed method;

Figure 8.2 is a flow diagram representing the overall architecture of the method;

Figure 8.3 is a schematic illustration showing a linear mesh of the final part shape illustrated in Figure 8.1, the mesh comprising a plurality of equi-spaced points identified by crosses;

Figure 8.4 is a schematic illustration showing a convergence line provided within the confines of the final part shape and an array of spaced-apart forming shape points arranged around the meshed final part shape, prior to any convergence;

Figure 8.5 is a schematic illustration showing the forming shape points at intermediate positions during convergence, and which shows the manner in which the points are moved towards different positions along the convergence line;

Figure 8.6 is an enlarged view of the region VI denoted in Figure 8.5, showing in more detail the region of an end of the convergence line; Figure 8.7 is a schematic illustration showing principles involved in determining the convergence point on the convergence line to which each forming shape point will move during convergence;

Figure 8.8 is an enlarged view of the region VIII denoted in Figure 8.7, showing in more detail the region of an end of the convergence line;

Figure 8.9 is a flow diagram representing an iterative primary convergence process which may be carried out on the forming shape points so that they are moved towards the convergence line;

Figure 8.10 is a schematic illustration showing principles involved in determining a distance pertaining to the primary convergence process shown in Figure 8.9

Figure 8.11 is a schematic illustration, in enlarged view, depicting characteristics of an iterative secondary convergence process to move the forming shape points towards or away from the convergence line;

Figure 8.12 is a flow diagram representing the secondary convergence process; Figure 8.13 is a flow diagram representing a tertiary convergence process;

Figure 8.144 illustrates details of the tertiary convergence process in respect of an individual forming shape point;

Figure 8.15 is a schematic illustration showing alternative precursor shapes for a part to be produced by casting and roll forging;

Figure 8.16 shows the resulting precursor shape imported into a CAD package as a sketch;

Figure 8.17 shows a component whose two-dimensional shape may be estimated as a precursor shape via the proposed method;

Figure 8.18 is a schematic drawing showing a computer apparatus which may be configured to perform the method; and

Figure 8.19 is a schematic drawing showing elements of the computer apparatus illustrated in Figure 8.18.

Turning now to consider the drawings in more detail, Figure 8.1 illustrates the final machined part shape 1 of an exemplary part 2 whose precursor shape may be estimated using the method of the present proposal, as will be described in detail below. It is to be noted that the final part shape 1 is thus the final target shape of the part 2 which is to be produced by appropriate forming and machining processes. The particular part 2 whose final part shape 1 is illustrated in Figure 8.1 is shown in simplified form for reasons of clarity but may, for example, be an annular casing component for a gas turbine engine. It is to be noted that Figure 8.1 represents the two-dimensional final part shape 1 (in radial cross section) of the part 2. The final part shape 1 will typically be created in a Computer Aided Design (CAD) software package in a manner known per se.

Figure 8.2 illustrates schematically the overall architecture of a method in accordance with an embodiment of the present proposal. In brief outline, if the final machined part shape 1 which is created in CAD is two-dimensional as illustrated in Figure 8.1, then its outline is imported into a computer program configured to implement the method, as denoted at 3 in Figure 8.2, whereupon the outline is meshed (as denoted at 4) to create a plurality of equispaced final machined part points 5 along the outline of the final part shape 1, as shown schematically in Figure 8.3 (each final part point 5 being defined by the intersection of a respective cross). However, it is possible instead to mesh the outline of the final part shape 2 directly within the CAD package prior to import, particularly in the case of a three dimensional surface shape, as denoted at 6 in Figure 8.2.

Following creation and/or importation of the meshed final part shape outline, the method may then be applied in dependence on the particular initial manufacturing process selected to form the precursor part prior to further finishing. Figure 8.2 thus illustrates a bifurcation in the process flow into two possible alternative branches A, B, each of which is representative of the method pertaining to a respective manufacturing process. For example, branch A could be considered to represent the method applicable to estimate a precursor shape for a forging process, whilst branch B could be considered to represent the method applicable to create a precursor shape for a casting process. It is to be noted, however, that more than two manufacturing processes could be provided for. It is also to be noted that the subsequent method steps within each branch A, B of the method are generally identical in many respects, and are based on use of the same processes, techniques and algorithms, regardless of which manufacturing process is to be used. As will be described in more detail, however, the implementation of the method will be based on different parameters depending on which manufacturing process is to be used, whilst the algorithms use by the method will remain the same for all possible manufacturing processes.

Branches A and B thus each comprise a series of stages and processes, some of which are optional as will be described in more detail hereinafter, namely:

- the creation of an array of spaced-apart forming shape points around the meshed final part shape outline, as denoted at 7;
- the application of adaptive grid spacing to the array of forming shape points in order to ensure an even distribution of the points, as denoted at 8;

- primary convergence of the forming shape points, performed in an iterative manner, generally towards the final part shape 1, as denoted at 9,
- further application of adaptive grid spacing to the converged forming shape points to ensure even distribution, as denoted at 10;
- secondary convergence of the forming shape points, performed in an iterative manner, generally towards the final part shape 1, as denoted at 11;
- further application of adaptive grid spacing to the converged forming shape points to ensure even distribution, as denoted at 12;
- tertiary convergence, involving the use of a smoothing algorithm and again performed in an iterative manner , as denoted at 13;
- further application of adaptive grid spacing to the converged forming shape points, as denoted at 14; and
- export of the resulting precursor shape to a CAD package, as denoted at 15, for subsequent automatic creation of reference dimensions to inform a CAPP system how much machining will be necessary to achieve the final part shape 1.

Turning now to consider Figure 8.4, the meshed final part shape 1 (illustrated in more detail than in Figure 8.3) is shown in a native Cartesian coordinate system 16 comprising a horizontal x-axis and a vertical y-axis. The final part shape 1 is located over the origin 17 of the coordinate system. An array 18 of spaced-apart forming shape points 19 is then created around the meshed final part shape 1. In Figure 8.4, each forming shape point 19 is denoted by a respective cross, the crosses for each respective point 19 being arranged in a closed contour around and entirely encompassing the meshed final part shape 1 in spaced relation thereto. The forming shape points 19 are thus distributed in the native coordinate system, such that each individual point 19 will have a respective unique (x, y) location within the coordinate system. As will be appreciated, there are a very large number of forming shape points 19 provided in this manner, such that the resolution of Figure 8.4 makes it appear that their crosses coalesce to form a thick line. The particular shape of the closed contour array will depend (indirectly) on the final part shape 1, as will be described in more detail hereinafter. It is also to be noted that Figure 8.4 depicts the forming shape points 19 in respective starting positions that are considerably closer to the final part shape 1 than would actually be the case in practice, for illustrative purposes. The manner in which the actual starting positions of the forming shape points 19 are determined, and thus also the shape of the initial array 18, will become apparent hereinafter.

As will also be explained in more detail below, the method involves iterative convergence of the forming shape points 19, generally towards the final part shape 1, such that upon completion of the convergence process the locations of the forming shape points 19 will collectively define the outline of a precursor shape for the part which is to be produced. As already indicated above, convergence of the forming shape points 19 may involve a number of discrete stages. For example, in the embodiment of the method illustrated schematically in Figure 8.2, the overall convergence process may involve primary convergence, followed by secondary convergence, which is then followed by tertiary convergence via a smoothing algorithm. However, an important aspect of the method is that the forming shape points 19 do not all converge towards the same convergence point. It is this aspect of the method which allows the precursor shape created to be as close as possible to the final part shape 1 in a near net shape manufacturing process.

The method therefore also involves the creation of a convergence line relative to which the forming shape points 19 will each be converged. An exemplary convergence line 20 is illustrated in Figure 8.4. It is to be noted that the convergence line 20 is entirely enveloped by the final part shape 1. The convergence line 20 may thus include a number of bends or corners 21 to ensure that it remains within the final part shape 1 at all points along its length.

It may alternatively be possible for the convergence line 20 to be curved. The ends 22, 23 of the convergence line 20 should not be too close to each other relative to the origin 17 of the coordinate system 16.

Returning now to consider the forming shape points 19, each point will be assigned a respective and unique fixed angular theta value (θ) of between 0° and 360°, as measured counter-clockwise from a notional horizontal line parallel to the x-axis of the coordinate system 16. For illustrative purposes, an example may be considered in which the closed contour array 18 comprises one-hundred forming shape points 19 (although in practice it is envisaged that many more forming shape points will be used for reasons of accuracy). In such an example: the first forming shape point 19 would thus be assigned a theta value calculated as 1 x (360/100), so $\theta_1 = 3.6°$; the second forming shape point 19 would be assigned a theta value calculated as 2 x (360/100), so $\theta_2 = 7.2°$; the third forming shape point would be assigned a theta value as 3 x (360/100), so $\theta_3 = 10.8°$; and so on. The angular difference between the theta values of each neighbouring pair of forming shape points 19 will thus be equal throughout the full 360° of the closed contour array 18.

As indicated above, convergence of the forming shape points 19 will involve movement of the points relative to the convergence line 20. In this regard, it is to be noted that many of the points 19 will be moved relative to respective and unique convergence points along the convergence line 20, although it is possible that some of the points 19 around the ends 22, 23 of the convergence line 20 will all need to be moved relative to the same end point. This principle can be seen in Figure 8.5 which illustrates the forming shape points 19 at a point in time which is part-way through the convergence process. In more detail, each forming shape point is shown connected to its convergence point 24 on the convergence line 20 by a notional line having a length r measured from the respective convergence point 24, and which makes an angle ϑ relative to a horizontal line through the convergence point 24, where ϑ is the fixed theta value assigned to the respective forming shape point 19. The position of each forming shape point 19 relative to its convergence point 24 may thus be defined by polar coordinates (r, ϑ). During convergence, as will be explained in more detail below, the theta value ϑ of each forming shape point 19 will remain fixed, but the point's r value will change, such that each forming shape point 19 will be moved along its aforementioned notional line. Figure 8.6 illustrates the region of the end point 22 of the convergence line 20 shown in Figure 8.6 in more detail, and in particular illustrates how several of the forming shape points 19 around the end of the convergence line will all be converged relative to the same convergence point 24 (coincident with the end of the convergence line 20) despite evertheless still having different fixed theta values.

As will be appreciated, it is therefore necessary to define a convergence point 24 on the convergence line 20 for each forming shape point 19. In this regard, it is to be noted that the location along the convergence line 20 of the convergence point 24 for any given forming shape point 19 will depend solely on the theta value (ϑ) assigned to that forming shape point. The process by which this is done will now be described below in more detail.

When the convergence line 20 is constructed, its end points 22, 23 are each assigned a range of angular values, based on respective lines extending from the origin 17 of the coordinate system 16 and passing through each end point 22, 23 as illustrated in figures 7 and 8, of plus and minus 3°. In figures 7 and 8, the extent of these angular values at each end 22, 23 of the convergence line 20 are denoted by respective pairs of boundary lines 26, 27. The interbound arc angle between the boundary lines 26, 27 of each pair will thus be 6°.

Any forming shape point 19 falling between either pair of boundary lines 26, 27 will be converged relative to a convergence point 24 coincident with the respective end point 22, 23 of the convergence line 20.

So, taking the example illustrated in figures 7 and 8, it will be noted that the first end point 22 of the convergence line 20 has Cartesian coordinates of (-130, 60), whilst the second end point has coordinates of (28, 20). The angles from each of these end points, as measured from the positive x-axis of the coordinate system 16 can thus be calculated via trigonometry to be 155.2249° and 35.5378° respectively. Thus, any forming shape points which have a theta value (θ) of 155.2249 $\pm$ 3 will take the first end point 22 as their (shared) convergence point, whilst any forming shape points 19 which have a theta value (θ) of 35.5378 $\pm$ 3 will take the second end point 23 as their (shared) convergence point.

All of the other forming shape points 19 will lie between one of the boundary lines 26 at the first end 22 of the convergence line 20 and one of the boundary lines 27 at the second end 23 of the convergence line, and their convergence points 24 are determined in dependence on how far between the boundary lines, in an angular sense, the forming shape points 19 lie.

Having regard to Figure 8.7, the upper boundary line 27 of the start-point ("sp") end 23 of the convergence line has an angle denoted spmax which in this specific example will be

35.5378+3 = 38.5378. Similarly, the upper boundary line 26 at the end-point ("ep") 22 of the convergence line 20 has an angle denoted epmin which in this specific example will be 155.2249-3 = 152.2249.

So for forming shape points located generally above the convergence line 20 shown in Figure 8.7, or more accurately between spmax and epmin , the position of their respective convergence points 24, as measured along the convergence line 20 from the start-point end 23, may be expressed as a percentage of the total length of the convergence line 20 by the expression:

$$100 \cdot \left(\left(\theta - \mathrm{sp}_{\mathrm{max}}\right) / \mathrm{A}_{\mathrm{IB}}\right) \tag{8.1}$$

Where AIB is the interbound arc angle: (i.e. $\mathrm{ep}_{\mathrm{min}}$-$\mathrm{sp}_{\mathrm{max}}$ or (155.2249-3)-(35.5378+3)=113.6871 degrees in the specific example illustrated).

So, taking an example of a forming shape point 19 having a theta value (ϑ) of 45°, and the specific convergence line 20 illustrated in Figure 8.7 and thus having the above stated spmax and $\mathrm{ep}_{\mathrm{min}}$ values, then this expression will give a percentage value of 5.68%, such that the forming shape point's convergence point 24 will be located at a position on the convergence line 20 and 5.68% of the total length of the convergence line 20 along the line from the start point 23 of the line.

The same principle may be used to identify which of the convergence points 24 should be used for those forming shape points 19 which are located generally below the convergence line shown in Figure 8.7, or more accurately between the lower boundary line 27 of the endpoint 23 (having an angle denoted $\mathrm{ep}_{\mathrm{max}}$.) and the lower boundary line 26 of the start-point 22 (having an angle denoted $\mathrm{sp}_{\mathrm{min}}$). So for forming shape points located generally below the convergence line 20 shown in Figure 8.7, or more accurately between $\mathrm{sp}_{\mathrm{min}}$ and $\mathrm{ep}_{\mathrm{msx}}$, the position of their respective convergence points 24, as measured along the convergence line 20 from the start-point end 23, may be expressed as a percentage of the total length of the convergence line 20 by the expression:

$$100 \cdot \left(\left(\theta - \mathrm{ep}_{\mathrm{max}}\right) / \mathrm{A}_{\mathrm{IB}}\right) \tag{8.2}$$

Where AIB in this specific example = 32.5378 +360 - 158.2249 = 234.3129.

When the converging point 24 for each forming shape point 19 is known (as calculated above from the respective theta value of each point 19) then the initial array 18 of forming shape points 19 can be created around the convergence line 20. The forming shape points are positioned relative to their respective convergence points 24 in dependence on their polar coordinates (r, ϑ), where r is the same for each forming shape point 19.

As indicated above, an adaptive grid spacing (also known as adaptive mesh refinement) technique may optionally be used to ensure that the forming shape points 19 are approximately equi-spaced before and after each stage of convergence. An adaptive grid spacing algorithm may be used which works by identifying the distance between neighbouring forming shape points 19 and comparing the distance to a predetermined minimum spacing m. If the distance between a pair of neighbouring forming shape points 19 is less than the minimum spacing m, then one of the pair of points will be removed from the array 18. Conversely, if the distance between a pair of neighbouring forming shape points 19 is greater than the 2.1m (2.1m being used to avoid the unnecessary computation of adding point in only to take them out shortly after as the shape converges slightly and the points draw closer together) then an additional point will be added to the array 18, the added point having a theta value (ϑ) mid-way between the theta values of the two points, and a radius r which is an average of the radii of the two points.

Figure 6 illustrates schematically an optional primary convergence process which may be used to converge the forming shape points 19 generally towards their respective convergence points 24 on the convergence line 20 in a relatively swift manner to reduce the overall time required for convergence, and thus to avoid unnecessary use of processing power. The primary convergence process will be repeated for a number of iterations which, for example, may be between 5 and 10. The flow diagram of Figure 8.9 illustrates the stages involved in each iteration.

For each iteration of the primary convergence process, the convergence point 24 is found for each forming shape point ("FSP") 19, as denoted at 28 in Figure 8.9. It is then necessary to find the nearest final machined part point ("FMPP") 5 to each forming shape point 19, as denoted at 29 in Figure 8.9. This is done via an algorithm which identifies the final machined part point 5 which is closest in absolute distance to the particular (forming) shape point of interest ("POI"):

$$\text{AbsoluteDistance} = |\text{POI} - \text{FMPP}| \tag{8.3}$$

However, the nearest final machined part point 5 is only selected from those which lie within a 3 degree cone 30 of the notional line 31 between the forming shape point 19 of interest and its respective convergence point 24, as illustrated schematically in Figure 8.10, where the eligible final machine part points 5 are those falling within zone 32. This process simply involves an exhaustive search through all of the final machined part points 5, checking whether or not the dot product of the vector from the forming shape point 19 of interest to the convergence point 24 and the vector from the forming shape point 19 of interest to the final machined part point 5.

When the closest eligible final machined part point 5 each forming shape point 19 has been identified, the distance between the two is then calculated as indicated at 33 in the flow diagram of Figure 8.9. The distance which each forming shape point 19 should then be moved towards its convergence point 24, along its notional line 31 is then calculated, being ten units less than the linear distance between the forming shape point 19 and its nearest eligible final machined part point 5, as indicated at 34 in Figure 8.9. The forming shape points 19 are then moved accordingly, as indicated at 35 in Figure 8.9 to complete the iteration. It is to be noted in this regard that whilst the distance moved by each forming shape point 19 is calculated on the basis of its distance from its nearest eligible final machined part point 5, each forming shape point 19 is actually moved along its notional line 31. In the example used, the distance moved by each forming shape point 19 is set at ten units less than the distance between the respective forming shape point 19 and its nearest eligible final machined part point 5 so that even a very course mesh of the final machined part points 5 (for example less than 20 units) will not allow the forming shape point 19 to move past the meshed final part shape 1. It is to be appreciated that care should be exercised when defining the actual proportion of the distance between each forming shape part 19 and its nearest final machined part point 5 which is used for movement of the forming shape points 19, on the basis of the intended size of the final machined part. The example of ten units given is intended to be representative of a suitable degree of movement appropriate to estimate the precursor shape of a moderately sized gas turbine component such as the annular casing component illustrated, but a larger degree of movement may be appropriate for larger components.

Primary convergence is continued in the manner described above and illustrated in Figure 8.9 until the forming shape points 19 stop moving.

Following primary convergence of the forming shape points 19 as explained above for an appropriate number of iterations, the forming shape points 19 are then moved further via a secondary convergence process, as depicted in the flow diagram of Figure 8.12. The secondary convergence process is performed for a considerably larger number of iterations that the primary convergence process and typically for between 500 and 1000 iterations.

Conceptually, the secondary convergence process represents a balance between convergence forces and inverse squared repulsion from each forming shape point's nearest eligible final machined part point 5 (determined in the same manner as described above). Each point 19 is thus drawn along its respective notional lines 31, relative to its respective convergence point 24 according to:

- A force (F_1) pulling the point 19 towards its convergence point 24;
- A force (F_2) pulling the point 19 towards its nearest eligible final machined part point 5; and
- An inverse squared force (F_3) pushing it away from its nearest eligible final machined part point 5.

Having regard to Figure 8.11, it will thus be noted that this characteristic of the secondary convergence process serves to pull the forming shape points 19 which are relatively distant to their nearest eligible final machined part points 5 inwardly towards the convergence line 20, whilst pushing the forming shape points 19 which are relatively close to their nearest eligible final machined part points 5 outwardly and away from the convergence line 20, as indicated schematically by the arrows in Figure 8.11.

For each iteration of the secondary convergence process, the convergence point 24 is found for each forming shape point ("FSP") 19, as denoted at 36 in Figure 8.12. It is then again necessary to find the nearest eligible final machined part point ("FMPP") 5 to each forming shape point 19, as denoted at 37 in Figure 8.12, and this is achieved in the same manner as described above in connection with the primary convergence process. The movement of each forming shape point 19 will then be determined via a movement algorithm, as denoted at 38 in Figure 8.12, and the application of a saturation factor, as denoted at 39, as will be described in more detail below.

Where n denotes a unit vector from a forming shape point 19 towards its convergence point 24, L denotes the location of the forming shape point 19 in the Cartesian coordinate system 16, Mp denotes the position of the nearest eligible final shape point 5, and Cp denotes the location of the convergence point 24, then it follows that:

$$C_P = \begin{pmatrix} x_c \\ y_c \\ 0 \end{pmatrix} \tag{8.4}$$

Cp represents the co-ordinates of the convergence point 24

$$L = c_p + r \begin{pmatrix} \sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} \tag{8.5}$$

L represents the Cartesian location of the forming shape point 19

$$n = r \begin{pmatrix} \sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} \tag{8.6}$$

n represents the vector from the forming shape point 19

$$\hat{n} = \frac{n}{|n|} \tag{8.7}$$

The unit vector of n

$$B = L - M_p \tag{8.8}$$

B represents the vector from the final machined part point 5 to the forming shape point 19

$$\hat{B} = \frac{B}{|B|} \tag{8.9}$$

The unit vector of B.

The forces F1 , F2 , and F3 mentioned above may be expressed thus:

$$F_1 = k_1 \cdot \hat{n} \tag{8.10}$$

$$F_2 = k_2 \cdot \hat{B} \tag{8.11}$$

$$F_3 = k_3 \cdot \frac{\hat{B}}{|B|^2} \tag{8.12}$$

Where k_1, k_2, and k_3 is are empirically derived coefficients whose values will depend on which particular manufacturing process is to be used for production of the part, and thus for which manufacturing process the precursor shape is to be estimated. It will therefore be noted that the values for k_1, k_2, and k_3 will be different for a casting process than for a roll forging process for example.

So:

$$F = \sum_{n=1}^{3} F_n \tag{8.13}$$

The required distance m by which each forming shape point must be moved along its notional line 31 may thus be expressed as:

$$m = F \cdot \hat{n} \tag{8.14}$$

And so each forming shape point 19 is thus moved accordingly, as denoted at 40 in Figure 8.12. The secondary convergence process is repeated iteratively until the forming shape points 19 stop moving with each iteration.

Some exemplary values for the coefficients k_1, k_2, and k_3 for casting, die forging and roll forging are set out in the table below:

	Casting	Die Forging	Roll Forging
k_1	1	1	1
k_2	1	1	1
k_3	5	50	60

Table 8.1: Example values for the coefficients for FoSCo

Saturation limits can be placed to reduce movement where it exceeds a value x (either positive or negative). Where x is reduced linearly to zero at the end of secondary convergence. This is to inhibit the creation of numerical instability which is inherent in Euler integration analysis of second or higher order systems. Without placing saturation limits in this manner, the forming shape points 19 can develop a tendency to oscillate along their notional movement lines 31. If the movement vector is too large, it is capped at a value x which will depend on the progress through the convergence process. The cap may be either positive or negative depending on the direction or required movement of the forming shape point 19. The value x is reduced throughout the convergence. A large movement of the forming shape point 19 may be tolerated at the beginning of the process to allow rapid convergence, but may be reduced towards the end of the process in order to damp out unwanted oscillations of the forming shape point 19. The movement of the forming shape points 19 may also be restricted if any individual movement may shift a forming shape point 19 past the final machined part points (i.e. effectively into the finished part geometry), which may arise if the mesh of the final machined part points 5 is too coarse and/or the value of the coefficient k3 is too low. If the computed movement of the forming shape point 19 is greater than the distance d, then the movement will be set to d/10.

Following secondary convergence of the forming shape points 19 as explained above for an appropriate number of iterations, the forming shape points 19 are then moved further via a tertiary convergence process, as depicted in the flow diagram of Figure 8.14, which may be considered to be a smoothing process.

The tertiary convergence process is again iterative and is performed until the forming shape points 19 stop moving with each iteration or until a preset counter is reached (for example 1000 since some forming shape points 19 may have a movement vector perpendicular to the required movement direction). The tertiary convergence process is effective to increase the radius r (along the notional line 31 from the forming shape point to its convergence point) of any forming shape point 19 which is found to lie too far inside a notional line interconnecting the neighbouring forming shape points 19 on either side, thereby moving the point 19 outwardly relative to its convergence point 24.

As will be noted from Figure 8.13, the tertiary convergence process again involves finding the convergence point 24 for each forming shape point ("FSP") 19, as denoted at 41 in Figure 8.13. It is then necessary to find the locations of the neighbouring forming shape points 19 to each side of the forming shape point 19 of interest, as denoted at 42 in Figure 8.13, after which a smoothing algorithm is solved for the point, as denoted at 43, in order to determine whether or not the point should be moved outwardly from its convergence point 24.

The manner in which this is achieved may be understood having regard to Figure 8.14, in which the forming shape point 19 of interest is denoted as P, and the two neighbouring forming shape points 19 to each side of the point of interest are denoted A and B respectively. The vector between the two neighbouring points A and B is found and a line AB is drawn between the two neighbouring points. The shortest vector, denoted by PI in Figure 8.14, between the forming shape point of interest P and the line AB is then calculated. The allowable magnitude (M) of vector PI is defined by:

$$M = k_4 \cdot |\text{AB}| \tag{8.15}$$

Where k_4 is an empirically derived coefficient whose value will depend on which particular manufacturing process is to be used for production of the part, and thus for which manufacturing process the precursor shape is to be estimated. By way of example k_4 equals,

- 1 for a casting process;
- 0.005 for a die forging process;
- 0.005 for a roll forging process.

Vector CP shown in Figure 8.14 denotes a vector from the forming shape point's convergence point 24 to the forming shape point itself. If the dot product of vector PI and vector CP is negative, then the forming shape point of interest P will lie within the line AB relative to its convergence point 24. If this is the case, and if the magnitude of vector PI is greater than the allowable amount M, as calculated above, then the forming shape point P will be moved outwardly, further away from its convergence point 24, along its position vector until PI no longer exceeds M. In this regard, it is proposed to move the forming shape point outwardly by a distance equal to 0.999M - $\left|\text{PI}\right|$. This expression ensures that, when used iteratively the point P will lie just inside the acceptable region, provided that the movement vector is not perpendicular to the required direction of travel.

Tertiary convergence (smoothing) continues as explained above, in an iterative manner, until there is no longer any movement of the forming shape points 19. At this point, the forming shape points 19 will collectively define the outline of an approximately optimised precursor shape which can be produced by the chosen manufacturing process, and which will require minimum subsequent machining to achieve the final part shape 1. Figure 8.15 illustrates exemplary precursor shapes achieved via the method explained above for casting and for roll-forging, relative to the final part shape 1 and the convergence line 20. Outline shape 44 represents an exemplary precursor shape for casting, and outline shape 45 represents an exemplary precursor shape for roll-forging. As will be noted, the precursor shape 44 for casting is considerably closer to the final part shape 1 than is the precursor shape 45 for rollforging.

Following estimation of the precursor shapes 44, 45 via the method described above, the precursor shapes will then be imported back into the CAD package which was originally used to create the final part shape 1, as illustrated in Figure 8.16. The CAD package can then be used to calculate the dimensions of the precursor shapes 44, 45 automatically, so that the dimensions can then be used in the chosen production process to achieve the precursor shape.

Figure 7 illustrates an exemplary part 2, whose 2-dimensional precursor shape 46 may be estimated via the method described above. It is to be appreciated that whilst the method has been described above with specific reference to the estimation of a 2-dimensional precursor shape, the method can also be used to estimate a 3-dimensional precursor shape.

As will be appreciated, the above-described method is suitable for implementation on a computer.

Figure 18 illustrates an exemplary computer apparatus 47 which may be configured to implement and run the above-described method. The computer apparatus illustrated is shown in the form of a conventional desktop computer, and thus comprises a system unit 48 to which are connected a display unit 49, and input devices in the form of a keyboard 50 and a mouse 51. Having regard to Figure 8.19, the computer apparatus 47 may comprise one or more processors 52 configured to perform the above-described method. The computer apparatus 47 may also comprise, or be configured to receive, a storage medium 53. The storage medium is a non-transistory computer readable storage medium which can take any convenient form, but which excludes merely transistory or propagating signals. The storage medium 53 contains computer readable instructions in the form of a computer program 54 which, when read by the computer 47 via its processor 52, cause performance of the above described method.

When used in this specification and claims, the terms “comprises” and “comprising” and variations thereof mean that the specified features, steps or integers are included. The terms are not to be interpreted to exclude the presence of other features, steps or integers.

The features disclosed in the foregoing description, or in the following claims, or in the accompanying drawings, expressed in their specific forms or in terms of a means for performing the disclosed function, or a method or process for obtaining the disclosed results, as appropriate, may, separately, or in any combination of such features, be utilised for realising the invention in diverse forms thereof.

While the invention has been described in conjunction with the exemplary embodiments described above, many equivalent modifications and variations will be apparent to those skilled in the art when given this disclosure. Accordingly, the exemplary embodiments of the invention set forth above are considered to be illustrative and not limiting. Various changes to the described embodiments may be made without departing from the spirit and scope of the invention.