A virus-evolutionary, multi-objective intelligent tool path optimisation methodology for sculptured surface CNC machining

by

Nikolaos A. Fountas

A thesis

submitted to Kingston University (Faculty of Science, Engineering and

Computing - School of Engineering and the Environment, Department of

Mechanical Engineering)

in partial fulfillment of the requirements for the Degree of

Doctor of Philosophy

in

Mechanical Engineering

Major Subject: Intelligent Manufacturing

Kingston upon Thames, London, UK, 2019

This is a true copy of the thesis, including any required final revisions, as accepted by my supervisors.

Abstract

Today's production environment faces multiple challenges involving fast adaptation to modern technologies, flexibility in accommodating them to current industrial practices and cost reduction through automating repetitive tasks. At the same time the requirements for manufacturing functional, aesthetic and versatile products, turn these challenges to clear and present industrial problems that need to be solved by delivering at least semi-optimal results. Even though sculptured surfaces can meet such requirements when it comes to product design, a critical problem exists in terms of their machining operations owing to their arbitrary nature and complex geometrical features as opposed to prismatic surfaces. Current approaches for generating tool paths in computer-aided manufacturing (CAM) systems are still based on human intervention as well as trial-and-error experiments. These approaches neither can provide optimal tool paths nor can they establish a generic approach for an advantageous and profitable sculptured surface machining (SSM).

Major goal of this PhD thesis is the development of an intelligent, automated and generic methodology for generating optimal 5-axis CNC tool paths to machine complex sculptured surfaces. The methodology considers the tool path parameters "cutting tool", "stepover", "lead angle", "tilt angle" and "maximum discretisation step" as the independent variables for optimisation whilst the mean machining error, its mean distribution on the sculptured surface and the minimum number of tool positions are the crucial optimisation criteria formulating the generalized multi-objective sculptured surface CNC machining optimisation problem.

The methodology is a two-fold programming framework comprising a virus-evolutionary genetic algorithm as the methodology's intelligent part for performing the multi-objective optimisation and an automation function for driving the algorithm through its argument-passing elements directly related to CAM software, i.e., tool path computation utilities, objects for programmatically retrieving tool path parameters' inputs, etc. These two modules (the intelligent algorithm and the automation function) interact and exchange information as needed towards the achievement of creating globally optimal tool paths for any sculptured surface.

The methodology has been validated through simulation experiments and actual machining operations conducted to benchmark sculptured surfaces and corresponding results have been compared to those available from already existing tool path generation/optimisation approaches in the literature. The results have proven the methodology's practical merits as well as its effectiveness for maintaining quality and productivity in sculptured surface 5-axis CNC machining.

Acknowledgements

First and foremost, I want to wholeheartedly express my sincere gratitude to my supervisors, Prof. Dr. Redha Benhadj-Djilali, Prof. Dr.-Ing. Constantinos I. Stergiou and Prof. Dr.-Ing. Nikolaos M. Vaxevanidis for all their support and guidance during my studies through all these years. I am also grateful to Dr.-Ing. Agathocles A. Krimpenis for his scientific interest, his technical recommendations and fruitful discussions on intelligent algorithms and their engineering applications. I am deeply indebted to Mr. Panagiotis Mermelas, senior CNC programmer of Hellenic Aerospace Industry (H.A.I.) for agreeing to provide the necessary equipment so as to validate my research results on sculptured surface CNC machining operations.

Declaration of originality

This is to certify that the content of this thesis is my own work. This thesis has not been submitted for any degree or other purposes. I certify that the intellectual content of this thesis is the product of my own work and that all the references used in preparing this thesis as well as other sources have been acknowledged.

Table of Contents

List of Figures

List of Tables

Abbreviations and Symbols

AIM:	Arc-Intersect Method
API:	Application Programming Interface
APT:	Auto Programmed Tool
CAD:	Computer-Aided Design
CAM:	Computer-Aided Manufacturing
CCD:	Central Composite Design
CL:	Cutter Location
CNC:	Computer Numerical Control
C_{pop}:	Population of Candidate Solutions in a Genetic Algorithm
CS_{MS}:	Candidate solution for the machining strategy (Tool path Chromosome)
DOE:	Design of Experiments
EAs:	Evolutionary Algorithms
FCO:	Efficient Convergent Optimisation
G:	Number of Generations to be evolved in a Genetic Algorithm
GAs:	Genetic Algorithms
ITM:	Inclined tool Method
MEM:	Mechanical equilibrium Method
MPM:	Multi-Point Machining
MS:	Machining Strategy
MSW:	Machining Strip Width
NoCLs:	Number of cutter location points
OAs:	Orthogonal Arrays
OF_{PO}:	Multi-objective Pareto optimisation function
OF_{WS}:	Single-objective (weighted) optimisation function
PAM:	Principal Axis Method
Prm:	Tool path Parameter
RBM:	Rolling Ball Method
RCM:	Rotary Contact Method
RSM:	Response Surface Methodology
$rate \downarrow V_{life}$:	Virus life reduction rate (algorithm-specific parameter)
SS:	Sculptured Surface
SSM:	Sculptured Surface Machining
a_e (%Ø):	Stepover (tool path interval or side step)
a_L (∘):	Lead angle
a_T (∘):	Tilt angle
c_1, c_2:	Principal curvatures
C_1, C_2, C_3:	Criteria for multi objective optimisation
h:	Scallop height
$\bar{h}$:	Mean scallop height
$\bar{\delta}$:	Mean chord error (or chordal deviation)
$\overline{stdevh}$:	Mean standard deviation for scallop height
$\overline{stdev\delta}$:	Mean standard deviation for chord error
i, j, k, :	Tool axis vector components for x,y,z tool path Cartesian coordinates
MaxDstep:	Maximum discretization step

ME $\overline{\delta} + \overline{h}$ Mean machining error

$V_{\text{inf Rate}} \max :$ Virus infection rate (algorithm-specific parameter)

$V_{pop} :$ Population of Virus Individuals in a Virus-Evolutionary Genetic Algorithm

$V_{strlength} \max :$ Maximum allowable length of a Viral chromosome (algorithm-specific parameter)

x, y, z,:   Tool path Cartesian coordinates

Chapter 1

Introduction

The usage of sculptured surfaces in product development has increased dramatically in the past few years owing to their property of combining high aesthetics and long-term functionality to a wide range of products found in aerospace, automobile, mold/die, electronics and bioengineering. Consequently, there is an ever-increasing requirement to manufacture parts comprising sculptured surfaces.

Such surfaces are mainly produced by means of material removal (cutting) operations with emphasis to 5-axis machining. Since such a machining process can only be implemented using computer numerical control (CNC), part programs in the form of ISO codes need to be prepared in dedicated manufacturing software (CAM systems) as direct results of tool path planning.

Tool path planning is a critical task that is usually conducted by experienced machinists and programmers with the use of a typical and commercially available CAM system. Considering that the main stages of sculptured surfaces production are roughing, semi-finish, finishing and polishing (benchwork), tool path planning is required for roughing and finishing. The purpose of roughing stage is to remove the unnecessary material volume from the raw stock at high production rates and to approach a semi-final part geometry close to the final one. The purpose of finishing is the removal of roughed part's remaining volume achieving thus, the final requirements in terms of quality, dimensional accuracy and tolerance. Since it is physically impossible to reach the ideally designed surface of a product, time-consuming benchwork is needed for polishing sculptured surfaces. The labor needed mainly depends on the varying complexity and the material of a sculptured surface product. It is estimated that over 78% of the overall production time is devoted to finishing, grinding and polishing of such products (Warkentin et al. 1997). Therefore it is obvious that more advantageous (if not optimal) finishing tool paths are needed to achieve better surface consistency so that benchwork is reduced or ultimately avoided.

A common technique to plan tool paths for the finishing stage is to select one of the standard currently available pattern distribution styles and apply it to the designed stock with reference to the target model and the upper and lower part levels. The tool path may be applied either as an entire cutting pattern to provide complete surface coverage or to limited contours constituting the surface

regarding its complexity and special geometrical features. In either case the tool path planning operation must ensure that the finished surface will come to meet the morphological and quality demands in order to be considered in agreement with technical specifications. Towards this goal, a number of parameters crucial to tool path planning task should be properly set.

The most important obstacles for achieving the above goal are three. The first obstacle suggests that no unique combination among tool path parameters and their corresponding result exists. The second obstacle deals with the particular relations between varying cutting tool geometries and surface configurations (i.e. curvature) which is impossible to be examined on-the-job. The third obstacle is the fact that magnitudes of crucial performance objectives representing part quality and process productivity may strongly fluctuate towards the tool path trajectory owing to the complexity of a sculptured surface, i.e., convex, concave and saddle surface regions. This obstacle is even more profound in the case of 5-axis surface machining where two additional axes need to be determined to effectively adapt to part's changing curvature and perform smooth tool positioning variation as well.

5-axis sculptured surface machining is a well-established field for which numerous tool path strategies have been already proposed and seen service in the market's leading CAM systems, yet, they all share several shortcomings such as lack of essential optimisation functions, insufficient automation level and significant end-user intervention. At a practical level, approaches based on expertise, trial-and-error experiments and empirical adaptations of known feasible solutions to new, similar problems are generally followed. Undoubtedly, such approaches impose general assumptions/simplifications of the problem at hand, whilst the majority of cases end up with conservative solutions without general application.

Even though significant contributions have been made to all research branches of sculptured surface machining, there is still a need to come up with a comprehensive, generic, intelligent and practically viable methodology for generating optimal tool paths for the 5-axis machining of sculptured surfaces. By recognizing aspects involving the need to develop such a methodology, the need for higher intelligent and automation environment in modern production and the problems/obstacles of traditional 5-axis tool path planning approaches, this Ph.D. thesis establishes important research questions related to the problem as follows:

- How the sculptured surface CNC machining problem should be expressed to acquire a generic representation?

- How the sculptured surface CNC machining problem should be solved to have practical rationale and deliver applicable results to industry?
- How to reduce labour cost and human intervention when it comes to tool path planning for sculptured surface CNC machining?
- What kind of environment facilitates the generic solution of the sculptured surface CNC machining problem and how its outputs can be directly transferred to industry?
- What kind of module should be selected to provide "intelligent" optimisation capabilities to problem solving?

The research questions stated above are answered through the major goal and the objectives that this Ph.D. thesis establishes.

1.1   Research aim and objectives

The major aim of this PhD thesis is to develop a generic methodology for the intelligent optimisation of 5-axis sculptured surface CNC machining (end-milling) tool paths. To fulfill this aim the methodology takes under full consideration the above problems as well as the literature related to the sculptured surface CNC machining problem and establishes criteria with global optimisation perspectives.

The methodology represents the environment where the sculptured surface CNC machining problem is turned to a machining modeling task to be optimised for producing tool paths with superior performance. The sculptured surface CNC machining problem is treated in the Ph.D. thesis as a problem of multi-objective nature, being influenced by the settings corresponded to key tool path parameters.

The criteria of this thesis aim to represent the simulated part quality and the productivity of the process under which quality should be achieved. The machining error (as a combination of chordal deviation and scallop height) has been selected to formulate the first criterion that has to be minimized. The second criterion deals with the local variations of machining error as the cutting tool moves from point to point throughout the tool path and should be minimized. The third criterion corresponds to the total number of cutting points constituting a tool path under the assumption that such magnitude directly affects the time needed for processing and cycle time itself. This objective should also be minimised. The rationale behind the inclusion of the aforementioned optimisation

criteria to the optimisation problem stems from the need to simultaneously satisfy the two essential aspects of quality and productivity. Machining error as a combination of the cutting tool interpolation error and scallop height, characterises surface finish and thus, quality. Machining error distribution constitutes an indication to quantify local tool axis variations responsible for producing the machining error throughout the entire tool path. Typical and commercially available CAM systems estimate local machining errors and incorporate additional cuts, if applicable, into a single block. However, such a utility would inherently lead to a substantial increase of cutting points comprising a tool path and consequently an increase of the machining time. Therefore the total number of cutting points for tool paths is considered an important optimisation criterion to reflect machining time and thus, productivity. Note that an alternative objective for representing productivity may be tool path length.

Tool path parameters that need to be set in any given 5-axis finish-machining strategy have been considered as the independent variables for which not only feasible but beneficial values should be determined so as to come up with an advantageous tool path. These parameters are the cutting tool type (referring to flat end-mills and filleted end-mills), the stepover (step for determining the distance among adjacent cuts to machine a surface), lead angle (the rotation of the cutting tool in feed direction), tilt angle (the rotation of cutting tool towards a vector perpendicular to feed direction) and maximum discretisation step (step for determining the maximum distance among tool path points towards feed direction).

Obviously, the different settings for any of the aforementioned tool path parameters inevitably affect the resulting tool path and as a consequence the machining operation itself. Each of these parameters plays an essential role and contributes just as significantly when it comes to both productivity and precision machining.

The methodology that this research has been established focuses on the need to simultaneously examine the tool path parameters so as to deliver a set of values for their settings and satisfy the three optimisation criteria stated above. To manage the establishment of such an environment, three discrete objectives were defined as follows:

1. Experimental exploration of crucial relations to be selected for their capability to represent 5-axis sculptured surface CNC machining as a generic problem and extract globally optimal solutions when these relations are to be handled by an intelligent module.
2. Development of a fully automated interface to handle CAM software properties as needed, to automate repetitive and time-consuming tasks and to interact with an intelligent module for proper feedback and control.

3. To develop, to test and finally to deploy an artificial intelligent algorithm to undertake the optimisation process.

To meet the requirements of these objectives, development tasks were planned for each one as follows:

For the first objective's requirements the development tasks are:

- Design of standard benchmark sculptured surfaces based on the literature for machining simulation experiments.
- Design of experiments and statistical evaluation-comparison between analytical results and experimental measurements taken from machining simulation outputs.

For the second objective's requirements the development tasks are:

- The development of programming functions for automating CAM environment's repetitive activities namely: (1) project tree scanning, (2) tool path strategy retrieval, (3) automatic assignment of values for tool path parameters, (4) automatic tool path computation, (5) automatic cutter location data (CL data) file creation and (6) CL data evaluation with respect to the theoretical sculptured surface (target CAD model).
- The development of routines either to import or to extract important data according the methodology's processing phase, argument-passing capabilities among functions, etc.

For the third objective's requirements, the development tasks are:

- The development of data structures to represent tool path chromosomes through an encoding scheme to facilitate the methodology's computations.
- The formulation of the objective function and its representation as a "Pareto" triple-bounded criterion.
- The development of conventional genetic operators for building a genetic algorithm compatible to CAM environment's open application programming interface (API).
- The development of additional non-conventional intelligent operators to improve the functionality of the genetic algorithm.

Having investigated the proposed methodology's strengths and weaknesses, it was implemented to benchmark case studies with experimental results available from the literature for fair and rigorous comparisons. Validation of the proposed methodology has been conducted by performing both algorithmic and process-related experiments. Process-related experiments refer to actual machining operations performed to represent the positive impact, the gain and the merits of the methodology's implementation.

Based on the results obtained by implementing the methodology developed as well as their comparison to those available in the literature for the same problem, same impact cases and – wherever it was feasible – same resources, its contributions have been recognised as follows:

1. The methodology constitutes a practically viable tool to optimise complex sculptured surface tool paths by using standard and known resources to practitioners.
2. The methodology pushes further the envelope of profitability and efficiency of intelligent manufacturing by supporting automation and optimisation.
3. The methodology handles simultaneously a large number of parameters and achieves optimisation under a global sense.
4. The methodology shares and develops new ideas for the next generation's manufacturing software development, dealing with artificial intelligence and its effective implementation.

1.2 Thesis outline

The detailed description of the proposed methodology for optimizing 5-axis tool paths for sculptured surface CNC machining is given in seven chapters. The first chapter (Chapter 1) motivates and introduces the important problem of 5-axis sculptured surface CNC machining. It also mentions the major goal and the objectives of the work as a consequence of important research questions as well as the development tasks in order to reach the final status.

Chapter 2 presents the fundamentals of tool path generation for 5-axis sculptured surface CNC machining and gives a critical review of the most noticeable and latest research contributions in the broader scientific field of sculptured surface tool path generation. Emphasis is given to the philosophy underpinning the different approaches to solve the problem and their key attributes they implement. The review discusses also the types of independent process parameters as well as the criteria that other research works have already presented as promising aspects for solving the problem. The chapter ends with the knowledge gap and the conclusions-shortcomings of the existing literature.

Chapter 3 deals with the problem formulation, the definition of criteria for global optimisation and their experimental validation prior to their integration with the proposed methodology's environment. Main effects and interactions among critical tool path parameters are examined to be taken into account in the methodology's development.

Chapter 4 presents the steps and the overall workflow of the methodology developed to optimize 5-axis sculptured surface CNC machining tool paths. The multi-objective virus-evolutionary genetic algorithm (MOVEGA) and the interactive function comprising the methodology are presented in detail as the intelligent module and the automation module respectively.

Chapter 5 presents the methodology's implementation on a benchmark sculptured part under the scope of investigating the effects of algorithm-specific parameters on results dealing with solution quality and convergence speed. Investigation of algorithm-specific parameters is accomplished by examining the coverage of non-dominated solutions from several Pareto fronts according to experimental runs as well as by assessing the generations where final points are observed in convergence diagrams. Through this parametric study, advantageous settings for algorithm-specific parameters are determined to improve further the overall performance of the methodology to solve the sculptured surface CNC machining problem.

Chapter 6 presents the results from the implementation of the proposed optimisation methodology to several impact cases using benchmark sculptured surfaces. Moreover, results for validating the proposed methodology are also reported and compared to those already published by other researchers for the same problem and same surfaces. These results have been obtained by performing actual 5-axis machining experiments using optimized tool paths. In addition, results from comparisons among the algorithm developed and others found in the literature are also given in the chapter.

Chapter 7 presents the overall conclusions are drawn and achievements accompanied with the pros and cons of the methodology are discussed. The chapter is concluded by proposing new directions for further research work and future perspectives.

1.3 Publications

- Fountas N.A., Vaxevanidis N.M., Stergiou C.I., Benhadj-Djilali R. (2018). Globally optimal tool paths for sculptured surfaces with emphasis to machining error and cutting posture smoothness. International Journal of Production Research, DOI: 10.1080/00207543.2018.1530468.

- Fountas N. A., Benhadj-Djilali R., Stergiou C. I., Vaxevanidis N. M. (2017a) An integrated framework for optimizing sculptured surface CNC tool paths based on direct software object evaluation and viral intelligence. Journal of Intelligent Manufacturing, http://dx.doi.org/10.1007/s10845-017-1338-y

Chapter 2

Literature Review

Among the problems that need to be addressed in metal cutting processes, two are of particular importance. The first deals with the determination of those values of process parameters that will maintain high product quality meeting thus the general technical requirements. The second refers to the simultaneous maximization of profit and process performance. Owing to the complexity characterizing machining processes, the noise factors and the interactions among several operational parameters, delivering an optimal solution sounds difficult if not impossible. Furthermore stat-of-the-art technology and ever-increasing developments in computerized production systems impose new research directions involving intelligence, automation and flexible decision-making.

At a practical level, practices based on experience are still preferred. Experience-based practices may involve either the application of previous successfully implemented technical approaches to solve a new problem or the application of empirical relations. Such options are based on assumptions/simplifications and, to a large extent, they can only lead to conservative solutions without generic characteristics. At a scientific level, most of the research directions where contributors have shown interest are the correlation among influential parameters of a process to its crucial quality criteria, the development of algorithms based on local geometrical data for computerized control and the application of artificial intelligent techniques for the heuristic search of optimal results.

The aforementioned problems become even more tedious in terms of their solution in the case of tool path planning to machine parts comprising sculptured surfaces. Their complexity implies a number of points to concern, i.e., each product is "unique" hence, restricting the adaptation of a previously successful technique to a new part, machining time increases owing to the large number of cutting tool positions a tool path generates in order to stay under tolerance and the fact that such products are deemed of high precision despite their free-form geometry.

This chapter attempts to provide a solid background concerning the essentials of tool path planning for the machining of sculptured surfaces using 5-axis CNC technology. A detailed literature review is also presented with emphasis to the most noticeable contributions. The methodologies are given

through a number of categories based on their similarities whilst their pros and cons are critically discussed. The chapter ends by mentioning the conclusions derived from the literature review as crucial attributes to reveal their contribution range they have achieved so far as well as the remaining knowledge gap.

2.1 Tool path planning for 5-axis sculptured surface machining

Tool path planning is an important activity of machining modeling process. During this process the 3D model representation of the part is imported to a CAM system. With reference to the 3D model a number of manufacturing attributes are determined. Given the part's geometry, the stock is decided to be either an offset form of the 3D model or a standard prismatic solid (i.e. rectangular, orthogonal, plate, etc). According to the machining setup and fixture, the machining reference axis system (G54) is determined. A machining operation along with its corresponding tool path is then applied to the model. The top, bottom and safety planes are determined next. Several tool paths are available in CAM systems such as multi-axis sweeping, concentric, spiral, Z-level and iso-parametric to name a few. Although these tool paths differ significantly in terms of their cutting style, they all need to be planned by selecting the cutting tool type, the distance between adjacent passes (known as stepover), the two inclination angles (lead and tilt) for varying the tool axis towards cutting direction and the forward (or discretisation) step for determining the interpolation error with reference to the theoretical surface (Turnier and Duc 2005, Lavernhe et al. 2007).

The tool path is represented as a set of cutting points from which the tool will pass on its way to machine the surface towards feed direction. The cutting tool interpolates subsequently these points whilst it performs several adjacent passes across the entire surface. The number of adjacent passes influences directly the height of the scallop which is the uncut material remained among tool passes. The interpolation error among individual tool positions with regard to subsequent cutting points may be large enough to cause the tool to mismatch the surface. This error can be reduced by properly defining the step the cutting tool takes to move forward to the next cutting point. Both the number of adjacent passes and the step defining interpolation error affect the magnitude of cutting points or, equivalently, the number of tool positions. Thus, an advantageous tool path should simultaneously maintain low scallop height, low interpolation error and reduced number of cutting points/tool positions.

Tool path points are converted to Cartesian coordinates and tool axis vectors with the usage of a post-processor engine embedded to CAM software. The selection of post-processor depends on the type and functions of the CNC unit integrating the 5-axis machining center to be employed for machining the sculptured surface. The particularities of such equipment deal with kinematical properties as well as lead and tilt angle configurations (Warkentin et al., 2001) therefore the post-processor to be selected for generating the ISO code from cutting data should simulate and translate exactly the same functions.

2.1.1 Cutting tool geometry

5-axis CNC technology provides many beneficial utilities for sculptured surface machining. One of those is the ability to select from a variety of cutting tool geometries, as opposed to 3-axis machining where only ball end-mills can be used for finish-machining sculptured surfaces. In order to ensure surface quality in terms of low scallop height and interpolation error, many closely spaced adjacent passes and forward steps need to be determined in 3-axis machining. In each tool pass a hemi-spherical posture is left on the surface as an impression of the removed volume from the work piece. In addition, much of the ball end-mill's machining is conducted near the bottom end of its center where tangential speed is the lowest, hence, deteriorating surface quality. In this case much time should be spent on benchwork to finish the part.

On the contrary in 5-axis machining flat end as well as filleted end-mills can be selected for machining sculptured surfaces (Figure 2.1). 5-axis machining technology allows the cutting tool to be inclined about surface curvature avoiding this way machining with bottom end where cutting speed is theoretically zero and favoring machining at cutting tool's edge where speed reaches its highest level. Inclined cutting leaves more advantageous material removal postures which have elliptical shapes. By changing inclination angles the dimensions of these elliptical shapes may be altered to better approximate the surface curvature, lead to smaller scallops, avoid gouging and allowing for less adjacent tool passes to machine the surface. Consequently, fewer cutting points are required compared to traditional 3-axis surface machining. Numerical and experimental results have been provided by Vickers and Quan (1989) as well as by Bedi et al., (1997) to show the beneficial nature of flat end-mills and filleted end-mills against ball end-mills in 5-axis surface machining.

Flat head   Bull head   Ball head
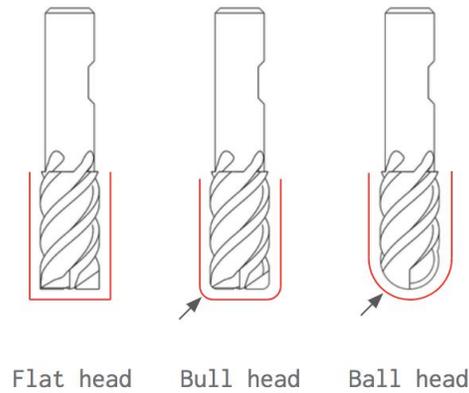
Figure 2.1: Standard cutting tool geometries for sculptured surface CNC machining: flat-end, filleted-end and ball-end mills.

2.1.2   Stepover

Stepover (or tool pass interval) parameter is responsible for determining the cutting tool's transversal step among adjacent tool passes. Its value can be directly determined using either distance units, i.e. mm or decimals of an inch, or it can be expressed as a percentage of the cutting tool's nominal diameter. It is also possible to be determined as the overlap distance among tool passes or via the number of total paths with regard to the part's nominal length (Figure 2.2). In the case of 3-axis machining it can also be determined by the required scallop height. Stepover parameter along with the cutting tool type determines the magnitude of scallop height. Stepover alone influences the overall tool path length and therefore machining time. Large stepover values would result to less cutting passes and machining time but larger scallop heights at the same time.
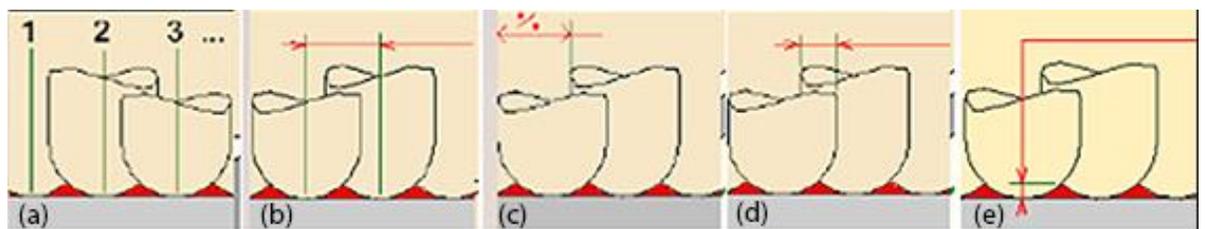
Figure 2.2: Standard functions for stepover parameter adjustment: (a) number of paths, (b) distance, (c) distance as a percentage of tool diameter, (d) overlap, (e) scallop height (*Dassault Systèmes* CATIA V5 R18).

2.1.3   Inclination angles

Lead and Tilt angles determine the cutting tool inclination regarding the machining surface. The former angle is the angle between the surface normal and the new tool orientation in the direction of the machining path tangent, whereas the cutting tool's inclination with reference to the surface normal from this position corresponds to tilt angle (Figure 2.3).
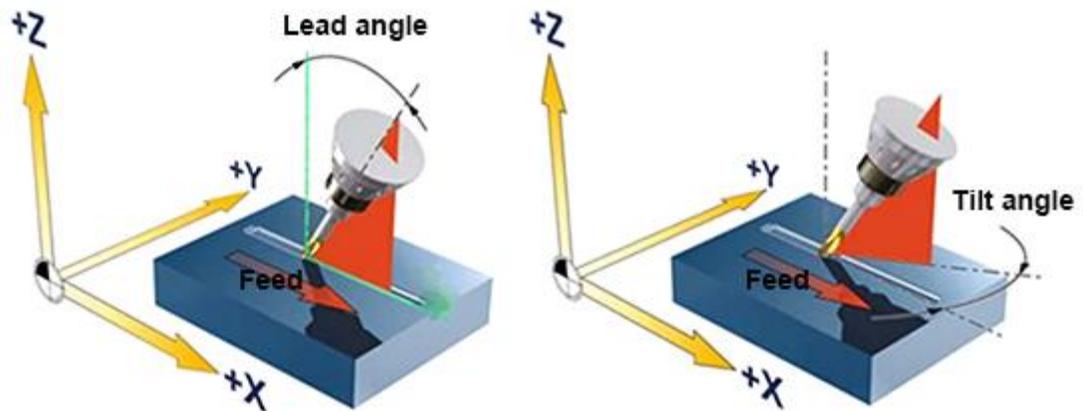
Figure 2.3: Tool inclination angles (lead and tilt) in 5-axis sculptured surface CNC machining (Siemens© AG SINUMERIK, Manual, 5-axis machining, 2009**)**.

In 5-axis sculptured surface machining it is evident that different cutting tool orientations in terms of lead and tilt angles determine different effective cutting shapes which in turn influence scallop height, number of adjacent passes and machining strip width. Machining strip should be as wide as possible to yield a high material removal rate and simultaneously allow for reducing tool path intervals (step-over passes) as well as scallops towards feed direction. In order to maintain wider machining strips, lead and tilt angles ought to be as low as possible. On the other hand, low inclination angles might yield gouges between the cutting tool and the machining surface.

2.1.4   Maximum discretisation step

Maximum discretisation step (Figure 2.4) allows the determination of the largest spacing between subsequent cutting points/tool positions along a cutting tool pass in feed-forward direction. Therefore, it refers to the determination of the maximum allowable value for feed-forward distances along a tool pass according to a preset tolerance. Tool positions along a tool pass should be closely spaced to avoid significant deviations from the theoretical surface as the CNC unit conducts

interpolation. On the other hand, the overall number of cutting points should be minimised to facilitate the functions of the CNC controller. Such a problem is difficult to be solved in the case of 5-axis surface machining owing to kinematics and the requirement to remain within a preset tolerance.

Traditional tool path planning would suggest first to ensure machining accuracy by applying low values for maximum discretisation step parameter rather than prioritizing the throughput of the CNC unit. Another important issue that would tempt a process planner to select low discretisation step for generating a sculptured surface tool path is gouging avoidance. Maximum discretisation step parameter should be taken under careful consideration because the smallest change in its corresponding value could result to different topological properties of cutting points, denser point spacing, larger number of tool orientations and as a consequence larger cutting tool joint trajectory. Therefore, there is a need to adjust maximum discretisation step parameter such that machining accuracy is maintained, yet, without too closely spaced cutting tool orientations. An illustration of the effect of maximum discretisation step parameter on surface quality is shown in Figure 2.5.
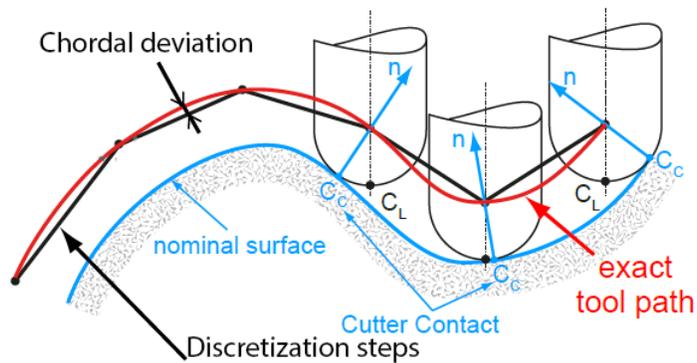
Figure 2.4: Discretisation step parameter for sculptured surface CNC machining tool paths (Beudaert et al. 2014).
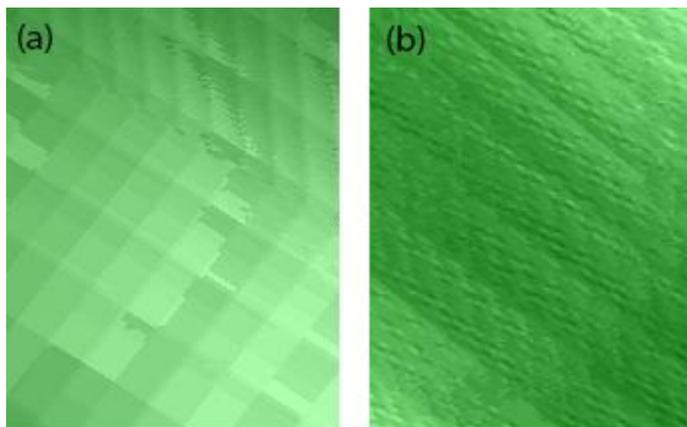
Figure 2.5: Effect of maximum discretisation step parameter on surface quality: (a) large discretisation, (b) small discretisation (*Dassault Systèmes* CATIA V5 R18).

2.2   Geometrical indicators for evaluating machining accuracy

It is obvious that the traditional decision-making followed for generating tool paths to machine complex sculptured surfaces may significantly affect final results referring to machining accuracy and productivity. Under the premise that CAM software and other related systems for virtually planning manufacturing processes (Lopez de Lacalle et al., 2005, Altintas et al., 2014,) constitute reliable, trustworthy and time-saving environments, much of the research works concerning tool path planning and optimisation, have been focused on the identification of crucial performance metrics for assessing tool paths, towards their ultimate goal of developing, deploying and testing their approaches.

Approaches aiming towards beneficial tool paths, i.e. tool path planning strategies, tool positioning strategies, intelligent systems, etc., have inevitably evaluated their contribution using performance metrics (criteria or objectives) that can be handled by computational algorithms and not process-related indicators such as tool wear, surface roughness, system stability, etc., which can only be assessed by conducting actual manufacturing operations. Nevertheless, the practices these approaches suggest can deliver promising outcomes when accompanied to reliable decision-making referring to the determination of process parameters (i.e. feed rate, cutting speed, depth of cut) that affect physical objectives such as those reported above.

The most important performance metrics known also as "criteria" or "objectives" to evaluate the techniques available to the existing literature so far are scallop height, chordal deviation, machining error and machining strip width. Based on the evidence concerning the influence of the aforementioned tool path planning parameters on such criteria, it has to be noted that the selection of a single criterion to assess resulting tool paths not only implies the existence of another but also the introduced trade-off.

2.2.1   Scallop Height

The material left uncut among consecutive tool passes in the transverse direction is known as scallop (Lin and Koren, 1996), whereas its maximum limit on the height is known as scallop height. In 3-axis machining scallop volume inherits the negative ball-end shape of the tool. Therefore it is relatively simple to predict or control scallop height given the stepover distance and the diameter of a ball end-mill. Functions for computing scallop height with reference to the diameter of the ball end-mill and

the stepover distance have already been established by Feng and Huiwen (2002) and Chen et al., (2005). Figure 2.6 illustrates the scallop height formation in the simple case of 3-axis machining.
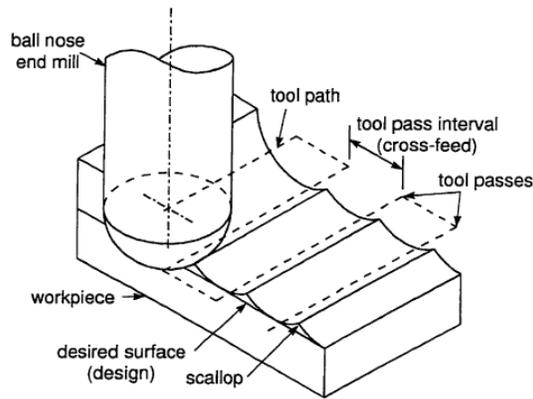
Figure 2.6: Scallop height in 3-axis CNC machining.

In the case of 5-axis surface machining scallop height is affected by cutting tool geometry, stepover distance and cutting tool inclination angles. Cutting tool geometry alters its swept posture while travelling towards feed direction to remove the material from the work piece, owing to lead and tilt angles. This can be observed by examining the projection of an incline tool's bottom-end onto the machining surface. The projection of the inclined tool's bottom-end is an elliptical silhouette where the minor axis is affected by lead angle whilst the major axis is affected by tilt angle. Consequently the geometrical properties of elliptical silhouettes for inclined tools depend on the inclination angles for a given cutting point/tool position and their magnitudes determine the effective cutting radius which finally influences scallop height. Figure 2.7 depicts the relation between effective radii / elliptical postures and inclination angles whereas Figure 2.8 illustrates how scallop geometry may vary under different inclination angles (lead-tilt). Therefore a more advantageous geometrical matching can be achieved by employing flat and filleted end-mills compared to ball-end mills.
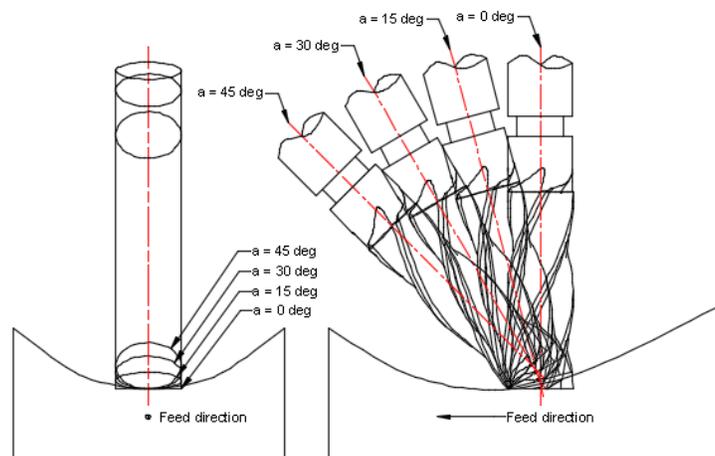
Figure 2.7: Relation between effective radii/elliptical postures and different tool inclination angles.

16
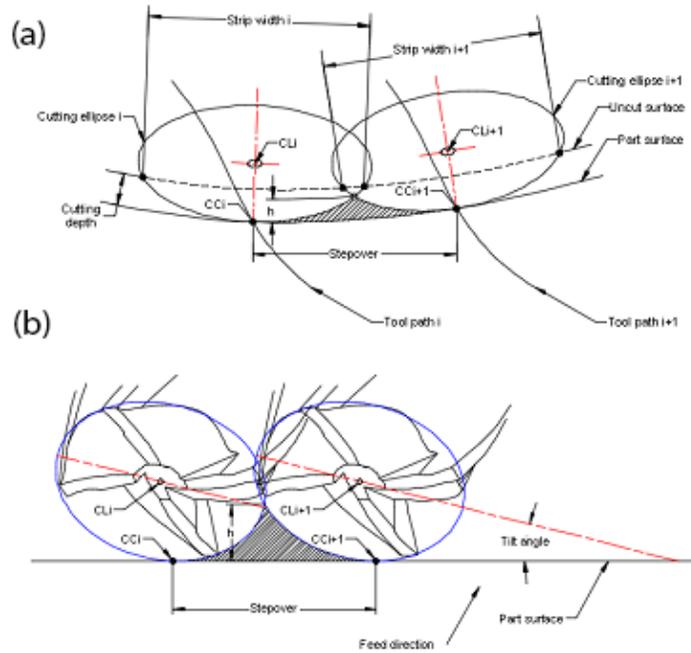
Figure 2.8: Variation of scallop geometry owing to 5-axis cutting tool inclination angles.

2.2.2 Chordal Deviation (chord error)

During surface machining operation the cutting tool's segmented trajectory deviates from the theoretical sculptured surface profile resulting to the chordal deviation (Figure 2.9). Chordal deviation is the resulting error owing to the linear segmentation of a given curved surface profile among a pair of cutting points. It is the maximum Euclidean distance between a chord whose connecting points lie on the original curve and a point on this curve (Yeh and Hsu 2002, Mayor and Sodemann, 2008).
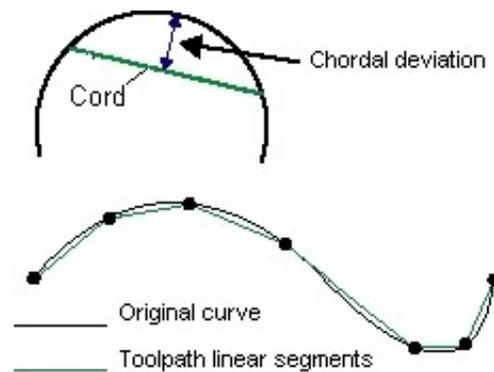
Figure 2.9: Chordal deviation between actual and theoretical trajectory owing to tool interpolation.

As an additional error to that of scallop height, it should be minimised so as to maintain machining accuracy within tolerance, yet, not at the expense of machining efficiency. This implies that proper values for profile discretisation should be used to end up with dense tool path points as much as it is required to maintain tool path efficiency as well and confront this way to the "productivity-quality" trade-off.

Chordal deviation is mainly observable when adopting conservative interpolation techniques such as linear and circular interpolation (Yang and Hong, 2002). To overcome the limitations of such conservative interpolation strategies many significant works have been focused on developing enhanced interpolation methods under the major goal of reducing large numbers of cutting points found in corresponding NC part-programs. Most noticeable ones are those referring to the non-uniform rational B-spline (NURBS) interpolators (Liu et al. 2015, Jahanpour and Alizadeh 2015, Chen and Khan 2014, Annoni et al. 2012). Nevertheless, there is enough evidence in these works to support that the special merit of implementing NURBS interpolation is found in feed acceleration capabilities and not that much in its superiority concerning machining accuracy (Sun et al. 2014, Cheng and Tsai 2004). From an industrial engineering perspective, same precision may be achieved by implementing linear interpolation as well, provided that huge NC files must be stored in NC units to enable the accurate representation of varying slope and local curvatures (Chu et al. 2012). Nevertheless, this is not of major concern given the current state of high-tech CNC controllers which have become more sophisticated and efficient while coping with large NC data, under fast processing rates (Lin et al. 2014). In addition, high frequency servo loop functions integrated to CNC systems, allow smoother machining operations whilst maintaining good transition from one move to the next, in terms of feed rate (Yang and Altintas, 2015). NURBS interpolators come with their own expensive policy as extra modules to integrate only few cutting-edge CNC units found today in industry. It has been also stated that NURBS equation to represent high order curves for tool paths can be overly complex, hence, imposing additional time to compute real-time trajectories during cutting (Mayor and Sodemann, 2008). Such aspects have already led to the reconsideration of still employing common interpolators when it comes to high-precision machining (Lin et al. 2014). Besides NURBS converters and other similar utilities are embedded to CAM software for converting an "optimised" point-to-point end milling tool path to a NURBS part program for 5-axis machining (Cheng et al. 2002).

2.2.3 Machining error

Cutting points comprising a surface machining tool path are sequentially met to position the tool according to its configuration and the properties (curvature) of the sculptured surface. With every machining step the cutting tool takes from a point to another a local scallop height and a local chordal

18

deviation are generated. The combined effect of scallop height and chordal deviation introduces the machining error (Kayal 2007). Apparently the machining surface will be characterized by many local machining errors whilst their magnitudes are depended by the overall number of cutting points, their coordinates in 3D space, the local curvatures of the surface, the cutting tool type and the trajectory followed to produce the cuts. Figure 2.10 gives a graphical depiction of the combined error of scallop height and chordal deviation, hereafter referred as machining error.
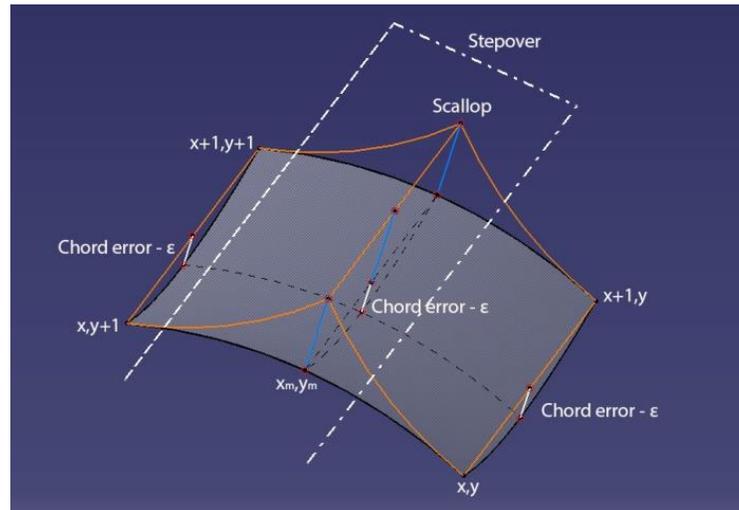
Figure 2.10: Machining error as a combined effect of scallop height and chordal deviation.

2.2.4 Machining Strip Width (MSW)

Machining strip width (MSW) is the distance taken between the fringes of two consecutive scallop curves formulated by a tool pass and it can be considered as an alternative performance criterion to that of scallop height. The larger the machining strip width is, the smaller the scallop height occurs as well as the number of adjacent tool passes. Obviously, such a result increases production rate with the simultaneous benefit of reducing the time needed for following benchwork processes.

By machining sculptured surfaces using either flat or filleted end-mills under 5-axis mode maximization of machining strip width can be maintained provided that the effective cutting tool profile closely matches the surface curvature through proper inclination regarding the surface normal (Figure 2.11a). An important aspect when studying machining strip width as a performance objective is that adjacent passes should overlap to some extend for reducing scallop height between them (Figure 2.11b), however, excessive overlap may lead to repeated cutting in limiting contours of the surface. Under this prism cutting tool type, inclination angles and tool pass interval (stepover) have to

19

be adjusted accordingly to achieve the aforementioned benefits and ensure gouge-free machining at the same time.
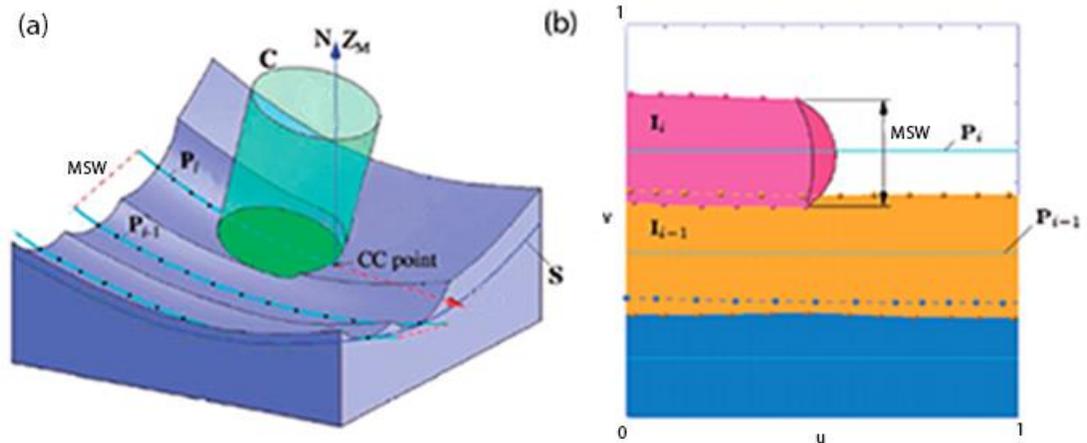
Figure 2.11: (a) Machining strip width (MSW), (b) Tool pass overlap.

2.3 Tool positioning strategies

Tool positioning, or equivalently, tool orientation strategies aim to position the cutting tool in each cutting point so as to generate the entire tool path. Such approaches constitute a large part of the research related to the sculptured surface machining problem. The benefit of properly inclining flat end-mills or filleted end-mills with regard to the surface was captured at an early stage by Vickers and Quan (1989) who presented the well-known Sturz method. According to this tool positioning method, a tool is tilted at a fixed angle in feed direction about the corresponding cutting point in the plane to where the point belongs, the feed direction and the surface normal (Gray et al., 2003). The angle of which the tool is inclined varies typically between 5 and 10 degrees (Figure 2.12).
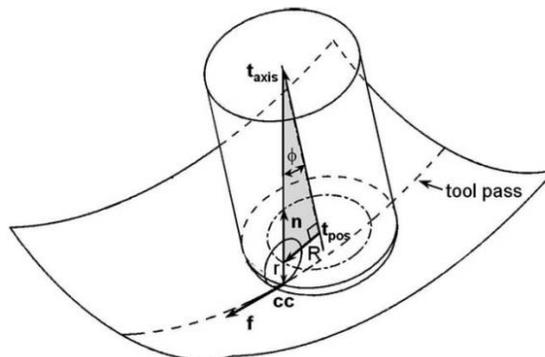
Figure 2.12: The "sturz" method for surface machining (Vickers and Quan 1989).

Undoubtedly by arbitrarily selecting a constant value for positioning the tool cannot ensure optimal orientations for the entire surface since local curvature variations are not considered. Cho et al., (1993) enhanced tool inclination by proposing the Z-map strategy. According to this strategy a finite set of (x,y) points represents the XY plane. Both cutting tool and machining surface are illustrated as z values for each (x,y) point whereas collision checking is conducted by observing whether any of the z values representing the surface exist above the z values representing the tool. With reference to the weighted average of interference points, tool positions are conducted by rotating the tool about the center contact point. If the collision problem still exists, end-user has to manually set a feasible tool orientation. To improve this approach, Li and Jerard (1994) automated the tool orientation and presented the cutting tool and the machining surface as faceted models against their inefficient representation as infinite (x,y) point sets. To perform collision checking they processed the geometrical entities models such as lines, points and planes. However, their approach accounts for gouge-free tool positions with uncertain results in terms of scallop minimization. Rao et al. (1997) proposed the Principal Axis Method (PAM) which is another tool positioning variant similar to that of Sturz method. However, PAM takes into account two principal curvatures $\kappa 1$, $\kappa 2$ at a given cutting point as well as their associated principal directions $\lambda 1$, $\lambda 2$ (Figure 13). PAM aims at matching the surface curvature of a cutting point to the projected effective radius of the inclined tool. This way proper local curvature matching is guaranteed for all cutting points in the surface, nevertheless PAM method shown major limitations in the case of saddle and convex surface contours using filleted end-mills, as shown by the experimental work reported in Rao et al. (1997).
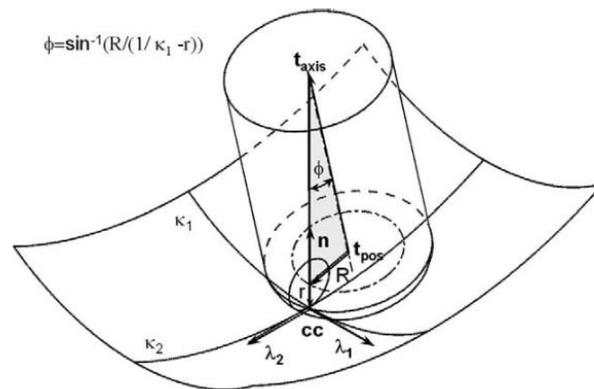
Figure 2.13: The principal axis method (PAM) for surface machining (Rao et al., 1997).

The aforementioned tool positioning strategies manage to produce subsequent cutting points that will result to cutting trajectories free of gouges and under the predetermined cut tolerance. However, machining efficiency is not recognized as an essential issue. Warkentin et al (2000) successfully

captured the necessity of generating gouge-free tool positions by simultaneously maintaining efficiency in machining passes and presented the multi-point machining (MPM) tool positioning strategy. Their algorithm undertakes a number of computations dealing with tool position, surface normal and local curvature at two cutting points instead of a single one. In first the algorithm establishes the first cutting point on a convenient surface region to where the first hypothetical cutting path is set. Thereby, a number of subsequent cutting points are generated to formulate the first cutting path. With reference to the initial set of cutting points constituting the first cutting path, the second cutting path is obtained, having a distance from the first, equal to the desired machining strip width (Figure 2.14). As a consequence, their procedure lead to machining passes with larger strip widths and, to a large extent, free of collisions. However, MPM seeks to maintain a constant separation distance between contact points that should be in the opposite side of the tool. Under such a requirement it is quite possible that the second cutting point might not be found to position the tool. Moreover, when setting the constant separation distance as a requirement to machine sculptured surfaces, sequences of flat and sharp scallops can be left on the surface owing to its arbitrary and complex geometrical variation (Gray et al. 2005).
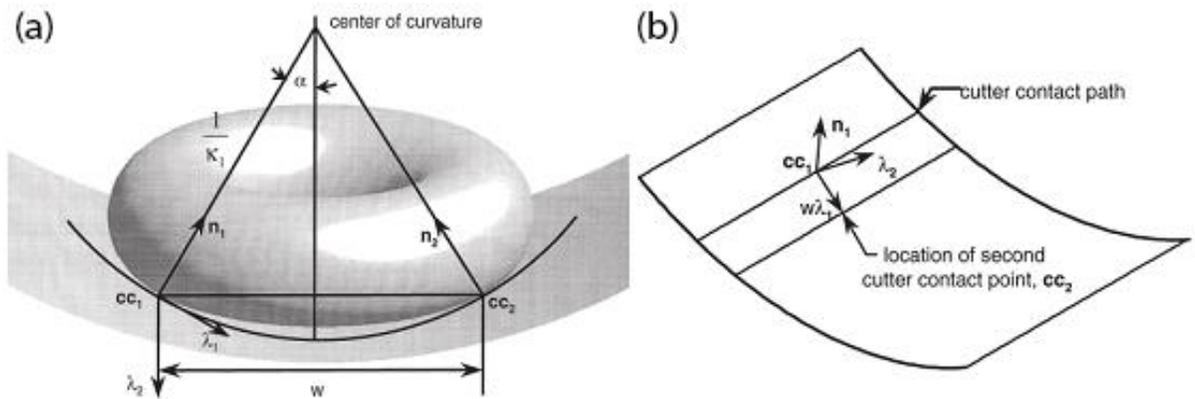
Figure 2.14: The multi-point machining method (MPM): (a) determination of MPM tool positioning, (b) path of cutter contact points in MPM method (Warkentin et al., 2000).

Gray et al. (2003) took advantage of the best features from PAM and MPM strategies to propose an enhanced and stable algorithm to position a filleted end-mill, called Rolling Ball Method (RBM). As a pure derivative of PAM and MPM, RBM inherits much of their properties and computational process. RBM implements MPM to locate the tool inside of a rolling ball. In PAM method curvature computations for a given cutting point neglecting neighboring regions, thus, imminent gouges with the surface may occur. In contrast, RBM method (Figure 2.15a) utilizes the area underneath the tool named as the "shadow checking area" (Figure 2.15b). This area is then discretized into a finite set of

points, the "shadow grid" points. A pseudo-radius of curvature is then computer for each of these points as the radius of a sphere whose center lies along the surface normal of a given cutting point and contacts both the specific cutting point and the "shadow-grid" point. With reference to a hierarchy of such radii the rolling ball's radius is selected as the most concave radius when compared to the rest of "shadow grid" points, to finally position the tool. Thereby the tool is inclined such that a circular profile is created and is in contact with the rolling sphere. Since the most concave radius is selected the tool position in the cutting point results as gouge-free (Gray et al. 2003). Through this approach the curvature of a filleted end-mill is sequentially matched to local regional curvatures of a sculptured surface and has also a built-in gouge detection function unlike to other strategies where this process is implemented separately.
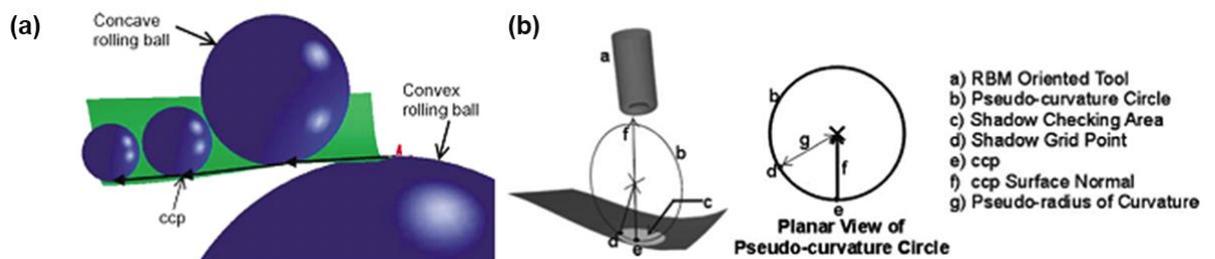
Figure 2.15: The Rolling Ball Method (RBM): (a) basic principles of RBM method, (b) The "shadow checking" area for RBM method (Gray et al., 2003).

The same authors recognized the need to further simplify their tool positioning method in terms of parametric surface equations and provide a practical tool for implementing their algorithm. Their efforts towards this direction resulted to a graphics-assisted environment for applying the RBM method (Gray et al. 2004). Their enhanced approach can be implemented to surfaces represented through triangulated data instead of surface equations and their experimental work involves the selection of a challenging sculptured surface with multiple patches to show that their method is prominent. Unfortunately, surface triangulation has been considered as an extremely complex operation owing to pre-filtering and post-processing procedures required. In addition, tool path planning for triangulated sculptured surfaces is degraded owing to the absence of curvature information in the case of polygonal models (Zhang et al. 2009). Rolling ball method (RBM) for tool positioning was applied to several experiments by its inventors. These experiments revealed that RBM overestimated the area underneath the tool to consider it as the shadow check area thus a larger region was considered for creating the rolling sphere's radius and further proceed to tool positioning. As an outcome inclination angles larger than necessary were recommended. Therefore, it

23

was clear that RBM created conservative tool inclinations under the ultimate scope of ensuring gouge-free tool orientations.

On their way to deliver a more reliable solution, Gray et al. (2005) developed the Arc-intersect method (AIM). As in the case of RBM method, AIM is a derivative of previous tool positioning methods under the philosophy of considering their beneficial attributes whilst trying to rectify their shortcomings. AIM is initialized by considering a cutting point and computing its associated surface normal either numerically or implicitly whereas feed direction is manually determined by end-user. The method employs the "shadow-check" area as well as its discretisation to set of points as in the case of RBM. Two types of constraints are applied to AIM method. The former refers to the constraint of the cutting tool to be positioned on the given cutting point whilst the latter refers to the tool axis constraint with regard to the tilting plane and the surface normal (Figure 2.16a). Major scope of this technique is to achieve a beneficial contact of the tool to a second cutting point after its contact to the first one. Unlike MPM method, AIM rotates the whole shadow grid of points until the tool is met instead of repetitively inclining the tool and checking for contact. In this algorithm, the shadow checking region is depicted as a circle and its discretisation to points is based on the transformation of rendered volumes and their corresponding pixel coordinates to Euclidean coordinates. As a shadow grid point is rotated about the cross vector it postures an arc. Thereby the arc radius of each shadow point is computed as the shortest distance between the point and the cross-vector along another vector perpendicular to cross-vector (Figure 2.16b).
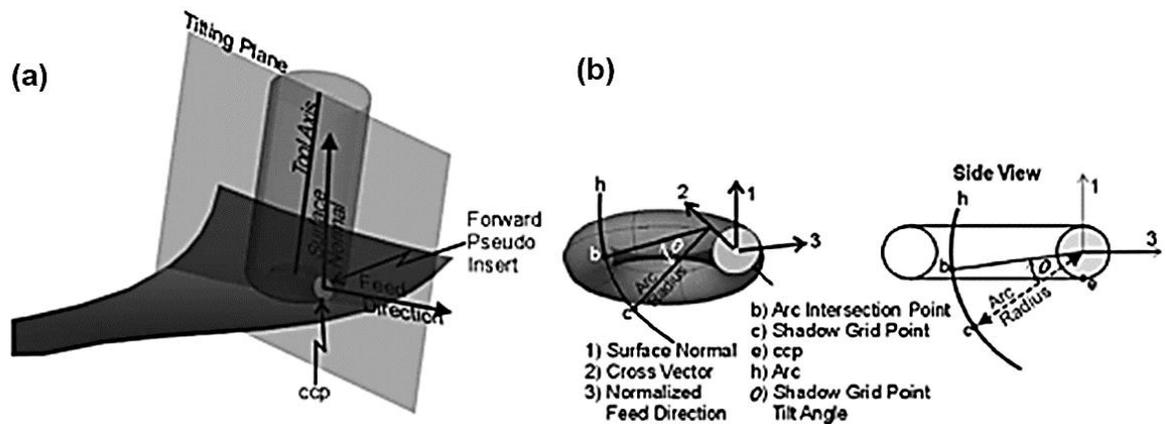
Figure 2.16: The Arc-Intersect Method (AIM): (a) tool axis and tool positioning constraints, (b) arc intersection and shadow grid point tilt angle (Gray et al., 2005).

Inclination angles are considered as the angles about the cross-vectors by which the toroidal shapes of the tool should be rotated to touch the shadow grid points. The exact values of angles are then

obtained by finding the intersections of arcs with the toroidal tool postures (Gray et al. 2005). A prerequisite of the successful tool positioning is to previously investigate which arcs are to intersect the tool since not all shadow grid points will guarantee intersection. Finally, the largest inclination angle tracked from the shadow grid of points is selected to achieve a gouge-free tool position. Apparently AIM can be considered as a multi-point tool positioning strategy since it suggests a second tool contact with the surface. Although this is a common attribute between AIM and MPM several significant differences distinguish the former from the latter and reported in Gray et al. (2005). A major drawback of AIM is the fact that it implements a fixed user-determined discretisation step to formulate the machining cuts. This implies that quality requirements cannot be satisfied since high-curvature surface contours would demand denser tool positions along the tool path.

The rotary contact method for tool positioning (RCM) proposed by Fan et al. (2012) developed by adopting an alternative way to suggest the MPM proposed by Warkentin et al. (2000). Based on this way an offset surface is generated regarding the original one. The offset distance is equal to the corner radius of a filleted end-mill. Thereby the position of the toroidal tool (filleted end-mill) on the original surface is equivalent to that between the inclined tool's elliptical profile and the offset surface. Another offset surface is then created with a distance to the first offset restricted to cut tolerance. The basic idea is to manage the tool position by taking the tool's elliptical profile and rotating it regarding a given cutting point until it touches the first offset surface regarding feed direction. The final inclination angle should be gouge-free when the toroidal shape contacts the original surface at that cutting point. A graphical depiction of the RCM of Fan et al. (2012) is given in Figure 2.17. This work wouldn't provide an integrated solution since the case of convex sculptured surface machining had yet to be investigated. As an extension to the already existing work of Fan et al. (2012) the possibility of machining convex shapes was also examined (Fan et al. 2013). However, this later work needs to be further extended to cover the case of mixed (convex, concave) sculptured surfaces.
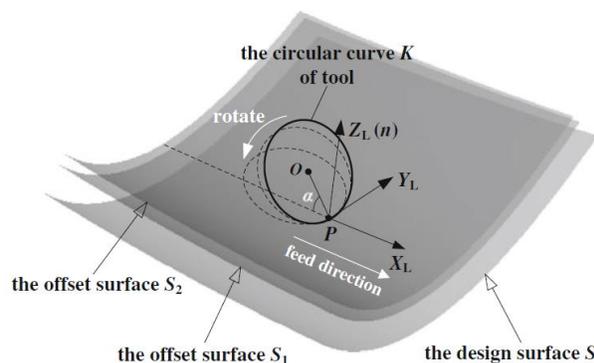
Figure 2.17: Graphical illustration of the RCM method (Fan et al., 2013).

The main principles of the several mechanisms presented in the tool positioning methods inspired Duvedi et al. (2014) to develop a multi-point machining strategy for the machining of triangulated surfaces. The method is applied as a zig-zag tool path style. In each tool position the tool is "dropped" on the triangulated surface to meet its first cutting point. A loop in their algorithm suggests the iterative check based on STL triangles in terms of finding either a tangency between the tool and the interior of a node or a tangency between the tool and any edge or vertex. Finally, the highest point of tangency among them is selected as the first point. The tool is then rotated accordingly with reference to this point on the surface to achieve a contact with a second point. Duvedi et al. (2017) extended also their approach to cover the case Bezier surfaces, yet, significant research efforts are needed to interface the method to industrial manufacturing systems and numerical algorithms should be implemented to solve surface equations for higher order curves and surfaces.

He et al. (2015) contributed to tool positioning methods by presenting a technique that not only accounts for productive machining strips during machining but also aims at reducing fluctuation in terms of their width. To achieve this goal, they examined the changes of inclination angle towards a machining strip by applying a middle point error control. This control point is in the middle of the region existing underneath the toroidal tool. Chen et al. (2017) examined the capabilities of this approach by changing its primary objective of computing maximum machining strip width to that of multi-point tool orientation. The need to switch from evaluating machining strip width to multi-point tool orientation was the ambiguity of the relation between the machining strip width and the inclination angle.

Based on the review of most noticeable tool positioning methods in the literature several conclusions - shortcomings can be derived as follows:

- Tool positioning methods require mathematical solvers for surface equations to succeed on their goal of producing advantageous tool paths for sculptured surfaces. To solve surface equations, data regarding the surface is required and their availability is not always ensured.
- If assumptions made for the surface region existing underneath the cutting tool are violated tool positioning methods may fail to deliver results.
- Tool positioning methods cannot guarantee optimised tool path planning since the latter comes as a result not only by successfully generating cutting points but considering a specific cutting strategy as well (i.e. zig-zag, concentric, 3D offset, etc.)
- Many tool positioning strategies are available for dealing with the same problem, but no assistance is provided to decide which strategy should be employed to solve it.

- Many of the methods for tool positioning are implemented using separate modules, one for generating the tool contacts with the surface and one for gouge detection. Such a philosophy increases the time needed to plan the tool path.

- Most of tool positioning strategies make use of local differential geometry to obtain results for tool positions. If the research problem is extended beyond local towards globally optimal tool path planning the benefits of local differential geometry are lost.

- Many of the tool positioning strategies utilize conservative magnitudes about crucial tool path parameters i.e. inclination angle, constant forward and/or side step (stepover) and constant separation distance among cutting points (like MPM). Such settings cannot achieve optimised tool path generation.

2.4 Intelligent techniques for optimal 5-axis tool path planning

The later research concerning the optimisation of tool paths for the CNC machining of sculptured surfaces spans several different directions distinguished by the different philosophy and perspectives they follow to solve the problem. These directions have been identified to include:

- Methodologies based on theoretical fundamentals concerning tool path planning for the development of analytical models and algorithmic procedures for prediction – problem solving.

- Systematic approaches that aim to identify the impact of influential parameters on major criteria and correlate them through design of experiments and resulting regression models.

- Methodologies based on evolutionary algorithms developed under the scope of predicting quality criteria or directly optimised them through fitness function evaluations.

2.4.1 Methods using analytical models and algorithmic procedures

The approaches falling to this category emphasize to several aspects of theoretical knowledge i.e. kinematics of machine tools and geometrical properties of cutting tools and surfaces. In general, the approaches implement programming modules, mathematics, predictive modeling and technical computing to produce efficient tool paths for the machining of sculptured surfaces using 3 and/or 5 axis CNC machine tools. Representative works of this category are those of Lazoglu and Liang (1997), Lazoglu (2003), Budak et al. (2004), Lamikiz et al. (2004) and Lopez de Lacalle et al. (2007). The aforementioned works have as a major objective the creation of analytical process models to predict

27

cutting forces in the case of 3-axis ball end-milling. An important attribute of these works is that the mechanics of processes are considered to optimise tool path generation. Nevertheless, a measurement or at least, a statistical significance test among samples of analytical and experimental data to quantify the contribution of the models in terms of their prediction capability for cutting force is not clearly reported. In addition, cutting force has been studied under the convenient case of using ball end-mills where inclined postures are always of a spherical shape regardless of the inclination angle in 5-axis milling mode.

A significant research effort based on analytical relations and algorithms for improving machining quality is that of Jun et al. (2003) who proposed the configuration space approach to optimise tool orientations in 5-axis sculptured surface machining. Their method is based on the machining error to find a set of feasible tool orientations through a boundary search module. The requirement of minimum scallop is given as a fitness function and then locally optimal tool orientations are created in a given configuration space to minimise machining error. Thereby adjacent part geometry is considered with regard to the alternative feasible tool orientations to finally end up with a globally optimised tool path. However, their algorithm had the tendency to consider as optimal, all feasible solutions for orienting the cutting tool which had to be a flat end-mill. The authors recognized that their method should consider the trade-off between machining error and its fluctuation during tool path generation to guarantee globally optimal tool path generation. Quinsat and Sabourin (2006) emphasized on the development of an algorithmic procedure to assist on the selection of optimal milling direction in 3-axis milling for sculptured surfaces. To develop their algorithm the authors took advantage of the most often-implemented cutting style, the parallel-planes, that allows sweeping the entire surface using a ball end-mill. Although it is obvious that the best direction to feed a cutting tool on a sculptured surface is always the direction with the lowest curvature, their work still contributes since actual feed rate is assessed at each cutting point. This way feed adaptation can be achieved according the local curvature among pairs of cutting points and further improve the machining operation. Giri et al. (2005) proposed a strategy to generate master cutter paths for the machining of sculptured surfaces. The philosophy underpinning this approach is that the edges or the boundaries of a surface can be utilized as trustful drive curves to construct tool passes under the perspective that intrinsic properties of all cutting points will be considered. These tool paths were recognized as muster cutter paths whilst their proper orientation was based upon the maximum convex and maximum concave curvatures for smooth surfaces, i.e., if the master cutter path is oriented towards the maximum convex curvature the side step will be studied in the direction of maximum concave curvature. Such a side step would then result as the largest one (Rong and Koren, 1996). A prerequisite for the successful implementation of the aforementioned strategy is that the surface

28

should not vary abruptly, yet, in production engineering such a premise is not always ensured. The work of Giri et al. (2005) established also the simultaneous requirement of minimizing machining time and cutting data file size (or equivalently the number of cutting points). They identified that the trade-off was not satisfied, and the problem should be set as a multi-objective optimisation problem to be handled by a suitable evolutionary algorithm. Li and Chen (2006) proposed a tool positioning method to advocate the global tool path optimisation for sculptured surface machining. Their algorithmic procedure differs significantly from the conservative tool positioning strategies because both inclination angles are considered as well as the instantaneous cutter position error in each forward step towards feed direction. In their work both postures of flat end-mills and filleted end-mills were examined. Based on the properties of virtual cutting edges of these types of end-mills their trajectories towards feed were investigated after their discretisation to linear segments whilst the instantaneous characteristic curve was deduced with the properties of motion envelopes using a mathematical formula. The goal of the work was to determine cutting tool positioning adjustments capable of extending the segments of instantaneous cutter position error curves satisfying tolerance to the longest possible extend. The first step of their algorithm undertakes a smooth and symmetrical position of the instantaneous cutter position error curve by adjusting lead and tilt angles simultaneously. The second step handles the changes in terms of Z-height distance where in the case of 5-axis machining varies significantly. The scope of this step is to maintain the length of instantaneous error satisfying the tolerance as long as possible. During the procedure each parameter change is to suggest a new computation for the instantaneous error and seek for an efficient cutting strip width. The two steps are sequentially repeated until the longest cutting strip is obtained whereas the data (x,y,z,a,b) is stored as the optimal parameters. The whole procedure is repeated to find tool positions for the entire tool pass and thereby the entire surface. In a recent work proposed by Lu et al. (2016) machining strip width was examined via an intelligent algorithm and tool orientations were sequentially computed for cutting points and tool passes under a similar fashion to that of Li and Chen (2006). Lu et al. (2016) realized that optimal solutions of next positions/passes were affected by those preceded them whereas the properties satisfying next positions could not be considered during the computation of previous ones. Based on this result they concluded that such an approach partially solves the problem and cannot be considered as a generalized solution for optimizing the whole sculptured surface. Makhanov et al. (2002) proposed a tool path optimisation approach based on a global interpolation of the required surface by a virtual surface composed from tool trajectories. Their approach was determined to opitmise tool paths of milling robots.

Noticeable contributions have also recognized CAM software as a standard environment to take advantage of its already existing utilities and examine automation potentials as per the properties of

the problem to solve. Xu et al. (2010) proposed an automated approach hosted to a commercial CAM system to generate successive cutter contact points for sculptured surface tool paths whilst the cutting tool is controlled by two guide-curves. Their approach utilized the primary curve for generating cutter contact points according to a preset tolerance whilst another group of points is created through the secondary guide-curve to complete the two-point contact towards the whole machining strip. Zeroudi et al. (2012) presented an approach for computing cutting forces by taking advantage of all tool position points regarding local inclination angle provided by a typical CAM system. Unfortunately, their work is referred to 3-axis sculptured surface machining and consequently the usage of ball-end mills. Based on this work, Zeroudi and Fontaine (2015) presented a methodology to compute tool deflection and corresponding error compensation for the prediction of cutting forces in 3-axis sculptured surface machining. Prior to these works Lee and Chang (1996) developed an automatic cutting tool selection system which was later interfaced to constitute a manufacturing modeling software. Lartigue and Tournier (1999) made efforts to characterize machining error with reference to CAM software parameters, machining direction, stepover and discretisation step in the case of 3-axis sculptured surface machining. To examine quality through machining error repetitive machining simulation tests to measure scallop heights from 3D CAM outputs need to be conducted. If the tests do not suggest promising results they need to be repeated to meet requirements. To this end, Gray et al. (2003) presented an algorithm to reduce the repetitive tasks of scallop height computations on simulated machining outputs. Their algorithm was built using a computer-graphics environment to allow the practical user interaction.

Segonds et al. (2017) presented the latest work concerning the correlation of scallop height to the effective cutting radius of filleted end-mills by considering as well as all important characteristics affecting it such as stepover distance (tool path interval) and inclination angles, lead and tilt. Their latest research was preceded by two studies (Redonnet et al. 2013, Redonnet et al. 2016) demonstrating the relation between stepover distance and effective cutting radius. Their research was based on several fundamental lemmas concerning geometrical aspects such as curvature analysis and vector algebra. Their results were theoretically generalized for any given machining strategy.

Despite the solid background on which the research of this category is based, generic solutions have not been delivered owing to the complexity of the sculptured surface machining problem, the increased number of influential parameters and the suggested trade-offs among criteria. Much of their integrity is also dependent on the several assumptions made by most related contributions and is further degraded by the bottleneck of analytical expressions to capture the interactions introduced among the parameters.

2.4.2 Methods for experimental design and regression modeling

The approaches involving design of experiments (DOE) may be considered as the most straightforward ones. According to their common principle, experiments are established and then conducted by to identify the significance of effect of independent parameters on the response under question. Such approaches may be implemented instead of analytical techniques if they fail to associate independent parameters to critical objectives. Experiment-based approaches follow the procedure of "experiment-observation-conclusion". The benefits gained by adopting DOE approaches are their direct implementation and data acquisition as well as high resolution in terms of accuracy. On the other hand their major drawbacks suggest that results might not be interpreted correctly whereas hypotheses in the form of assumptions should be set since, neither it is possible to investigate all important factors, nor the behaviour under which each one of them affects the response, is known in advance. The rationale behind their inclusion in a distinct category is their systematic methodology of defining experiments, obtaining and interpreting results. Additionally, the analysis of main effects and interactions of independent parameters on dependent objectives is also a unique characteristic of the approaches fall in this category. Most important and often-employed approaches are the response surface methodology (RSM) and Taguchi's techniques based on orthogonal arrays (OAs).

In RSM the significant parameters are utilized to develop a polynomial model in which independent variables and their numerical coefficients can predict the experiment's response under a given percentage. To find the global minimum/maximum experiments are conducted to sweep the response surface towards several directions. The model's generation involves the computation of the surface's slope and the implementation of a "steepest ascent" algorithm (Myers and Montgomery, 1995).

Taguchi's orthogonal arrays (OAs) are considered as entities for preparing multi-factorial experiments where the columns are assigned to factors, column entries correspond to factor levels and the rows designate the number of experimental runs (Taguchi 1986, Ross 1996). The implementation of an OA allows reducing the overall number of experiments, yet, without the loss of statistically significant information. Experimental results are further analyzed using analysis of variance (ANOVA).

The difference between RSM and Taguchi's OAs is that RSM investigates the behaviour of significant factors in terms of their effect on the response and generates a second-order model which is more general and practical for its usage. Instead, Taguchi's OAs aim to identify the most significant factors and their corresponding values that produce the desired effect on the response without the necessity

of developing a model. In any case, the philosophy characterizing both methodologies can be applied to any problem when the reduction of its solution domain is necessary.

Representative research works of this category aim at correlating significant machining parameters to performance objectives emphasizing mostly on quality and productivity. Gittens et al. (2005) aimed on improving machining time and surface finish by creating multi-variable polynomial regression equations relating axial depth of cut and feed rate. Their research methodology involved a design of experiments based on Taguchi's approach. 25 machining experiments were conducted using the *Renshape®* 5030 as test material, and a 10mm carbide flat end-mill. For the axial depth of cut parameter levels, values from 1 mm to 5 mm were examined whereas the operational range for feed rate was from 20 mm/sec to 60 mm/sec. Spindle speed was kept equal to 15000 rpm. On their efforts to predict surface roughness in the case of CNC face milling, Benardos and Vosniakos (2002) used an artificial neural network (ANN) to predict surface roughness in face milling. In order to provide a complete dataset for training, testing and validating the network they conducted machining experiments using the Taguchi's design of experiments method. Their experiments considered depth of cut, cutting speed, feed per tooth, tool engagement (stepover), cutting tool wear and the usage of cutting fluid. An L_{27} OA was finally selected to design the experiments. Krimpenis et al. (2005) conducted machining simulation experiments under the L_{27} OA to study the remaining volume after implementing a roughing strategy to machine sculptured surfaces. In these experiments machining time was suggested as the second objective. As significant parameters for their experiments tool path interval, tool offset distance, stepdown, profiling, feed direction, cut tolerance and the joining range between two consecutive passes along Z-axis were selected. Major scope of their work was to examine the effect of the aforementioned parameters on the objectives of remaining volume and machining time. Fountas et al. (2015) conducted machining simulation experiments to study the effect of machining strategies and related parameters to minimise machining time and surface deviation in the case of 3- and 5-axis sculptured surface machining. Saroj and Jayswal (2013) investigated several strategies to machine sculptured surfaces using a CAM system for conducting machining simulation experiments. Parameters such as cutting tool diameter, stepover, stepdown and feed rate were examined in terms of their effect on machining time for all cutting strategies involved. Stahovec and Kandráč (2013) adopted RSM to establish the mathematical relationship between scallop height as the response and depth of cut, stepover and tool diameter as the independent parameters in the case of 3-axis sculptured surface milling. Their 2nd order prediction model was generated with regard to their experimental design and could explain 79.52% of the variation. Kumar et al. (2015) presented an empirical study considering cutting forces in ball end-milling for sculptured surfaces. Their RSM experiments were followed by regression analysis and

model fitting. Three models, -one per cutting force component- were generated and used for validating their efficiency by comparing experimental to predicted results. It was shown that the maximum correlation error was 7.1%, yet, the authors recognized that the models were only valid in the case of machining mild steel as work piece material.

From the literature presented in this category the methods for designing experiments cannot lead to the direction of establishing a generalized result for solving the sculptured surface machining optimisation problem. The results from systematic approaches of designing and analyzing experiments depend on the materials and the methods that the researchers choose to apply thus they can be considered as reliable only if similar attributes are suggested to a problem (i.e. same/similar work piece material, same/similar sculptured surface geometry to test, etc). Nevertheless, polynomial models, regression equations and other empirical relations can be successfully considered as "objective functions" to evaluate and compare the performance of superior problem solving techniques such as those suggested by artificial intelligence (i.e. genetic algorithms). Representative works that have already followed this concept can be found in Rao et al. (2016), Bhavsar et al. (2015), Kuriachen et al. (2015), Garg et al. (2012), Pandey and Dubey (2012), Zain et al. (2010) and Palanikumar et al. (2009).

2.4.3   Methods using artificial intelligence

Artificial intelligence attempts to simulate the behaviour of human interaction, information processing and decision-making. Artificial intelligence approaches that have drawn the interest of researchers worldwide and see services to almost all branches of science are based on artificial neural networks (ANNs) and genetic/evolutionary algorithms (GAs-EAs). Both approaches have already interfaced to engineering software to constitute practically viable tools when dealing with real world optimisation problems.

Artificial neural networks (ANNs) are mathematical models inspired by the functional behaviour of the human brain. They can be trained through datasets of solved problems, demonstrate the ability to "memorize" them and generalize their results to a large extend for solving similar problems but not identical (Fausett 1994). These properties constitute ANNs the ideal tools to model very complex problems since neither an analytical description of the problem, nor an algorithmic procedure for its solution are required. On their goal to develop a reliable model for predicting surface roughness in CNC face milling, Benardos and Vosniakos (2002) conducted experiments with the use of Taguchi's method of designing experiments. An ANN was trained using some of the results whilst others were kept for testing and validation. Their ANN handled the most influential parameters found in face

milling, depth of cut, feed per tooth, cutting speed, radial cut (stepover) distance, cutting tool wear and usage of coolant. The responses were the cutting force components and surface roughness. Cutting tool wear and usage of coolant were treated as categorical factors and designated accordingly (i.e. "small", "average" for cutting tool wear and "yes", "no" for coolant usage). Their ANN was of a "feed forward" architecture meaning that information flows from the nodes (neurons) of input layer to the output layer with no feedback connections or loops. Different ANN architectures are presented and explained in Haykin (2009). The Levenberg-Marquardt algorithm was implemented for training the network which is a variant of the classic back-propagation algorithm (Demuth and Beale, 1998). Finally, their model was examined through the mean squared error (MSE). El-Mounayri et al. (2002) moved towards the same research direction and developed a back-propagation ANN to optimise the CNC flat end-milling process. In their work radial cut (stepover), feed rate, spindle speed, cutting tool diameter, number of flutes, axial cutting depth, rake angle and clearance angle were the input parameters whereas the statistical results for cutting force (maximum, minimum, mean and average) were considered as the output parameters. To provide a practical tool for CNC machining, they developed post-processor engines to translate "optimal" data to recognizable commands for most CNC units found in industry. Their work was further extended to cover the case of ball end-milling (El-Mounayri et al. 2005) whereas the initial ANN model was replaced by a radial basis function network (RBFN). Radial activation function can reduce training time in contrast to the back-propagation algorithm. Their new ANN was trained using the leas mean squares (LMS) function embedded in *Mathworks' Matlab*® which has been the main development platform for their experiments. A thorough comparison between radial basis function and back-propagation neural networks can be found in Markopoulos et al. (2016). Krimpenis and Vosniakos (2004) investigated the capability of developing two feed-forward ANNs so as to predict machining time and remaining volume for sculptured surface machining in the roughing stage. A design of experiments was conducted to obtain the results in the form of a dataset to train, test and validate both network models. Their experiments were conducted using CAM software so as to reduce the resources needed for real cutting examples. They concluded that the proposed approach as it was developed did not reflect a generic characteristic since the ANN was trained by the experimental results of the specific part adopted and further integration would be needed to determine product families under similar properties to cover all sculptured surface machining problem cases. Towards their initial research on presenting solutions for the 5-axis sculptured surface CNC machining optimisation problem, Fountas et al. (2014) investigated the generalization capability of ANNs by conducting machining simulation experiments by adopting cutting tool type, stepover, lead angle and tilt angle as the input parameters. Surface deviation and machining time were selected as the outputs. Their ANN was of a feed-forward type

using the Levenberg-Marquardt algorithm to train it. Mean square error (MSE) was used as the objective function. Lasemi et al. (2014) proposed a strategy to take advantage of machining process-related errors through an on-line inspection system and compensate them such that machining accuracy is maintained. Their methodology involved two ANNs. The first was trained to predict error results referring to the tangential component and the second was trained to predict the error results given to the normal component. The neuro-fuzzy interference system (ANFIS) embedded to *Mathworks' Matlab®* was employed as the development platform. Li et al. (2015) adopted a back propagation neural network to optimise energy consumption, surface roughness and machining time regarding feed rate, spindle speed, cutting depth and tool path spacing as the independent process parameters for sculptured surface tool path generation. Their network is trained using known results obtained by an experimental design.

With reference to the authors' conclusions referring to the literature presented above regarding the application of ANNs for process optimisation it is evident that further work is still needed for these tools to become promising when it comes to engineering problem solving under a global and generic essence. ANNs prove to be unstable models owing to the reason that initial weights and bias should be randomly selected each time the ANN is trained. Random selection might influence the training operation as well as the network's final error. It has been stated that although a network may reach thorough training for a given set of weights and biases it might still be prone to fail to be trained again using different set of weights and biases (Markopoulos et al. 2016). The process of building different ANN architectures is still based on experience since a standard theoretical background on selecting proper ANN architectures or developing them, is yet to be provided.

Genetic algorithms are based on evolutionary principles to search for optimal solutions in a preset search domain. Candidate solutions are coded in the form of encoded strings - chromosomes and are evaluated using a fitness function. Elite chromosomes are then selected for transmitting their characteristics to next generations through genetic operators determining this way the searching procedure in the solution domain. The stochastic nature of genetic operators affects both convergence speed and ability to escape form trapping to local optima (Goldberg 1989, Mitchell 1999). However genetic algorithms might be vulnerable to local trapping because simple genetic operators are not fully capable of sustaining the balance between exploitation and exploration rate. The former term deals with the local search of the algorithm in regions where the optimal solution may be found whereas the latter term deals with the ability of a heuristic to efficiently navigate the entire solution space. Maintaining the balance between these two parameters is mandatory to GAs-EAs because, both the advantages of rapidly searching the whole problem's space and identifying

preferable solutions near to elite ones lying in local regions, are needed (Ortiz-Boyer et al. 2005). Thus, crossover operator allows for an "in-depth" search of local regions (exploitation) whereas mutation operator undertakes the breadth search of the entire solution space (exploration). Consequently, the task of balancing these two important parameters has led researchers to use hybrids or develop several artificial intelligence variants.

Intelligent heuristics (including GAs-EAs) have already been used by researchers to solve optimisation problems. In the field of machining operations, a number of different optimisation objectives (or criteria) are also suggested. In the work of Castelino et al. (2003) a genetic algorithm has been implemented to minimise idle time for tool paths connected to linear segments. In their case the machining problem is formulated as a generalized "traveling salesman" problem with constraints and it is solved using a simple genetic algorithm. The "traveling salesman" problem suggests that a salesman should visit N cities where, each of these N cities is visited only once and then return to his/her reference city (the starting city) with the minimal cost. Oysu and Bingul (2009) moved towards a similar way of minimizing idle time during pocket milling. In their work a hybrid procedure involving genetic algorithm (GA) and simulated annealing (SA) is implemented to solve their tool path planning optimisation problem. SA algorithm is a local search algorithm and was proposed by Kirkpatrick et al. (1983). SA simulates the annealing operation of materials to reduce defects and increase the crystals' magnitude. Both are material attributes affected by its thermodynamic free energy. Thereby Oysu and Bingul (2009) took advantage of the local search that SA provides to strengthen this ability on the GA they implemented. Agrawal et al. (2006) optimised the orientation of the primary master cutter path using a genetic algorithm to further generate the rest as needed to machine sculptured surfaces. Their goal was to provide optimised 3-axis ball end-milling tool paths for iso-scallop sculptured surface CNC machining that would minimise machining time as well. In iso-scallop tool paths (Lin and Koren, 1996), stepover is determined such that a constant scallop height is achieved for the entire sculptured surface. A master cutter path is first obtained whilst the rest are created as offsets of it regarding the preset stepover. Although overall cutting length is significantly reduced using iso-scallop tool path planning, it suffers from inaccurate curvature determinations in feed direction as well as inconsistencies referring to the conversion form Cartesian to parametric space (Lee and Yang 2002). In addition, sculptured surfaces need to be defined by their corresponding equations either implicitly or explicitly (Kayal 2007). Ülker et al. (2009) developed an artificial immune (Clonal-G) algorithm (De Castro and Von Zuben 2002, De Castro and Timmis 2002) to compute cutter contact points for optimal tool paths with reference to a predetermined tolerance in the case of 3-axis surface machining. Their work first focused on finding ideal steps in the first parametric direction (let it be "u") and generated a population of "yet to visit" points on a curve to drive the tool path.

Through the Clonal-G algorithm they evaluated sequentially the separation distances among pairs of cutting points and ensured that cut tolerance is satisfied. According the input tolerances their algorithm was then applied to the other parametric direction (let it be "*v*") to produce the new curve on the sculptured surface. Under this scheme tool path curves *i* and *j* were generated after each algorithmic evaluation for "*u*" and "*v*" parametric directions that should be intersected on the coordinate *S[(u(i), v(j)]* respectively. Despite the application of such a sophisticated artificial module the problem was formulated by subsequently creating targeted cutting points to satisfy cut tolerance whereas machining time or another equivalent criterion to suggest productivity was not involved. Moreover, their work handled only 3-axis tool path planning (ball end-milling) with no recommendation of its application to more advanced cases such as 5-axis surface machining.

Sculptured surface machining should be formulated as a multi-objective optimisation problem owing to the number of significant tool path parameters and the different optimisation criteria one may distinguish. Owing to the inherent trade-off among the different quality objectives multi-objective representation is reasonable for the case. In fact, there is no unique optimal solution for such problems since many candidates are generated. In order to achieve a hierarchy among different solutions the multi-objective Pareto optimal approach is generally preferred (Coello et al. 2002). According to its principle a solution dominates another if none of its objectives involved is inferior and at least one objective value is better. Solutions obtained by adopting this approach are known as Pareto optimal or non-dominated solutions. Therefore, a generic approach for solving the sculptured surface CNC machining problem should provide a set of solutions reflected to an entire Pareto front.

Kersting and Zabel (2009) proposed an intelligent solution for solving the 5-axis sculptured surface CNC machining problem under a multi-objective fashion. Their work involved a genetic algorithm that handled the minimization of the cutting tool's positioning deviation as the first objective and the minimization of tool trajectory fluctuations as the second objective. Unfortunately, their problem formulation was based on the control of multiple degrees of freedom referring to normal vectors. To accomplish this under a reasonable time span, they discretized the surface and optimised its corresponding tool path region-by-region to deal this way with smaller NC paths. Inevitably this idea turns the initial problem to many separate optimisation problems that need to be handled either at once (i.e. via parallel computing) or subsequently. Although such an approach can be considered as generic, its problem formulation with global criteria is absent. The number of tool orientations for the discretised tool path may significantly vary from few hundreds to thousands as the authors themselves advocate, hence processing time can be varied accordingly.

"Pareto optimal" method neglects important engineering aspects such as expert's knowledge, technical background, process planning time span available in the shop, etc., whilst the expert should be able to select between non-dominated solutions with regard to his/her technical background, specialized knowledge and specific needs-requirements of production. From a practical perspective engineers may prefer one optimal solution among the rest. Under such circumstances the multi-objective problem formulation may be facilitated to some extend by adopting the "weighted objectives" technique. The technique simplifies the multi-objective problem to a single-objective one by expressing it as a linear combination among the problem's objectives and their associated impact weights. Impact weights determine the importance of each objective to the total cost. The sum of impact weights referring to their corresponding objectives cannot be above 100% thus, each impact weight is determined as per the practical requirements of the expert or production.

Manav et al. (2013) presented an intelligent approach of selecting tool paths to machine sculptured surfaces based on the concept of the weighted objectives technique. In their work the sculptured surface machining problem is treated as a triple-bounded problem with mean cutting force, mean scallop height and machining time as the optimisation objectives. Mean cutting force is the result of several experimental samples-outputs estimated by a force-prediction model (Lazoglu and Manav 2009). thus, it suggests a global depiction of cutting force objective. Mean scallop height is adopted in their work as the global objective to represent the overall depiction of scallop height measurements computed by a 3D scallop model. These measurements were validated by direct measurements taken on CAM outputs for the same sculptured surface examined. Unfortunately, the methodology has been applied to 3-axis ball end-milling for sculptured surface machining with unknown potentials concerning its application to 5-axis machining.

Lu et al. (2017) implemented recently a differential evolutionary algorithm (DE) to solve the sculptured surface machining problem using a flat end-mill. The authors realized that significant tool path parameters should be simultaneously handled when it comes to a global and generic problem formulation for the sculptured surface machining optimisation problem. In their methodology, lead/tilt angles and feed directions at each cutting point are involved and optimised as a whole to maximise machining strip width with less overlap, maintain tool path smoothness and reduce scallop height. Their three-objective optimisation problem formulation involves a criterion for feed direction, a criterion for cutter location curve and a criterion for tool orientation. The first criterion is formulated regarding the forward step which is affected by its corresponding chord length during the interpolation. The second criterion deals with the cutter location curve and its associated cutting points. According to the authors, the curvature of this curve may dictate a metric for tool path

smoothness whilst the curve's curvature is computed after passing the tool from all cutting points along the curve's trajectory. Finally, the third criterion reflects the metric for tool orientation results, and it involves the angle between pairs of adjacent tool orientations. A differential evolutionary algorithm is utilized to evaluate the weighted summation among the aforementioned independent tool path parameters and their associated impact weights and this weighted summation is the objective function for their algorithm. Their evaluation procedure is conducted by following a number of steps. Adjacent paths are then created with reference to the optimised primary cutter location curve whilst they are subjected to constraints dealing with minimum-maximum overlap and minimum-maximum gap. Despite the potentials of these constraints to keep all cutting paths comprising the entire tool path under advantageous connections with low overlapping and as much gap as necessary, is not explained. Therefore, the ranges of constraints should be assumed as they are seen fit. It appears that the restriction to overlapping and gap conditions in order to compute adjacent paths, seems to suggest prior work for conducting experiments or trial-and-error tests to identify the exact values for these parameters. In such a case it is not known to what extend optimality is ensured. In addition, the approach is case-oriented to flat end-mills since the requirement for smooth tool path postures will be violated when dealing with postures produced by filleted end-mills which are completely different. In their work the authors comment on DE algorithm and advocate that may be prone to local trapping when dealing with large-scale optimisation problems. To ensure guaranteed convergence to global optimum they implement a sequence linear programming (SLP) algorithm and prevent local trapping. As regards adjacent tool path overlapping the work could include also the parameter of path interval instead of trying to weight a hypothetically targeted pass-overlap magnitude. After all, stepover parameter has the possibility of its adjustment by considering cutting tool's diameter both as a percentage of it and as an overlap magnitude in most commercially available CAM systems. As a final comment in the work of Lu et al. (2017) the usage of Pareto optimal method could be incorporated at least for academic purposes to present and exploit a full-spectrum multi-objective optimisation problem concerning sculptured surface machining.

2.5   Conclusions on the state-of-the art

Sculptured surface machining (SSM) is a well-established manufacturing field for which numerous individual strategies, methodologies and approaches have been proposed, including those reported in the literature presented in this Chapter. Nevertheless, several flaws are observed when it comes to both problem formulation and multi-objective optimisation aspect. The flaws of the already suggested optimisation techniques are stated as follows:

1. The majority of tool path optimisation strategies concerning the sculptured surface machining problem aim at generating the tool path point-by-point or pass-by-pass by neglecting the cutting strategy. Tool path optimisation should be achieved regarding the tool path strategy to ensure complete surface coverage, hence, obtaining optimised cutting positions or adjacent cutting passes individually is a philosophy away from delivering globally optimal tool paths.

2. Aspects adopted by the different existing techniques for optimizing the sculptured surface machining problem do not suggest a generic multi-objective solution from the perspective that only few parameters considered for optimisation whilst others are constrained to fixed magnitudes depending the case. Such attempts do not introduce a design space that will reflect a generic search domain. The same also goes for the different criteria selected to formulate the problem where the generic character is yet to be given to properly solve it.

3. Assumptions and constraints suggested for solving the sculptured surface machining optimisation problem renders noticeable approaches found in the literature to appear inadequate, although some of their properties may lead to reduced computational cost. Such assumptions/constraints do not only lead to partial problem solving but they jeopardize the practical validity of these methodologies and consequently the very essence of optimality of the solution.

4. By taking into consideration the current needs for versatile systems and production requirements, approaches that exhibit automation utilities seem to be favored against others. However only few works in the literature report automation capabilities to facilitate proper interaction among their modules and functions towards the ultimate goal of solving the multi-objective optimisation problem referring to sculptured surface machining.

5. Different methodologies exist to solve the same problem but human intervention is still required to decide which of them should be implemented and what parameter settings should be set.

Formulating the sculptured surface machining problem so that generic aspects are introduced for global optimisation is a complex task. Major reasons for this complexity are the different properties of sculptured surfaces, the variety of cutting tool types and the variable operational ranges for

40

determining tool path planning parameters, the non-unique relation among parameters to optimisation criteria, etc. By considering the limitations of already existing works proposed for solving the sculptured surface machining optimisation problem, this work comes to the important conclusion that the perfect algorithm to directly generate optimal tool paths to machine sculptured surfaces might never exist.

Instead, this thesis presents a methodology where a typical CAM environment is adopted whose functions are part of a generic, stochastic and global multi-objective optimisation methodology for solving the sculptured surface machining problem. CAM system's functions are automatically handled by a virus-evolutionary genetic algorithm through external programming to iteratively evaluate pairs of tool path parameter values as candidate solutions (encoded chromosomes) related to specific generic optimisation criteria. The generic character of optimisation criteria dealing with quality and efficiency is derived from local information of cutting points generated by the tool path strategy itself. The values for tool path parameters are subjected to applicable ranges that may vary according the needs of the operation and tool path planning characteristics. The ranges of tool path parameters lead to the establishment of a feasible solution space from which optimal solutions are to be found. Optimal results are presented by adopting both Pareto optimal and weighted summation techniques.

Such an optimisation methodology has neither been proposed nor applied so far and therefore its different philosophy as well its components proposed for implementation are original. In addition, it inherits the necessary fundamentals and aspects of previous successful research proposed on solving the sculptured surface machining optimisation problem. Original attributes of the research presented in this thesis may be distinguished by considering the perspectives of production engineering and evolutionary computation. These original attributes are as follows:

- The methodology proposed, provides a fully automated and user-friendly infrastructure to be directly transferred to industry,
- The methodology proposed, handles simultaneously the significant tool path parameters under variable encoding accuracies and regardless of their heterogeneous properties,
- The methodology can be applied to any sculptured surface which can be supported by a CAD interface.
- The methodology develops a multi-objective evolutionary algorithm based on the virus theory of evolution and deploys it as the optimisation module of the proposed methodology.

Chapter 3

Sculptured surface CNC machining problem definition

3.1   Introduction

In the case of sculptured surfaces, the effects and interactions among 5-axis tool path parameters may vary depending on the complexity of the surface. For this reason, CAM software constitutes the only safe and low-cost environment to investigate how 5-axis tool path planning parameters influence optimisation criteria. Since 5-axis machining of sculpted surfaces introduces different combinations among tool geometries, tool orientations positions, forward step values and tool path intervals, consideration should be given to the influence of the parameters on the set optimisation criteria.

This can provide important information both for the successful modeling of the problem and for the development of the optimisation methodology. Thus, the investigation of 5-axis tool path planning parameters is considered necessary whilst it should be carried out systematically by one of the available methods for designing experiments. The first step in such a process is to determine applicable ranges for tool path parameters that will be examined for a number of different experimental sculptured surfaces.

This chapter aims to investigate the 5-axis tool path planning parameters in terms of their significance to the objectives considered and to formulate the sculptured surface machining problem according to them. This activity assists on understanding the problem's main attributes and defining it clearly to take corresponding results into account to establish the optimisation methodology.

3.2 Problem definition

The problem is defined for the 5-axis sculptured surface CNC machining as it has already been mentioned. Since it is a pure finish-machining operation, machining accuracy requirements are most important and therefore any meaningful optimisation effort should aim to optimise criteria related to them. The problem is formulated to optimise 5-axis tool path planning parameters regardless of the conditions under which the roughed part is being under. The definition of the problem does not impose any standard limitations in terms of the applicable tool path parameter values; however, it is established by considering a machining modeling document already prepared (process planning).

The problem is formulated by representing a set of 5-axis tool path parameters under a machining strategy as candidate solution for heuristic evaluation. The set comprises an integer value for cutting tool designation and four decimal values for tool path interval (stepover distance), lead angle, tilt angle and maximum discretisation step. In other words, the entire machining strategy is considered as a string of values, consisted of the parameters responsible for its control. To generalize it, let n be the number of tool path parameters (*Prm*) of a machining strategy *MS* for assessment. The candidate solution *CS* for the machining strategy *MS* can be given as follows:

$$CS_{MS} = \left\{ \text{Prm 1; Prm 2; Prm 3; } ….. ; \text{ Prm } n \right\}$$
Eq. 3.1

Each of these parameters existing in the candidate solution needs to be optimised such that the rest of parameters' effects are also considered and thus, the optimal solution (as a sequence of parameters) is subjected to be obtained as an entire tool path for the machining strategy. From an evolutionary computation perspective this is interpreted as the process of evaluating a chromosome with regard to some optimisation criteria such that the tool type, the tool's orientation and the spacing of cutting points referring to both directions (u and v) are simultaneously considered. The identification of variables and their applicable ranges determine the problem's domain. Since the problem is to be solved using an intelligent evolutionary algorithm, the search step needs to be defined. This step will specify the accuracy of the search carried out by the algorithm whereas this accuracy is decided via the number of digits associated to the parameters' binary representations. By considering the number of accuracy digits (a_{bits}) of each tool path parameter in the machining strategy and the constrained minimum and maximum numbers of digits (min_{bits}, max_{bits}) referring to the entire candidate solution CS_{MS}, then the expression presented in Equation 3.1 is transformed to the one given in Equation 3.2.

$$CS_{MS} = \left\{ \left\langle [\text{Prm } 1] a_{bits}^1 \right\rangle, \left\langle [\text{Prm } 2] a_{bits}^2 \right\rangle, \left\langle [\text{Prm } 3] a_{bits}^3 \right\rangle, \ldots, \left\langle [\text{Prm } n] a_{bits}^n \right\rangle \right\},$$

$$\underset{\min_{bits}}{\overset{\max_{bits}}{a}} \in \left[\min_{bits}, \max_{bits} \right] \qquad\qquad \text{Eq. 3.2}$$

The tool path parameters involved in the machining strategy are evaluated to converge to a set of non-dominated solutions according to the Pareto optimal approach and the corresponding number of optimisation criteria. Let S_{SSM} to depict the objective domain of the multi-objective sculptured surface machining optimisation problem, $s \in S_{SSM}$ a non-dominated point, $c \in \{1, 2, \ldots, C_{\max}\}$ a single optimisation criterion and $C_{\max}$ the maximum number of the criteria. If the candidate solution CS_{MS} is considered a vector of decision variables (tool path parameters) as well, then the multi-objective optimisation problem for the sculptured surface machining can be expressed via Equation 3.3.

$$S_{SSM} = \left\{ s \in R^{C_{\max}} \middle| s_1 = f_1(CS_{MS}), s_2 = f_2(CS_{MS}), \ldots, s_{C_{\max}} = f_{C_{\max}}(CS_{MS}) \right\} \qquad \text{Eq. 3.3}$$

In Eq.3.3, $f(CS_{MS})$ designates the feasible objective domain of candidate solution CS_{MS} and is associated to the corresponding feasible decision domain for the CS_{MS}.

3.3 Machining strategy and cutter location points

Multi-axis sweeping is a global surface machining strategy available to *Dassault Systemes CATIA® V5*. The strategy produces parallel tool paths to the plane defined by feed direction (*F*) and view direction (*V*). Like in any other multi-axis strategy for surface machining, the cutter location (*CL*) points generated introduce a variable cutting tool orientation along the tool path with respect to the surface. Thus, the values of tool path parameters, type of cutting tool, stepover, lead and tilt angles and maximum discretisation step, alter the resulting work piece-engagement boundaries at each of cutter contact points, suggesting different tool path postures.

Cutter location data formulate a *m x n* pattern of points covering the entire sculptured surface represented in the *u, v* parametric space. A unique cutter location (CL) point is determined as *CL* (*x, y, z, i, j, k, c₁, c₂*), where (*x, y, z*) are the coordinates of the machining axis system whilst (*i, j, k*) are the components determining the unit normal vector representing the tool's position for a given CL point. Finally, c_1 and c_2 are the two principal curvatures of the surface for *u* and *v* respectively, responsible for the tool's inclined position to the CL point. The aforementioned instances play crucial role to a

multi-axis tool path definition since they affect the entire machining strategy's cutting style in terms of quality and efficiency.

3.4 Optimisation criteria definition

There are several possible alternative solutions to efficiently produce a specific product, each of which has a given manufacturing cost. In order to have a clear depiction whether a solution outperforms another, specific criteria should be determined. As it has been already mentioned, the primary objective of 5-axis sculptured surface machining is machining accuracy and surface quality. Although it is a metal cutting operation dedicated to finish-machining, it comes inevitably with excess material in the form of remaining volume. This volume is the direct result of machining error as a combined effect of scallop height and chordal deviation and it should have specific characteristics to accept the part from a production perspective. In addition, the topology of the material left owing to machining error should facilitate last operations of benchwork and polishing.

The criteria or objectives determined to solve the sculptured surface machining problem in this thesis are the machining error, the uniformity of machining error (tool path smoothness) and the number of cutting points comprising a tool path. To enable their evaluation for globally optimising the tool paths, local data is collected referring to each of the cutting points constituting the entire tool path and then results corresponding to machining error and uniformity are represented by their true means respectively. The number of cutting points is obtained by the cutting data file (APT source or CL data file). A prerequisite for using the means of measurements to represent global criteria is to consider all cutting points and not only a sample of them. The criterion of machining error uniformity is introduced in the problem to characterize the smoothness for tool paths when varying the tool axis along feed direction during the cut. Fluctuations of machining error indicate the abrupt changes among different tool orientations and consequently they may prevent machining strips from having smooth tool path postures. The beneficial control in terms of tool axis variations when machining sculptured parts is also a technical requirement since collisions between the tool and the work piece can be prevented.

It is clear that a trade-off for this triple-bounded problem is introduced and should be properly treated to positively judge multi-objective optimality. Machining error is an efficiency-opposed criterion since dense cutting point patterns reduce the error but increase machining time whilst it is uncertain whether a low machining error will maintain uniformity. Nevertheless, there should be at least one region in a Pareto front where the three criteria are simultaneously satisfied, that is, an

adequate number of cutting points for maintaining low and uniform machining error to the lowest possible processing time. In order to perform global multi-objective optimisation, tool path parameters should be simultaneously investigated to include their effects on the optimisation criteria, at once.

3.4.1   Machining error

As the tool follows the tool path towards feed direction, it subsequently meets cutter location points with the consequence of producing sequential chord errors whose magnitude depends on the 3D distance $L_{i,i+1}$ and the local curvature $\rho_{i,i+1}$ existing in between pairs of unit normals $\overrightarrow{n_i}$ and $\overrightarrow{n_{i+1}}$. If CL_i, CL_{i+1} are considered as two consecutive cutter location points then their chord length $L_{i,i+1}$ in 3D Cartesian space with reference to the machining axis system can be computed using Equation 3.4 (Fisher 1989).

$$\left| L_{i,i+1} \right| = \sqrt{\left(x_{i+1} - x_i \right)^2 + \left(y_{i+1} - y_i \right)^2 + \left(z_{i+1} - z_i \right)^2}$$

Eq. 3.4

Consecutive local curvatures may be computed by employing vector algebra and retrieving dot products of normal vectors utilizing the angle between them. Thereby, with reference to the two unit normals $\overrightarrow{n_i}$ and $\overrightarrow{n_{i+1}}$ the angle $\theta_{i,i+1}$ is determined as, $\theta_{i,i+1} = \arccos(\overrightarrow{n_i} \circ \overrightarrow{n_{i+1}})$, whereas local curvature $\rho_{i,i+1}$ (mm^{-1}) is computed by using Equation 3.5. Finally, the chord error $\delta_{i,i+1}$ (mm) is given in Equation 3.6.

$$\rho_i = 2 \times \sin\left(\frac{\theta_{i,i+1}}{2} \right) / L_i$$

Eq. 3.5

$$\delta_{i,i+1} = \rho_{i,i+1} - \sqrt{\rho_{i,i+1}^2 - \left(\frac{L_{i,i+1}}{2} \right)^2}$$

Eq. 3.6

Effective cutting postures differ regarding the cutting tool geometry and inclination in each cutter location point. The postures result to significant fluctuations of scallop height. Redonnet et al. 2013 and Segonds et al. 2017, managed to define the effective cutting radii R_{eff} by considering the three most often-used cutting tool geometries, flat-end (Equation 3.7), fillet-end (Equation 3.8) and ball-end (Equation 3.9) as well as the two inclination angles, lead and tilt. Note that for ball end-mills, R_{Eff} is not affected by tool inclination angles in 5-axis machining and is equal to the radius of the tool.

$$R_{eff}^{F} = \frac{R \times \cos^2 a_T}{\sin a_L \times \left(1 - \sin^2 a_T \times \sin^2 a_L\right)}$$

Eq. 3.7

$$R_{eff}^{T} = \frac{(R - r) \times \cos^2 a_T}{\sin a_L \times \left(1 - \sin^2 a_T \times \sin^2 a_L\right)} + r$$

Eq. 3.8

$$R_{eff}^{B} = R$$

Eq. 3.9

where,

$R_{eff}^{F}, R_{eff}^{T}, R_{eff}^{B}$: the effective cutting radius for flat-end, filleted-end and ball-end mills,

R : Radius of cutting tool,

r : Corner radius of filleted (toroidal) end-mills,

a_L : Lead angle in degrees,

a_T : Tilt angle in degrees.

Scallop height magnitude can be estimated by considering the effect of stepover, with reference to the effective cutting radius depending on the cutting tool geometry as follows (Segonds et al. 2017):

$$h = R_{eff} - \sqrt{R_{eff}^{2} - \frac{a_e^{2}}{4}}$$

Eq. 3.10

In order to achieve high accuracy for tool path generation, both chord error $\delta_{i,i+1}$ and scallop height h need to be controlled. Both chord error and scallop height formulas were examined for their accuracy and they were programmed via a "FOR-NEXT" function developed in *Visual Basic*®. The function retrieves the 5-axis machining strategy from the machining modeling document and executes the APT generator to produce the corresponding CL file. Thereby, the file is accessed as a *.txt file and tool positions (x, y, z, i, j, k) are examined for computing all local chord errors and scallop heights. From the entire set of computations, the mean value of machining error is finally obtained to represent the general error for the tool path. The "FOR-NEXT" function developed for automatic machining error computations is a part of an integrated programming module which is presented in Chapter 4.

3.4.2   Machining error uniformity (distribution)

Machining error criterion is mandatory for evaluating the geometrical accuracy of 5-axis tool paths to machine sculptured surfaces. Apparently, the values recommended as "optimal" will be under the preset tolerance. However, some local distances among pairs of subsequent tool path points might produce exceeding chord errors capable of affecting the smoothness of tool path trajectory (Zhou et al. 2015). In addition, the topology of tool path points may impose the sudden change in tool postures that eventually affect NC controllability.

In order to quantify machining error changes in local cutting points, the uniform distribution of machining error has been added to the optimisation problem as the second quality requirement. To evaluate the local error distribution regarding cutting points of the tool path and obtain a general outcome, statistical rules were implemented. Thus, the variance or standard deviation of the absolute difference among mean machining error and individual local machining error measurements was selected to formulate the second optimisation criterion and improve the tool path smoothness.

To implement the aforementioned optimisation criterion and allow for the necessary computations a second "FOR-NEXT" function developed in *Visual Basic*® was applied, which is also a part of the integrated programming module to be presented in Chapter 4. The statistical formulas involved to the programming procedure are given in Equations 3.11, 3.12, 3.13 and 3.14 and refer to the mean, the mean difference, the variance and standard deviation respectively.

$$\overline{x} = \frac{1}{v} \cdot \sum_{i=1}^{v} x_i$$
Eq. 3.11

$$\overline{x}\_diff = \frac{1}{v} \sum_{i=1}^{v} \left| x_i - \overline{x} \right|$$
Eq. 3.12

$$s^2 = \frac{1}{v-1} \cdot \sum_{i=1}^{v} (x_i - \overline{x})^2 = \frac{1}{v(v-1)} \left(v \cdot \sum_{i=1}^{v} x_i^2 - \left(\sum_{i=1}^{v} x_i \right)^2 \right)$$
Eq. 3.13

$$s = \sqrt{\frac{1}{v-1} \cdot \sum_{i=1}^{v} (x_i - \overline{x})^2} = \sqrt{\frac{1}{v(v-1)} \left(v \cdot \sum_{i=1}^{v} x_i^2 - \left(\sum_{i=1}^{v} x_i \right)^2 \right)}$$
Eq. 3.14

By deploying the property of machining error distribution as a quality objective via the standard deviation to assess tool path smoothness, the error's variability and, indirectly, local fluctuations of tool path smoothness are quantified for the entire tool path with reference to resulting tool orientations at all CL points comprising it.

3.4.3 Density and topology of tool path points

The number of cutting points comprising the tool path has been introduced as the third optimisation criterion since this number directly affects machining time. Therefore, optimal tool paths should contain an adequate number of cutting points for satisfying machining accuracy but not at the expense of machining time. Obviously, a varying density among cutting points is required according the surface curvature to sustain a topologically equivalent machining error variations among different regions of the surface.

3.5 Objective function

The objective function evaluates a candidate solution's state in an optimisation process and in that sense, it is used for representing the solution's quality/contribution. The objective function of each candidate solution is individually computed; thus, objective function values differ in a set of solutions. Objective function suggests the very first research step for formulating an optimisation problem.

Two widely applied techniques for formulating objective functions for optimisation problems exist. The first method is known as Pareto-optimal and deals with a set of non-dominated solutions depicted in a Pareto front. Each axis in a Pareto front depicts an optimisation objective, that is, the number of axes is equal to the number of optimisation objectives. Pareto-optimal is a pure multi-objective optimisation technique. According to its principles, a point $p^0 \in S_{Obj}$ is considered as a "Pareto-optimal" result or non-dominated result if (and only if) there is no other result in the objective space, let it be $p \in S_{Obj}$ which outperforms p^0 with regard to all objectives. In the objective space S_{Obj} the entire set of non-dominated solutions formulate a border, known as the "Pareto" front.

The impact of Pareto-optimal points in the objective space S_{Obj} is distinguished through their location in the Pareto front with reference to the coordinate system origin. Given the nature of the objective and the problem -minimization or maximization- each point's impact is given by its related distance

from the axis that corresponds to the objective. If the objective needs to be minimised then the point is preferred to be closer to the axis' origin and vice versa.

The second most popular technique for formulating objective functions is the weighted sum. In the weighted sum technique, the multi-objective treatment of criteria is turned to a single-objective one through the linear combination of the criteria involved as well as their corresponding weights of importance. In order to remove the inherent bias among different criteria magnitudes and avoid disorienting results, normalization should be carried out (especially when it comes to contradictory criteria) by mapping their objective space using the same percentage scale.

The weighted sum with respect to the single criteria and their weights is depicted in Equation 3.15, where $C_1^{nrm}, C_2^{nrm}, ..., C_n^{nrm}$ are the normalized optimisation criteria, $w_1, w_2, ..., w_n$ are their corresponding impact weights, whilst, $\sum_{i=1}^{n} w_i = 1 \big| 0 \le w_i \le 1.0$.

$$OF_{WS} = w_1 \times C_1 + w_2 \times C_2 + ... + w_n \times C_n \ , \ \ w_i(i = 1, 2, ...n) \qquad \text{Eq. 3.15}$$

The weighted sum technique seems to facilitate the optimisation process from a practical viewpoint, however it is not known whether it can ensure a clear and concise depiction in terms of the problem's global scale or its generic characteristics. An important drawback is also the absence of a scientific or standard philosophy for determining the weights of optimisation objectives thus, researches have to figure out how to decide the impact of each objective in the problem at hand (Das and Dennis, 1997).

In this thesis both techniques have been implemented and the objective functions have been designed accordingly. By considering the optimisation criteria presented above the objective functions are given in Eq.3.16 and Eq.3.17 following the Pareto-optimal and the weighted sum techniques respectively. Note that in the case of the weighted sum technique normalization of criteria has been conducted by using the maximum value among experimental results which are to be presented in the following sections.

$$OF_{PO} = \sqrt{\left(\bar{\delta} + \bar{h}\right)^2 + \left(\overline{stdev\delta} + \overline{stdevh}\right)^2 + \left(\sum CL\right)^2} \qquad \text{Eq. 3.16}$$

$$OF_{WS} = w_1 \times \left(\overline{\delta_i^{nrm}} + \overline{h_i^{nrm}}\right) + w_2 \times \left(\overline{stdev\delta_i^{nrm}} + \overline{stdevh_i^{nrm}}\right) + w_3 \times \left(\sum CL_i^{nrm}\right) \Big| w_1 = w_2 = w_3 \ \square \ 0.33$$

Eq. 3.17

3.6 Design of experiments

For an optimisation methodology to address the variables of the problem at hand in a generalized essence based on the criteria involved, it should somehow emphasize on the impact and the effect of these variables. This practically means that if the variable's effect on the interrelated combination of objectives has a specific/fixed impact regardless of the particularities of the sculptured surface being studied, then the design of the problem's solution domain should be such that it accounts for, and describes the range of applicable values for this variable accordingly, through more or less precision digits.

Additionally, analytical expressions presented for estimating chord error and scallop height values should be experimentally tested to clarify the percentage of successful predictions they provide for the objectives. Such an experimental process requires the exploration of tool path parameters and optimisation criteria on different sculptured surfaces with variable complexity characteristics so that results will technically contribute to the design and development of the generic methodology for globally optimizing the sculptured surface machining problem. Thereby, the methodology may achieve the appropriate representation of the problem's solution domain and adjust the optimisation requirements for tool path parameters accordingly, for any case of sculptured surface.

Based on the above, experiments were designed and conducted on four different sculptured surfaces and the corresponding results were statistically examined to determine the influence of tool path on the optimisation criteria. The sculptured surfaces studied are benchmark sculptured surfaces taken from the literature whilst they have been used in the past by other researchers. Thus, the ranges of applicable values for tool path parameters -as well as other related attributes- were determined on the basis of the proposed data that previous research works provide, so as to compare the proposed methodology as rigorous and as valid as possible with them.

The objectives of the following study are:

- To examine the effects of tool path parameters on the interrelated criterion of mean machining error, mean standard deviation of the error and number of cutting points to decide their impact during the development of the proposed optimisation methodology,
- To evaluate the scallop height prediction capability of Eq.3.10 by comparing its computational outputs to experimental results referred to the same objective,
- To verify the automated process of extracting results in terms of chord error using programming scripts,

- To generate regression models for the optimisation criteria and consider them as temporary objective functions that will be handled by several intelligent algorithms (including the one developed in this research) for comparisons regarding the same problem.

3.6.1 Benchmark sculptured surfaces

The geometries of benchmark sculptured surfaces developed are depicted in Figure 3.1. The first benchmark surface (Fig.3.1a) is a bi-cubic Bezier patch and it was designed to test the robustness of "rolling ball" tool positioning method (Gray et al. 2003). Its control points were selected such that convex, concave and saddle regions would be created to test all possible tool orientation challenges. With reference to the control points given in the work of Gray et al. (2003) the same surface was designed in *advanced free-form surface design* environment of *CATIA® V5 R18.*

The second benchmark surface is a double surface patch contour (Fig.3.1b). The two bi-cubic surface patches are symmetrical in x-axis and they are joined together with C_0 continuity. The surface was previously used in the work of Gray et al. (2004) to demonstrate the efficiency of their proposed tool positioning methodology, known as "graphics-assisted rolling ball".

The third benchmark surface (Fig.3.1c) is a "sin-cos" benchmark sculptured surface designed with the help of a function equation presented in Equation 18. The same surface has been used in the works of Lazoglu et al. (2009) and Manav et al. (2013) as a test surface to examine force-minimal tool paths.

$$z = 3 \times \left(\cos\left(\frac{y \times \pi}{20} \right) \times \sin\left(\frac{x \times \pi}{20} \right) \right) + 3$$

Eq. 3.18

The fourth benchmark surface has been used in the work of Roman et al. (2015) and is an arbitrary geometry with varying curvature (Fig.3.1d).

All surfaces have been designed using the exact CAD information provided by the aforementioned research works. The surfaces have been designed in the *advanced free-form surface design* environment of *CATIA® V5 R18.*
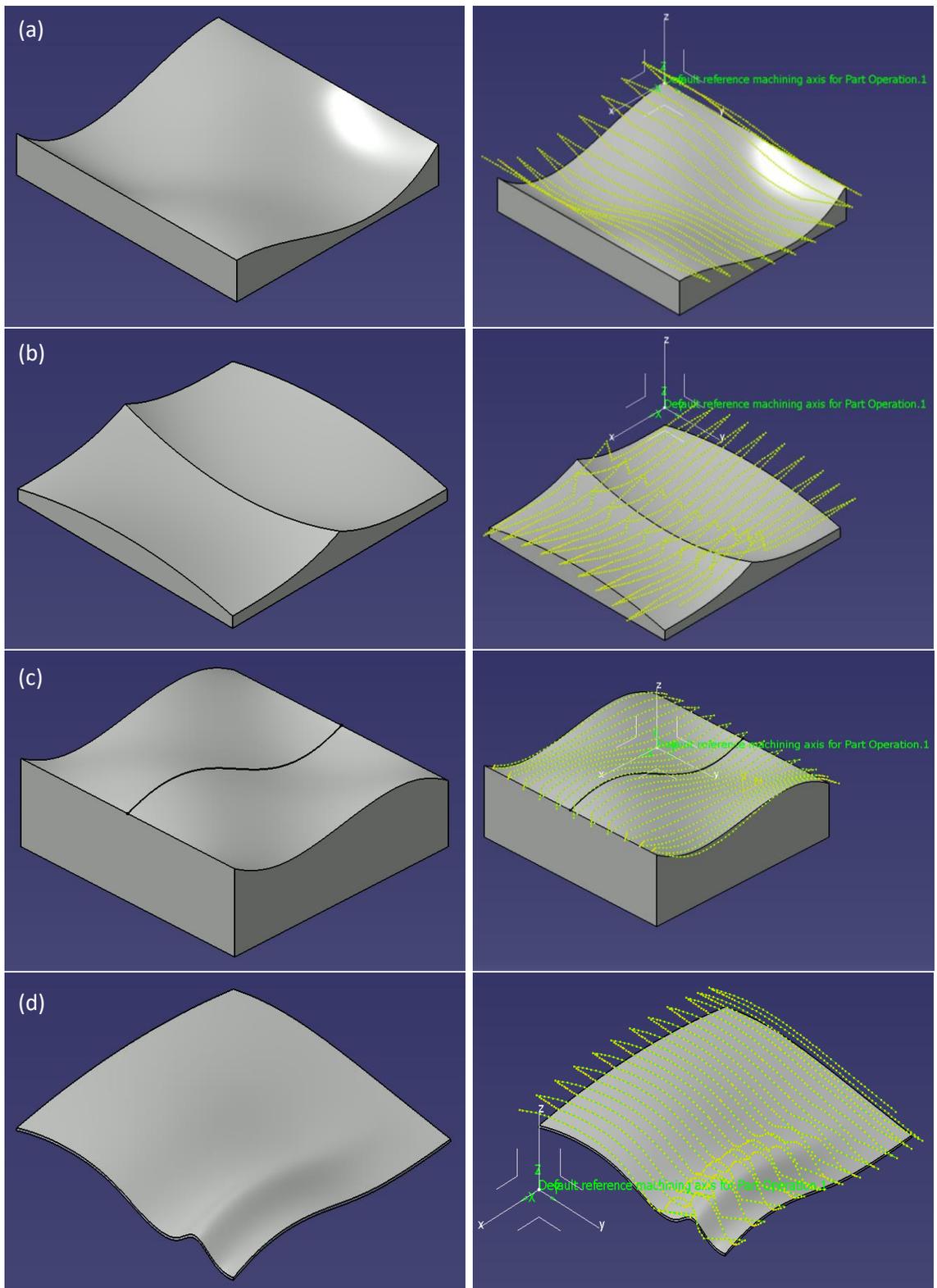
Figure 3.1: Benchmark sculptured surfaces and multi-axis sweeping tool paths: (a) SS-1, (b) SS-2, (c) SS-3, (d) SS-4.

3.6.2 Design of machining simulation experiments

The selection of a specific type to design experiments is based on the objectives and the number of independent parameters under examination. The major approaches for designing experiments are presented as follows (Montgomery, 2013):

1. Factorial designs of experiments

Factorial designs include one-factorial, fractional factorial and full factorial designs of experiments. One-factorial experimental design allows to design experiments such that only one factor at a time is investigated with reference to the objectives involved. The purpose is to identify the impact of the factor under investigation on the objective (or objectives) at different parameter levels. The factor may be either categorical or numerical. In the case of categorical factors, no predictions in terms of the objective(s) can be carried out of the vicinity of the levels tested; only the effect of the factor on the objective can be estimated. When it comes to numerical factors both effect examination and objective prediction may be achieved if adequate experimental results exist.

Fractional factorial designs examine only a fraction of all possible combinations leading thus to a reduced number of experimental runs which considered necessary, without the loss of statistically significant information. The experimenter should choose which combinations are to be excluded thus a significant knowledge is needed to identify the most important ones. In addition, some certain interactions cannot be determined owing to the excluded combinations. In general, multiple parameters can be examined simultaneously during the experiment whilst categorical and numerical factors can be involved to the problem as in the case of one-factorial designs. The aim is to identify the impact of each parameter affecting the objective(s) as well as the generation of predictive models. A limited number of interactions among parameters can also be investigated.

In full factorial designs of experiments each of the parameters involved may have different number of levels whilst they can be categorical and/or numerical. In this approach all experimental runs need to be conducted to obtain adequate statistical information to further investigate the impact of independent parameters on the objective(s). These designs obviously come with the shortcoming of increasing the cost which may significantly vary according the nature of experiments. However there is a possibility of reducing the number of experimental runs by adopting the two-level full factorial design of experiments where two levels (high and low) are considered for all factors. By reducing the number of experiments following this approach, one can benefit from the investigation of all factors as well as their interactions.

2. Taguchi's orthogonal arrays (OAs)

Taguchi's orthogonal arrays allow fractional designs, for estimating main effects with only few experimental tries without the loss of significant statistical information. These designs are applicable for investigating main effects when factors have up to two levels. Designs are also available to examine the effects for mixed-leveled experiments where the factors involved are not assigned with the same number of levels. However, the results provided by this approach may fail to indicate exactly which factor has the highest impact on the objective. Furthermore, difficulties when addressing interactions between independent parameters have been reported as an important drawback (Ross, 1996).

3. Response surface designs (RSM)

Response surface designs are often preferred when it comes to the determination of the suitable settings of parameters to attain an optimal value for the objective(s) under investigation. They try to interpolate the experimental results obtained by the experiment to locally or globally predict beneficial correlations among independent variables and objectives (responses). RSM methods are insensitive to unusual observations whilst they provide a consistent visualization of the problem's design space. However, when it comes to global approximations and many independent parameters, an RSM design may need an excessive number of function evaluations to provide reliable and qualitative results (Box and Draper, 1987).

According to the aforementioned information concerning the techniques available for designing experiments, the five surface machining tool path parameters, tool type, stepover, lead angle, tilt angle and maximum discretisation step were assigned to an L_{32} (2^5) array according the two-level full factorial experimental design approach. The selection of this approach facilitates the experimental process for obtaining the necessary results to further examine the impact of tool path parameters on the optimisation criteria which are the means of the machining error, the uniformity and the number of cutting points. The total number of experiments is deemed necessary in order to obtain reliable means. In addition, computational time is dramatically reduced anyway since the experimentation is based on automated machining simulations in CAM environment. The L_{32} (2^5) array allows also fitting a model including a mean term, five main effects, ten 2nd order interactions, ten 3rd order interactions, five 4th order interactions and a 5th order interaction (32 parameters).

The two-level full factorial experimental design approach was implemented to examine the impact of the 5-axis tool path planning parameters regarding the optimisation objectives by considering the benchmark sculptured surfaces depicted in Figure 3.1. The two-level full factorial design of

experiments was selected to reduce the number of experiments, obtain all statistically significant information and take into consideration all possible interactions up to the 3rd order. Four independent designs were established, and their experimental runs were conducted to obtain results for further statistical examination. Table 3.1 summarizes the independent experimental designs referring to the benchmark sculptured surfaces tested.

Table 3.1: Two-level full factorial experimental designs with reference to the benchmark sculptured surfaces examined.

Benchmark surface	Levels	Tool	Stepover (%D)	Lead angle (deg)	Tilt angle (deg)	MaxDstep (mm)
SS-1	Low	D37.4-Rc0	10	20	0	0.7
	High	D37.4-Rc6	45	35	7	1.397
SS-2	Low	D50.8-Rc6.35	10	15	0	0.762
	High	D50.8-Rc0	45	20	15	2
SS-3	Low	D12-Rc0	17	15	0	1
	High	D12-Rc3	45	20	15	5
SS-4	Low	D20-Rc0	10	30	0	0.5
	High	D20-Rc4	45	40	5	2.5

A number of steps were followed to conduct the experiments and obtain the necessary results for statistical analysis and interpretation. The steps are presented and explained in the following sub-sections.

1. Automatic tool path computation according to the inputs for tool path parameters corresponding to the "multi-axis sweeping" strategy.

Experimental values of parameters for the tool path were imported to the sculptured surface machining strategy and the outputs for performance metrics were extracted using the automated part of the proposed methodology through a function developed in Visual Basic®. Computations deal with all necessary attributes to obtain the average values of chord error, scallop height, standard deviations and the number of cutting points. The function that automates the entire manufacturing environment has been developed in *Microsoft Visual Basic*®. At that point the automation function was deployed to automatically extract computational results for the aforementioned objectives whilst its main goal is to interact with the intelligent algorithm developed in this research towards the ultimate purpose of heuristic tool path optimisation for sculptured surface machining. The automation function was executed as many times as the number of experimental runs for each of the four individual designs for the benchmark sculptured surfaces, hence, 128 times.

2. Machining simulation and storage of 3D CAM outputs in *.stl file format.

The computational outputs for all tool paths were stored and simulated using material removal functions embedded to CAM software. Corresponding machined models were stored as 3D CAM

outputs to examine their machining quality. Machining quality of 3D CAM outputs was investigated by means of virtual surface touch probing techniques available to the commercially available 3D metrology software *Geomagic® Qualify Probe® 2013*.

3. Real-time deviation measurements with respect to ideal surfaces, performed on scallop volumes of 3D CAM outputs in their *.stl version using 3D metrology utilities of *Geomagic® Qualify Probe® 2013*.

Experimental 3D CAM outputs were imported to *Geomagic® Qualify Probe® 2013* 3D metrology package as *.stl entities and compared to the original CAD sculptured surface models. The models were aligned regarding the same machining axis system as that determined in the process documents for the machining simulations. Virtual probing was conducted to all scallop curves in 3D CAM outputs whilst 500 to 1000 measurements were taken depending on the benchmark surface's nominal dimensions and profound scallop volumes (excess material). Real-time deviation measurements for machining error were then exported in *.txt format in order to compare them to those obtained by analytical computations.

3.6.3 Experimental results and descriptive statistics

The results of the experiments performed on all surfaces and the domains of their variables, as determined by the L_{32} (2^5) array, are shown in Figure 3.2 and summarized in Tables 3.2, 3.3, 3.4 and 3.5 corresponding to sculptured surfaces SS-1, SS-2, SS-3 and SS-4 respectively. The optimisation objective under interest is the Pareto criterion. The magnitudes of individual objectives were mapped to [0, 1] range using a simple normalization technique. According to this technique the results for each objective are divided to their largest observation-result, thus, the inherent bias in favor of larger magnitudes and different units is removed and comparison is achieved under fair means. Thereby the Pareto criterion for each experimental design has been computed using Eq.3.16.

With reference to the experimental designs summarized in Table 3.1, the experimental runs that minimise and maximise the Pareto criterion were investigated. For the first benchmark sculptured surface (SS-1) the 26th experiment minimised the Pareto criterion. The parameters corresponding to this run are shown in Table 3.2 along with the rest experimental runs. For the same surface the 5th experiment maximised the Pareto criterion whilst its corresponding values for tool path parameters are also summarized in Table 3.2. It can be seen that cutting tool and stepover parameters do not give a clear indication for the trend of Pareto criterion since both parameters minimise and maximise

its magnitude for the same values (Flat end-mill, stepover 10%). For the second benchmark sculptured surface (SS-2) minimum and maximum Pareto results were observed in 17th and 32nd experiments respectively. The results are also summarized in Table 3.3. In this case only MaxDstep parameter does not provide evidence for the trend of Pareto criterion since the same value is indicated to both experimental runs, 17th and 32nd. As far as the third benchmark sculptured surface (SS-3) is concerned, the 4th and the 20th experiment minimised and maximised respectively the result of Pareto criterion whilst MaxDstep parameter was the only parameter to exhibit a clear indication about the trend of Pareto result (Table 3.4). The rest of parameters minimised and maximised Pareto result under the same experimental values. In the case of the fourth benchmark sculptured surface (SS-4) minimum and maximum indications for the Pareto criterion were observed to the 22nd and the 1st experiment respectively. The results are also summarized in Table 3.5. The tool path parameters that gave a clear effect on Pareto criterion were cutting tool, lead angle and MaxDstep whereas stepover and tilt angle did not show evidence in terms of their effects on Pareto criterion.
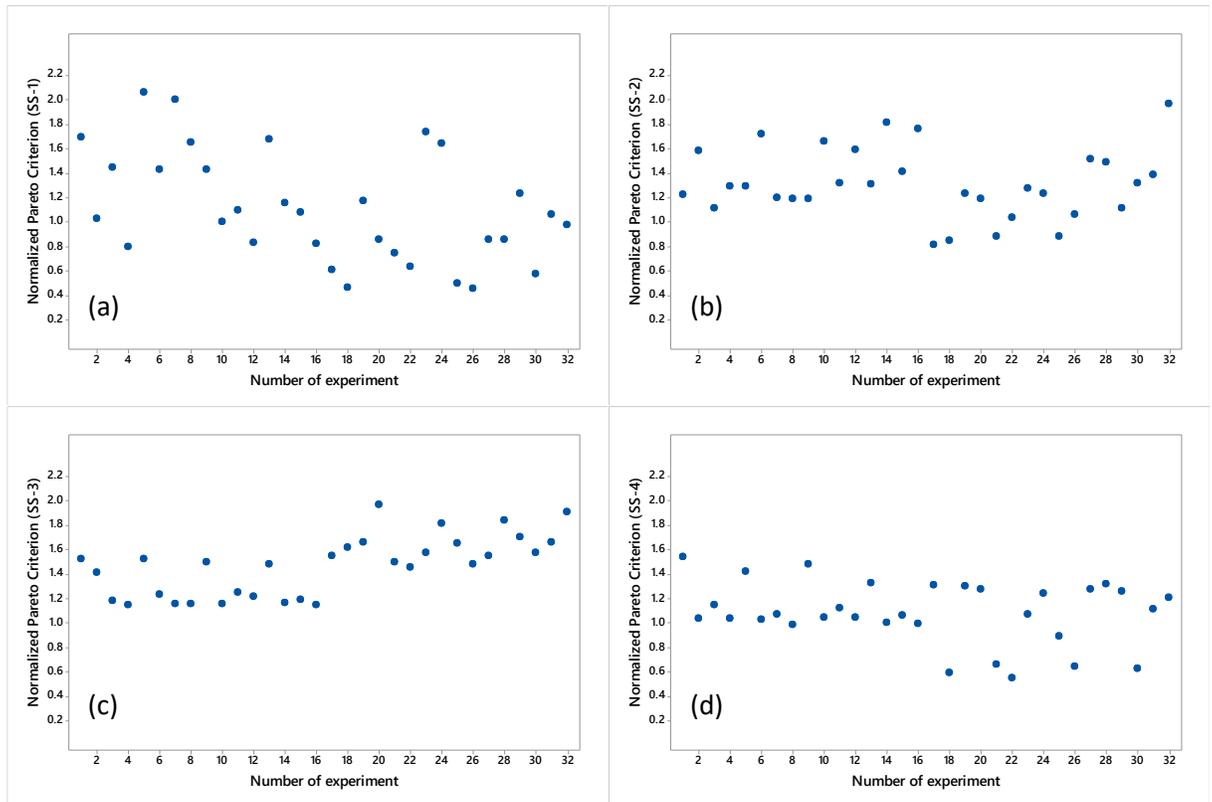
Figure 3.2: Benchmark sculptured surfaces and multi-axis sweeping tool paths: (a) SS-1, (b) SS-2, (c) SS-3, (d) SS-4.

Table 3.2: Two-level full factorial experimental results corresponding to the first benchmark sculptured surface (SS-1).

a/a exp	Tool	a_e (%Ø)	a_L (°)	a_T (°)	MaxDstep (mm)	$\overline{h}$	$\overline{\delta}$	$\overline{stdevh}$	$\overline{stdev\delta}$	NoCLs	OF_{PO}
1	D37.4Rc0	10	20	0	0.7	0.2692	0.1474	0.4022	0.9165	0.9829	1.6967
2	D37.4Rc6	10	20	0	0.7	0.1718	0.0526	0.0281	0.2338	0.9640	1.0238
3	D37.4Rc0	45	20	0	0.7	0.8700	0.4842	0.0907	0.3523	0.2255	1.4426
4	D37.4Rc6	45	20	0	0.7	0.6864	0.0632	0.0028	0.1401	0.2173	0.7934
5	D37.4Rc0	10	35	0	0.7	0.2873	0.3053	0.7474	1.0000	0.9163	2.0602
6	D37.4Rc6	10	35	0	0.7	0.1757	0.2316	0.6886	0.2488	1.0000	1.4299
7	D37.4Rc0	45	35	0	0.7	1.0000	0.9105	0.1559	0.4054	0.2226	2.0037
8	D37.4Rc6	45	35	0	0.7	0.8276	0.7684	0.1494	0.1971	0.2374	1.6503
9	D37.4Rc0	10	20	7	0.7	0.2757	0.0526	0.0236	0.9969	0.9492	1.4319
10	D37.4Rc6	10	20	7	0.7	0.1727	0.0474	0.0007	0.2345	0.9503	1.0034
11	D37.4Rc0	45	20	7	0.7	0.9414	0.0526	0.0001	0.4057	0.2394	1.1000
12	D37.4Rc6	45	20	7	0.7	0.7078	0.0737	0.0060	0.1330	0.2399	0.8292
13	D37.4Rc0	10	35	7	0.7	0.2506	0.3684	0.4435	0.8308	0.8990	1.6779
14	D37.4Rc6	10	35	7	0.7	0.1736	0.1895	0.2505	0.2422	0.9812	1.1565
15	D37.4Rc0	45	35	7	0.7	0.8199	0.1895	0.0079	0.2970	0.2190	1.0769
16	D37.4Rc6	45	35	7	0.7	0.6891	0.0842	0.0073	0.1336	0.2310	0.8192
17	D37.4Rc0	10	20	0	1.397	0.2384	0.2263	0.0560	0.1067	0.3588	0.6092
18	D37.4Rc6	10	20	0	1.397	0.1681	0.1368	0.0084	0.0272	0.3515	0.4666
19	D37.4Rc0	45	20	0	1.397	0.6125	0.5579	0.0118	0.0218	0.0823	1.1738
20	D37.4Rc6	45	20	0	1.397	0.6814	0.1684	0.0010	0.0184	0.0797	0.8538
21	D37.4Rc0	10	35	0	1.397	0.2375	0.4000	0.0980	0.0995	0.3354	0.7469
22	D37.4Rc6	10	35	0	1.397	0.1728	0.3263	0.0993	0.0332	0.3651	0.6325
23	D37.4Rc0	45	35	0	1.397	0.7318	1.0000	0.0207	0.0288	0.0814	1.7344
24	D37.4Rc6	45	35	0	1.397	0.7746	0.8684	0.0201	0.0211	0.0868	1.6458
25	D37.4Rc0	10	20	7	1.397	0.2238	0.1158	0.0072	0.0877	0.3481	0.4955
26	**D37.4Rc6**	**10**	**20**	**7**	**1.397**	**0.1661**	**0.1263**	**0.0001**	**0.0272**	**0.3465**	**0.4542**
27	D37.4Rc0	45	20	7	1.397	0.7168	0.1368	0.0000	0.0343	0.0874	0.8587
28	D37.4Rc6	45	20	7	1.397	0.6900	0.1632	0.0011	0.0163	0.0880	0.8579
29	D37.4Rc0	10	35	7	1.397	0.2053	0.4368	1.0000	0.0000	0.3293	1.2332
30	D37.4Rc6	10	35	7	1.397	0.1690	0.2789	0.0377	0.0291	0.3608	0.5790
31	D37.4Rc0	45	35	7	1.397	0.6266	0.4316	0.0087	0.0221	0.0800	1.0616
32	D37.4Rc6	45	35	7	1.397	0.6714	0.3000	0.0061	0.0170	0.0853	0.9754

Table 3.3: Two-level full factorial experimental results corresponding to the second benchmark sculptured surface (SS-2).

a/a exp	Tool	a_e (%Ø)	a_L (°)	a_T (°)	MaxDstep (mm)	$\overline{h}$	$\overline{\delta}$	$\overline{stdevh}$	$\overline{stdev\delta}$	NoCLs	OF_{PO}
1	D50.8Rc6.35	10	15	0	0.762	0.2058	0.3252	0.0549	0.4066	1.0000	1.2227
2	D50.8Rc0	10	15	0	0.762	0.2774	0.3497	1.0000	0.0631	0.9855	1.5794
3	D50.8Rc6.35	45	15	0	0.762	0.8830	0.1840	0.0009	0.2114	0.2230	1.1106
4	D50.8Rc0	45	15	0	0.762	0.9984	0.2209	0.0012	0.3655	0.2220	1.2924
5	D50.8Rc6.35	10	20	0	0.762	0.2068	0.4724	0.0928	0.3774	0.9971	1.2948
6	D50.8Rc0	10	20	0	0.762	0.2684	0.5706	0.1178	1.0000	0.9968	1.7167
7	D50.8Rc6.35	45	20	0	0.762	0.8925	0.2699	0.0018	0.1935	0.2229	1.1997
8	D50.8Rc0	45	20	0	0.762	0.8528	0.2945	0.0021	0.2339	0.2244	1.1926
9	D50.8Rc6.35	10	15	15	0.762	0.2117	0.5092	0.0757	0.3482	0.8522	1.1940
10	D50.8Rc0	10	15	15	0.762	0.2944	0.6810	0.1003	0.9452	0.8350	1.6558
11	D50.8Rc6.35	45	15	15	0.762	0.8831	0.4049	0.0037	0.1751	0.2226	1.3193
12	D50.8Rc0	45	15	15	0.762	0.9928	0.5521	0.0052	0.3137	0.2218	1.5930
13	D50.8Rc6.35	10	20	15	0.762	0.2099	0.6687	0.1085	0.3124	0.8760	1.3101
14	D50.8Rc0	10	20	15	0.762	0.2872	0.8773	0.1416	0.9544	0.8577	1.8147
15	D50.8Rc6.35	45	20	15	0.762	0.8615	0.5215	0.0050	0.1557	0.2224	1.4099
16	D50.8Rc0	45	20	15	0.762	1.0000	0.7117	0.0072	0.3381	0.2220	1.7602

17	D50.8Rc6.35	10	15	0	2.000	0.1948	0.5215	0.0093	0.0463	0.3788	0.8122
18	D50.8Rc0	10	15	0	2.000	0.2305	0.5215	0.0094	0.1057	0.3751	0.8481
19	D50.8Rc6.35	45	15	0	2.000	0.8712	0.3620	0.0002	0.0298	0.0849	1.2365
20	D50.8Rc0	45	15	0	2.000	0.8004	0.3865	0.0002	0.0332	0.0844	1.1904
21	D50.8Rc6.35	10	20	0	2.000	0.2072	0.5890	0.0115	0.0600	0.3752	0.8831
22	D50.8Rc0	10	20	0	2.000	0.2583	0.6994	0.0159	0.1345	0.3762	1.0398
23	D50.8Rc6.35	45	20	0	2.000	0.8843	0.3865	0.0002	0.0261	0.0844	1.2739
24	D50.8Rc0	45	20	0	2.000	0.7790	0.4479	0.0003	0.0312	0.0849	1.2302
25	D50.8Rc6.35	10	15	15	2.000	0.2062	0.6074	0.0094	0.0487	0.3263	0.8785
26	D50.8Rc0	10	15	15	2.000	0.2589	0.7485	0.0124	0.1186	0.3184	1.0646
27	D50.8Rc6.35	45	15	15	2.000	0.8535	0.6564	0.0007	0.0244	0.0847	1.5126
28	D50.8Rc0	45	15	15	2.000	0.8384	0.6442	0.0007	0.0344	0.0847	1.4854
29	D50.8Rc6.35	10	20	15	2.000	0.2089	0.8528	0.0164	0.0451	0.3338	1.1146
30	D50.8Rc0	10	20	15	2.000	0.2685	1.0000	0.0191	0.1221	0.3258	1.3173
31	D50.8Rc6.35	45	20	15	2.000	0.8160	0.5706	0.0005	0.0213	0.0845	1.3893
32	D50.8Rc0	45	20	15	2.000	0.9660	1.0000	0.0012	0.0444	0.0833	1.9683

Table 3.4: Two-level full factorial experimental results corresponding to the third benchmark sculptured surface (SS-3).

a/a exp	Tool	a_e (%Ø)	a_L (∘)	a_T (∘)	MaxDstep (mm)	$\bar{h}$	$\bar{\delta}$	$\overline{stdevh}$	$\overline{stdev\delta}$	NoCLs	OF_{PO}
1	D12Rc0	17	15	0	1	0.4253	0.0954	0.1596	0.8776	0.9835	1.5212
2	D12Rc3	17	15	0	1	0.3948	0.0983	0.4684	0.3941	1.0000	1.4096
3	D12Rc0	45	15	0	1	0.9087	0.1012	0.0572	0.4202	0.3791	1.1796
4	**D12Rc3**	**45**	**15**	**0**	**1**	**0.9595**	**0.0954**	**0.0025**	**0.2266**	**0.3851**	**1.1461**
5	D12Rc0	17	20	0	1	0.4508	0.0954	0.0245	1.0000	0.9838	1.5218
6	D12Rc3	17	20	0	1	0.3839	0.0954	0.2129	0.3272	0.9960	1.2302
7	D12Rc0	45	20	0	1	0.9037	0.1012	0.0574	0.3680	0.3768	1.1544
8	D12Rc3	45	20	0	1	0.9716	0.0954	0.0025	0.2411	0.3839	1.1598
9	D12Rc0	17	15	15	1	0.4498	0.0809	0.0075	0.9921	0.9772	1.4952
10	D12Rc3	17	15	15	1	0.3818	0.0925	0.0205	0.3172	0.9988	1.1561
11	D12Rc0	45	15	15	1	0.9436	0.1040	0.0769	0.4820	0.3856	1.2485
12	D12Rc3	45	15	15	1	1.0000	0.1040	0.0666	0.2768	0.3869	1.2193
13	D12Rc0	17	20	15	1	0.4401	0.0954	0.0301	0.9435	0.9815	1.4826
14	D12Rc3	17	20	15	1	0.3886	0.0954	0.0254	0.3409	0.9982	1.1682
15	D12Rc0	45	20	15	1	0.9170	0.1040	0.0759	0.4082	0.3839	1.1935
16	D12Rc3	45	20	15	1	0.9623	0.0983	0.0091	0.2109	0.3889	1.1508
17	D12Rc0	17	15	0	5	0.3439	0.8931	0.8911	0.0224	0.1901	1.5494
18	D12Rc3	17	15	0	5	0.3700	0.9162	0.9505	0.0143	0.1898	1.6190
19	D12Rc0	45	15	0	5	0.7666	0.8815	0.1269	0.0135	0.0737	1.6558
20	D12Rc3	45	15	0	5	0.9573	1.0000	0.1568	0.0110	0.0727	1.9658
21	D12Rc0	17	20	0	5	0.3415	0.8584	0.8457	0.0275	0.1931	1.4965
22	D12Rc3	17	20	0	5	0.3594	0.8382	0.7915	0.0118	0.1911	1.4547
23	D12Rc0	45	20	0	5	0.7660	0.8035	0.1051	0.0169	0.0737	1.5759
24	D12Rc3	45	20	0	5	0.8891	0.9162	0.1388	0.0088	0.0739	1.8128
25	D12Rc0	17	15	15	5	0.4038	0.9017	0.9546	0.0381	0.1930	1.6513
26	D12Rc3	17	15	15	5	0.3937	0.8295	0.7886	0.0167	0.1945	1.4773
27	D12Rc0	45	15	15	5	0.7810	0.7630	0.0998	0.0139	0.0764	1.5501
28	D12Rc3	45	15	15	5	0.9514	0.8786	0.1286	0.0110	0.0751	1.8368
29	D12Rc0	17	20	15	5	0.4150	0.9133	1.0000	0.0409	0.1965	1.6990
30	D12Rc3	17	20	15	5	0.4035	0.8699	0.8842	0.0169	0.1956	1.5722
31	D12Rc0	45	20	15	5	0.8355	0.8208	0.1221	0.0175	0.0772	1.6640
32	D12Rc3	45	20	15	5	0.9753	0.9249	0.1419	0.0121	0.0759	1.9079

Table 3.5: Two-level full factorial experimental results corresponding to the fourth benchmark sculptured surface (SS-4).

a/a exp	Tool	a_e (%Ø)	a_L (°)	a_T (°)	MaxDstep (mm)	$\bar{h}$	$\bar{\delta}$	$\overline{stdevh}$	$\overline{stdev\delta}$	NoCLs	OF_{PO}
1	D20Rc0	10	30	0	0.5	0.3182	0.0710	0.1336	0.9713	1.0000	1.5402
2	D20Rc4	10	30	0	0.5	0.2275	0.0625	0.0230	0.2601	0.9557	1.0381
3	D20Rc0	45	30	0	0.5	1.0000	0.0682	0.0016	0.3568	0.2178	1.1475
4	D20Rc4	45	30	0	0.5	0.9381	0.0625	0.0011	0.1574	0.2092	1.0344
5	D20Rc0	10	40	0	0.5	0.3109	0.2131	0.0087	0.9124	0.9455	1.4202
6	D20Rc4	10	40	0	0.5	0.2196	0.0625	0.0236	0.2438	0.9542	1.0303
7	D20Rc0	45	40	0	0.5	0.9444	0.0625	0.0012	0.2844	0.2096	1.0674
8	D20Rc4	45	40	0	0.5	0.8912	0.0625	0.0011	0.1474	0.2095	0.9877
9	D20Rc0	10	30	5	0.5	0.3191	0.0653	0.0283	1.0000	0.9881	1.4770
10	D20Rc4	10	30	5	0.5	0.2299	0.0625	0.0288	0.2746	0.9538	1.0427
11	D20Rc0	45	30	5	0.5	0.9738	0.0710	0.0197	0.3310	0.2300	1.1259
12	D20Rc4	45	30	5	0.5	0.9389	0.0682	0.0179	0.1587	0.2205	1.0460
13	D20Rc0	10	40	5	0.5	0.2904	0.2330	0.0055	0.7743	0.9430	1.3309
14	D20Rc4	10	40	5	0.5	0.2072	0.0625	0.0237	0.1868	0.9445	1.0045
15	D20Rc0	45	40	5	0.5	0.9380	0.0625	0.0012	0.2822	0.2097	1.0608
16	D20Rc4	45	40	5	0.5	0.8988	0.0625	0.0025	0.1514	0.2121	0.9964
17	D20Rc0	10	30	0	2.5	0.2654	0.5085	1.0000	0.0356	0.2185	1.3111
18	D20Rc4	10	30	0	2.5	0.2237	0.3068	0.1299	0.0124	0.2102	0.5882
19	D20Rc0	45	30	0	2.5	0.8574	0.4460	0.0350	0.0152	0.0473	1.3053
20	D20Rc4	45	30	0	2.5	0.9523	0.3239	0.0107	0.0077	0.0455	1.2771
21	D20Rc0	10	40	0	2.5	0.2310	0.3239	0.2510	0.0281	0.2131	0.6566
22	**D20Rc4**	**10**	**40**	**0**	**2.5**	**0.2167**	**0.2784**	**0.0565**	**0.0128**	**0.2186**	**0.5457**
23	D20Rc0	45	40	0	2.5	0.9444	0.0625	0.0012	0.2844	0.2096	1.0674
24	D20Rc4	45	40	0	2.5	0.9315	0.3097	0.0083	0.0105	0.0470	1.2422
25	D20Rc0	10	30	5	2.5	0.2694	0.4006	0.5112	0.0369	0.2180	0.8926
26	D20Rc4	10	30	5	2.5	0.2258	0.3267	0.2388	0.0128	0.2095	0.6422
27	D20Rc0	45	30	5	2.5	0.8477	0.4290	0.0313	0.0130	0.0504	1.2784
28	D20Rc4	45	30	5	2.5	0.9402	0.3807	0.0250	0.0072	0.0478	1.3221
29	D20Rc0	10	40	5	2.5	0.2383	1.0000	0.0823	0.0296	0.2137	1.2616
30	D20Rc4	10	40	5	2.5	0.2149	0.3125	0.2522	0.0120	0.2162	0.6282
31	D20Rc0	45	40	5	2.5	0.8134	0.3011	0.0066	0.0101	0.0470	1.1157
32	D20Rc4	45	40	5	2.5	0.8900	0.3182	0.0139	0.0074	0.0474	1.2093

Obviously, the observations derived from the experimental runs do not indicate clearly the effects of tool path parameters on the Pareto criterion. It can be also deduced that the trend of Pareto criterion does not follow the same trend when examining different sculptured surfaces under different tool path parameter values with reference to their applicable ranges. Further analysis has been conducted with reference to experimental results by examining the main effects of the normalized individual optimisation objectives and Pareto criterion. Main effects plots have been generated to investigate the differences among level means regarding the tool path parameters. The effect of each tool path parameter is illustrated with a straight line passing across the reference line (dashed line) that depicts the overall mean. In the case of obtaining a horizontal effect line (parallel to x-axis) no indication for the main effect will exist. This means that each parameter level will affect the objective under study in the same manner whilst the objective's mean will be maintained across the two levels of that

parameter. For a line to exhibit the corresponding parameter's main effect, both a steep slope and a large length should be noticeable.

Main effects of tool path parameters on the objective of machining error (as a combined effect of scallop height and chordal deviation) were investigated by generating corresponding main effects plots. The group of main effects plots for all benchmark sculptured surfaces regarding machining error objective is illustrated in Figure 3.3. A first observation suggests that significant differences in terms of main effects are indicated when dealing with a variety of sculptured surfaces and variable tool path parameter levels. In addition, a dominant effect by stepover parameter on machining error objective is also profound in all cases. In the case of the first benchmark sculptured surface (SS-1) the mean of machining error is reduced for a fillet end-mill, low stepover distance, low lead angle, high tilt angle and low MaxDstep. The largest main effect is indicated by stepover parameter, followed by the main effects of lead angle, tilt angle, cutting tool and MaxDstep. In the case of the second benchmark sculptured surface (SS-2) the mean of machining error is reduced for a fillet end-mill, low stepover distance, low lead angle, low tilt angle and low MaxDstep parameter values. Yet again the largest main effect is observed for stepover parameter, followed by the main effects of tilt angle, cutting tool, lead angle and MaxDstep. By comparing main effects of parameters on machining error for SS-1 and SS-2 it can be seen that main effects reduce the mean under the same levels respectively (except from tilt angle) but they differ in impact order. In the case of the third benchmark sculptured surface (SS-3) the mean of machining error is reduced for flat end-mill, low stepover distance, high lead angle, low tilt angle and low MaxDstep values. The largest main effect is indicated by MaxDstep, followed by stepover distance, cutting tool, tilt angle and lead angle. This is an entirely different main effect order compared to the case of SS-1 and SS-2. In the last case of the fourth benchmark sculptured surface (SS-4) results suggest that mean is reduced for fillet end-mill, low stepover, high lead angle, low tilt angle and low MaxDstep parameter values. Stepover parameter holds the largest main effect whilst the main effects of MaxDstep, cutting tool, tilt angle and lead angle follow next.
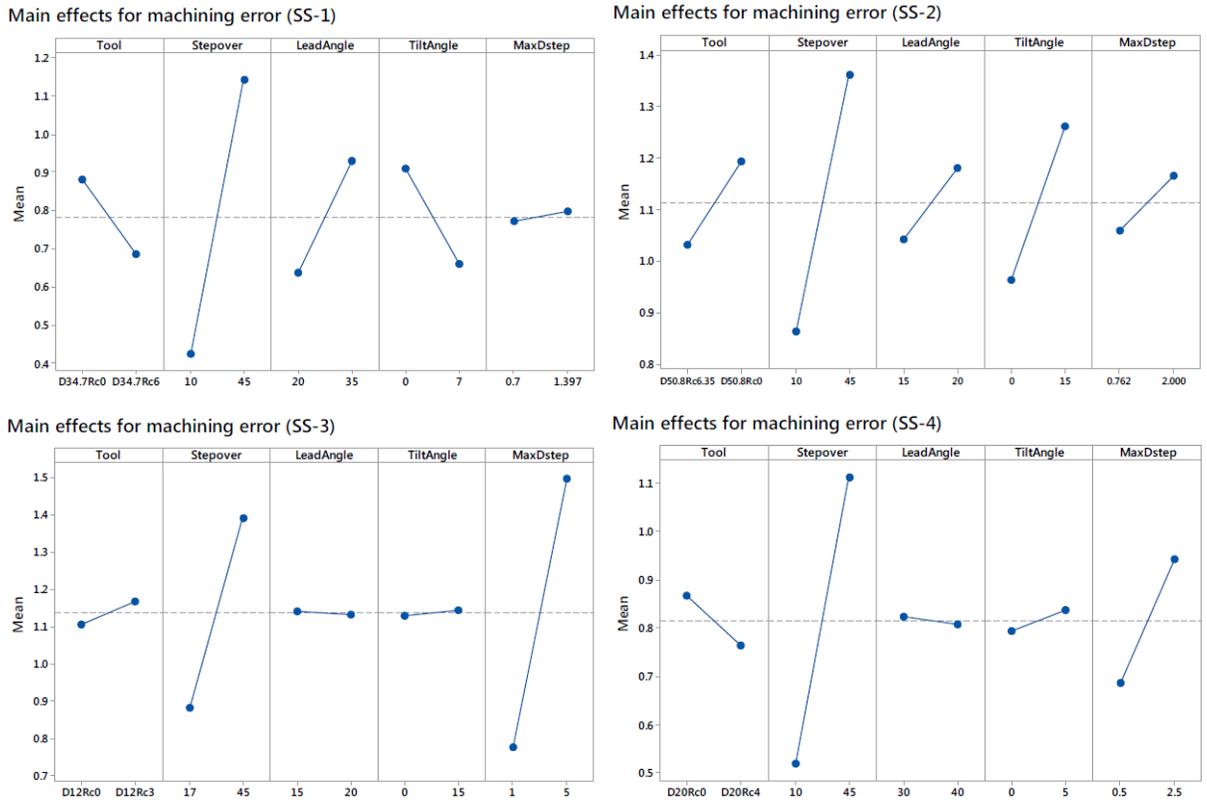
Figure 3.3: Main effects of linear terms on machining error objective, per benchmark sculptured surface experiment, SS-1, SS-2, SS-3 and SS-4.

The same plots for main effects were generated to investigate the impact of tool path parameters on the objective of machining error distribution. The main effects plots are depicted in Figure 3.4. For the case of the first benchmark sculptured surface (SS-1) machining error distribution is greatly affected by MaxDstep parameter. The main effect of MaxDstep is followed by the profound effects of stepover distance, cutting tool, lead angle and tilt angle parameters. The mean is reduced using fillet end-mill, large stepover distance, low lead angle, high tilt angle and high MaxDstep. In the case of the second surface (SS-2) MaxDstep holds the most dominant main effect machining error distribution whilst the main effects of stepover distance, cutting tool, tilt angle and lead angle parameters follow it. The main effect of lead angle is hardly observable since the mean is the same for both low and high lead angle parameter levels. The mean is reduced for fillet end-mill, large stepover distance, high tilt angle and high MaxDstep levels. In the case of the third surface (SS-3) stepover dominates against the rest of parameters in terms of its main effect. The main effect of cutting tool follows next as well as MaxDstep, lead angle and tilt angle. The mean is reduced when using flat end-mill, large stepover, high lead angle, high tilt angle and large MaxDstep. In the case of the fourth surface (SS-4) stepover holds the strongest effect whilst cutting tool exhibits also a significant impact. The effects of these

two parameters are followed by those of MaxDstep, lead angle and tilt angle. The mean is reduced using fillet end-mill, large stepover, high lead angle, high tilt angle and large MaxDstep.
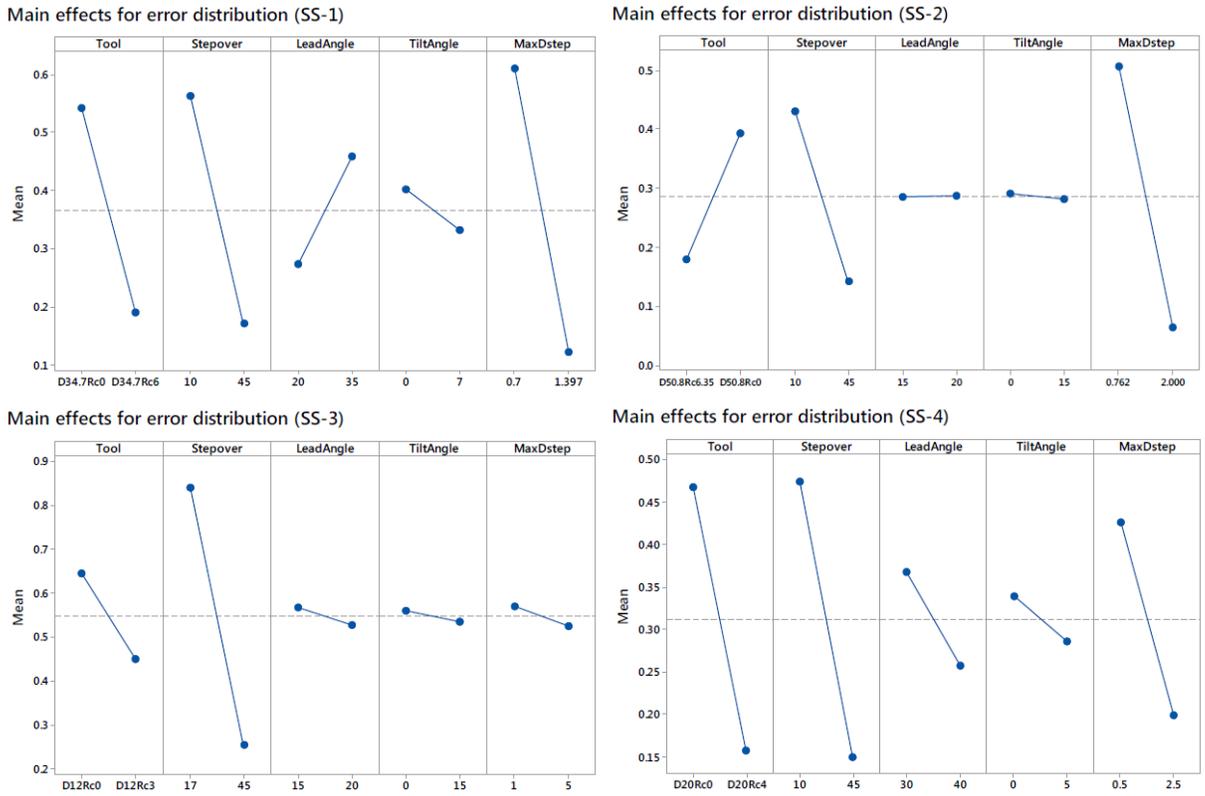
Figure 3.4: Main effects of linear terms on machining error distribution objective, per benchmark sculptured surface experiment, SS-1, SS-2, SS-3 and SS-4.

By comparing the main effects of machining error and those referring to its distribution, an important observation suggests that, while the error is benefited by low stepover distances and low discretisation steps (as it is expected), its distribution is maintained under large values for these tool path parameters. Therefore, an important trade-off is found between the machining error and its distribution.

The main effects of tool path parameters on the objective of the number of cutting points (CL points) were examined and the resulting plots are depicted in Figure 3.5. A significant observation for this objective is that the order of tool path parameters' main effects is more profound compared to those reported for the objectives of machining error and machining error distribution, at least when it comes to stepover and MaxDstep. For the first surface (SS-1) the number of CL points is mainly affected by stepover and MaxDstep followed by the main effects of cutting tool, tilt angle and lead angle. Lead and tilt angle effects do not seem to have a significant effect as regards their parameter levels. The mean is reduced using flat end-mill, large stepover, high lead angle, high tilt angle and

64

large MaxDstep. For the second sculptured surface (SS-2) stepover and MaxDstep suggest the strongest effect on CL points as in the case of SS-1. Their main effects are followed by those of tilt angle, lead angle and cutting tool. The effects of cutting tool and lead angle seem to be of minor importance.
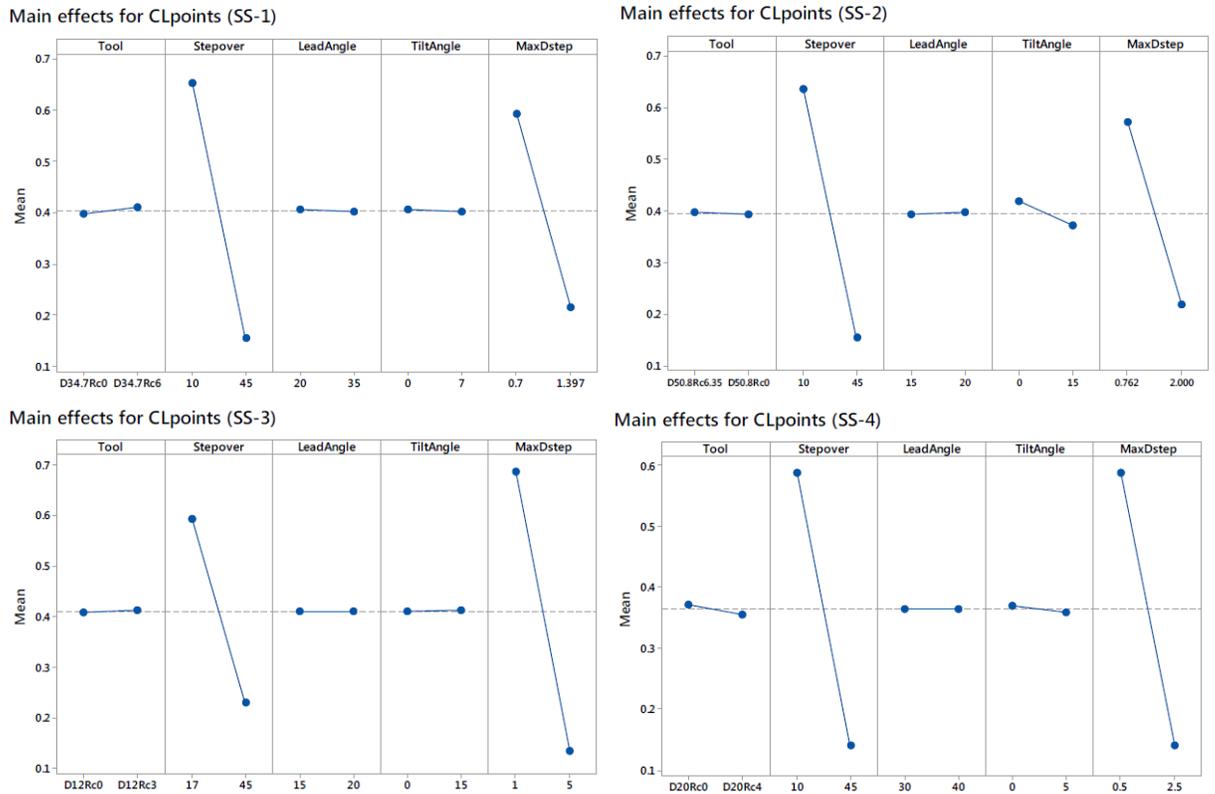
Figure 3.5: Main effects of linear terms on number of CL points objective, per benchmark sculptured surface experiment, SS-1, SS-2, SS-3 and SS-4.

The mean is reduced when using large stepover distance and large MaxDstep (as expected) as well as high tilt angle, whilst it is slightly reduced when using flat end-mill and low lead angle. In the case of the third surface (SS-3) MaxDstep holds a dominant effect followed by the effects of stepover and cutting tool. The main effects of lead and tilt angles do not seem to be significant. The mean is slightly reduced using flat end-mill whilst no change is observed referring to the effects of lead and tilt angles. As regards the case of the last sculptured surface (SS-4), main effects of stepover and MaxDstep dominate the same, followed by the effects of cutting tool and tilt angle. The main effect of lead angle is deemed as insignificant. The mean is reduced when using fillet end-mill, high tilt angle, as well as large stepover and large MaxDstep as expected.

By considering the results for main effects reported for the individual objectives of machining error, machining error distribution and number of CL points, the main effects on the Pareto criterion -which

is the criterion under interest- were examined. The resulting plots for main effects of tool path parameters on the Pareto criterion are shown in Figure 3.6.
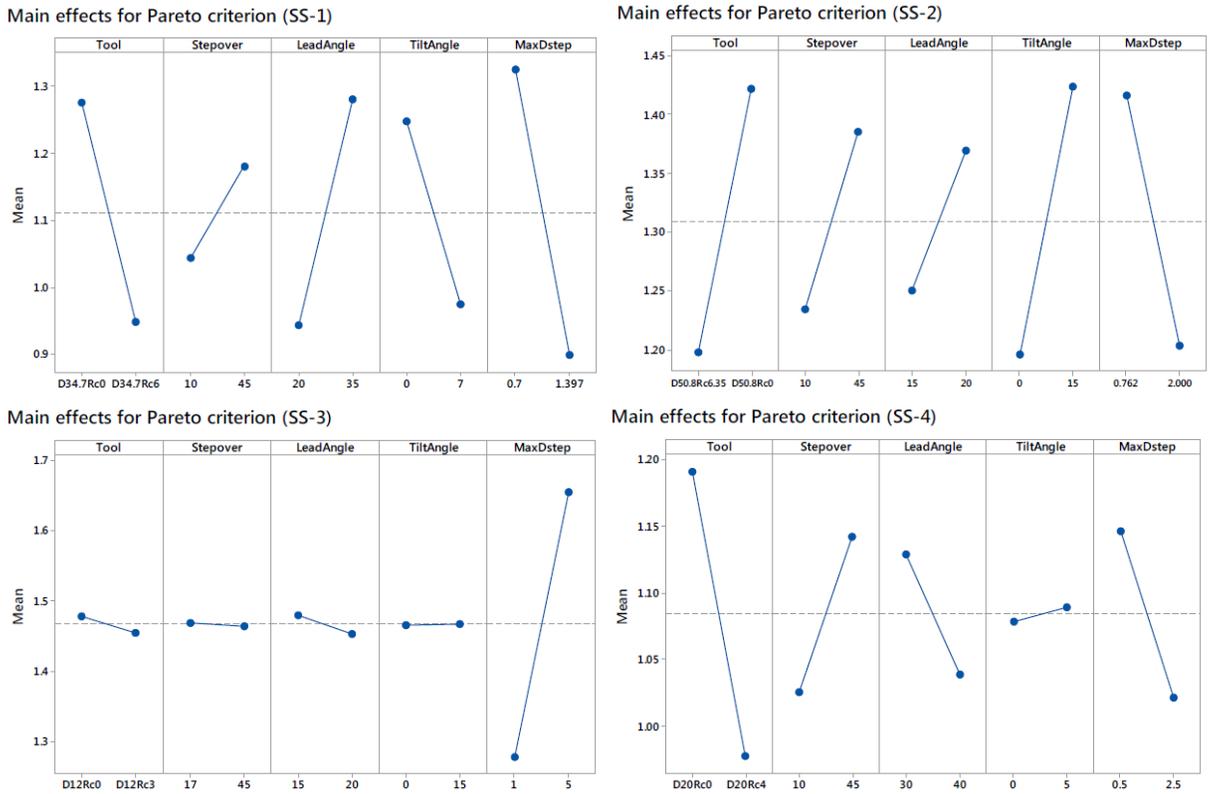
Figure 3.6: Main effects of linear terms on Pareto criterion, per benchmark sculptured surface experiment, SS-1, SS-2, SS-3 and SS-4.

For the first sculptured surface (SS-1) MaxDstep seems to hold the most significant effect on Pareto criterion. MaxDstep's effect is followed by the main effects of cutting tool, lead angle, tilt angle and stepover parameters. The mean is reduced using fillet end-mill, low stepover, low lead angle, high tilt angle and high MaxDstep. In the case of the second sculptured surface (SS-2) cutting tool and tilt angle exhibit the most significant effects on Pareto criterion. MaxDstep follows next with significant effect as well, followed by the effects of stepover and lead angle. Mean is reduced using fillet end-mill, low stepover, low lead angle, low tilt angle and high MaxDstep. As regards the third sculptured surface (SS-3), MaxDstep has the strongest effect on Pareto criterion. The main effects of lead angle, cutting tool, stepover and tilt angle follow next. The mean is reduced using fillet end-mill, large stepover, high lead angle, low tilt angle (with insignificant effect) and low MaxDstep values. In the case of the last sculptured surface (SS-4) cutting tool has the most significant effect on Pareto criterion. Its dominant effect is followed by the effects of MaxDstep, stepover, lead angle and tilt

angle whilst the mean is reduced using fillet end-mill, low stepover, high lead angle, low tilt angle and large MaxDstep parameter values.

It is obvious that main effects of tool path parameters referring to the individual criteria do not maintain the same trend and significance when compared to the ones corresponding to the Pareto criterion even though the latter is derived from the individual criteria. To draw the conclusion concerning the formulation of tool path chromosomes in terms of the representation accuracy of parameters, interactions among them were also examined. To establish a solidified assumption about the effects of tool path parameters, Pareto charts and normal plots of the standardized effects have been generated directly for investigating all effects and possible interactions up to the 3rd order for Pareto criterion, for all benchmark sculptured surfaces with reference to their corresponding experimental results. Similar results concerning interaction effects among tool path parameters referring to individual criteria have been also investigated, whilst the exact contribution in the form of percentages for all tool path parameters and objectives have been categorized accordingly.

A Pareto chart indicates the absolute values for standardized effects in descending order. Standardized effects are t-statistical results and as such they test a null assumption that an effect is zero. The chart is accompanied to a reference line for indicating the statistically significant effects. The reference line's position on the Pareto chart depends on the level of significance (dictated by α term or "alpha"). The reference line's value is determined according to the method for selecting terms in the regression model to be created (i.e. stepwise, backwards or forward) and the significance level selected (i.e. alpha = 0.05 or 95%). Through a Pareto chart it is possible to determine significant effects, yet, to determine which of them increase or reduce the objective under question, a normal plot of the effects is needed. Such a plot can reveal the magnitude, the direction and impact of the effects. Normal plot of the effects indicates the standardized effects accompanied to a reference line representing a distribution fit. Positive effects are dictated in the case where settings change from low to high parameter levels to increase the objective under question whilst negative effects are shown in the case where settings change from low to high parameter levels to reduce the objective under question. Effects further from 0 regarding X-axis (standardized effect) suggest higher magnitudes and consequently statistically significant results whilst the magnitude of significance is given by their distances from the reference line. Finally, these distances are depended on the selected level of significance.

Pareto charts and normal plots of the standardized effects of tool path parameters on the Pareto criterion were generated for conducting a deeper analysis that the one preceded referring to the main effects. Figure 3.7 illustrates the resulting charts and normal plots for all benchmark sculptured

surfaces. A straightforward indication of these results is that, main effects of tool path parameters as well as their interactions up to the 3rd order have an entirely different behaviour both in order, and magnitude.

Pareto Chart of the Standardized Effects (SS-1)
(Pareto criterion, α = 0.05)
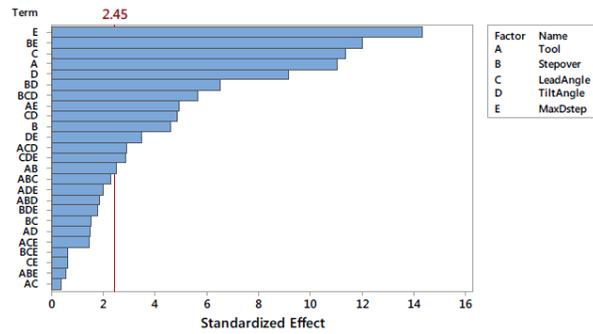
Normal Plot of the Standardized Effects (SS-1)
(Pareto criterion, α = 0.05)

Pareto Chart of the Standardized Effects (SS-2)
(Pareto criterion, α = 0.05)
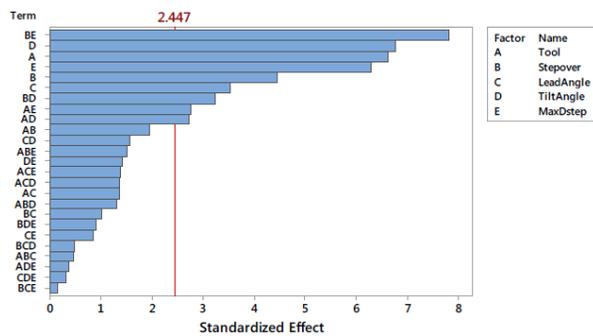
Normal Plot of the Standardized Effects (SS-2)
(Pareto criterion, α = 0.05)
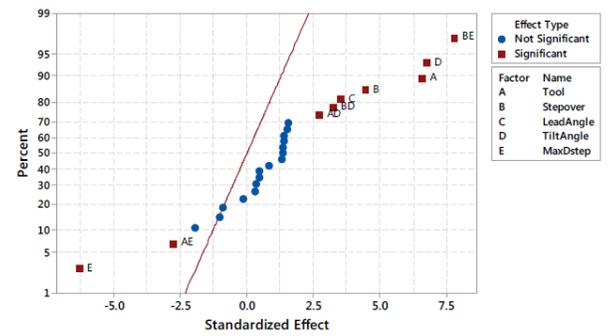
Pareto Chart of the Standardized Effects (SS-3)
(Pareto criterion, α = 0.05)
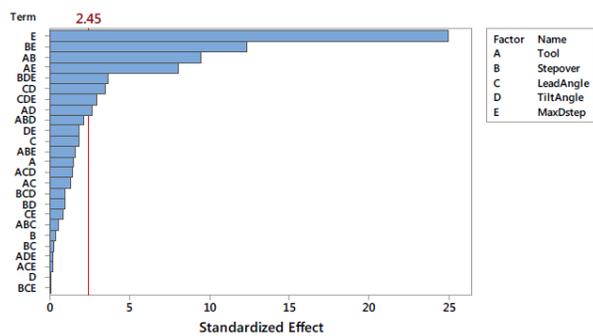
Normal Plot of the Standardized Effects (SS-3)
(Pareto criterion, α = 0.05)

Pareto Chart of the Standardized Effects (SS-4)
(Pareto criterion, α = 0.05)
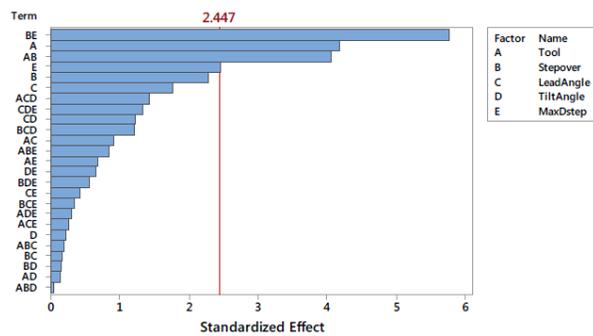
Normal Plot of the Standardized Effects (SS-4)
(Pareto criterion, α = 0.05)

Figure 3.7: Pareto charts and normal plots for the standardized effects on Pareto criterion, per benchmark sculptured surface experiment, SS-1, SS-2, SS-3 and SS-4.

By examining the Pareto chart and the normal plot of the standardized effects for the first benchmark sculptured surface (SS-1) it is shown that the most significant effects are those of MaxDstep, product of Stepover*MaxDstep, lead angle, cutting tool, tilt angle and product of stepover*tilt angle. MaxDstep and cutting tool parameters have the largest negative distance from the normal plot's reference line which means that the magnitude of Pareto criterion is reduced when changing levels from low to high. This result is in total agreement with the main effects plot generated for the case of sculptured surface (SS-1) and Pareto criterion. The product stepover*MaxDstep and lead angle have the largest positive distance from the normal plot's reference line which means that the magnitude of Pareto criterion is increased when changing levels from low to high. This result also agrees with the main effects plot generated for the case of sculptured surface (SS-1) and Pareto criterion, at least for lead angle parameter. The product stepover*MaxDstep comes first in the hierarchy of effects in the case of the second sculptured surface (SS-2) and it is followed by the effects of tilt angle, cutting tool, MaxDstep, stepover and lead angle parameters. MaxDstep parameter has the largest negative distance from the reference line followed by the product cutting tool*MaxDstep. The product stepover*MaxDstep, tilt angle and cutting tool have the largest positive distance from the reference line. These results are also in agreement with those reported in the main effects plot for SS-2 and Pareto criterion referring to the linear terms (MaxDstep, tilt angle and cutting tool). As regards the results for the sculptured surface (SS-3), the effects of MaxDstep parameter, stepover*MaxDstep product, cutting tool*stepover product, cutting tool*MaxDstep product, stepover*tilt angle*MaxDstep product and finally lead angle*tilt angle product are statistically significant at the 0.05 level. The product stepover*tilt angle*MaxDstep has a negative standardized effect with the largest distance whilst MaxDstep parameter and stepover*MaxDstep product have positive standardized effects with MaxDstep to have the largest positive distance. For the fourth sculptured surface (SS-4) the product stepover*MaxDstep, cutting tool, the product cutting tool*stepover and finally MaxDstep parameter are statistically significant. Cutting tool and MaxDstep are the only tool path parameters to be on the left side of the corresponding normal plot of standardized effects with cutting tool to have the largest negative distance. The products stepover*MaxDstep and cutting tool*MaxDstep are the only factors with positive distance on the right side of the normal plot of standardized effects. The product stepover*MaxDstep has the largest positive distance. The exact contributions of all terms' effects based on the experiments conducted for the surfaces SS-1, SS-2, SS-3 and SS-4 are given to Tables 3.6, 3.7, 3.8 and 3.9 respectively.

Table 3.6: Factorial regression analysis and ANOVA contributions of all model terms for sculptured surface (SS-1).

Benchmark sculptured surface SS-1

Objective		$\bar{\delta}$	$\bar{h}$	$\overline{stdev\delta}$	$\overline{stdevh}$	NoCLs	ME $\dfrac{}{\bar{\delta}+\bar{h}}$	$\dfrac{\overline{stdev\delta}}{\overline{stdevh}}+$	OF_{PO}
	(%) Contributions								
	Model	99.58	99.95	90.90	99.86	99.99	99.87	97.47	99.33
	Linear terms	68.34	95.57	44.51	70.15	87.75	85.17	68.69	62.96
Tool		5.40	1.77	5.04	16.97	0.05	4.67	15.29	13.61
Stepover		11.38	91.69	18.68	9.41	55.73	62.41	19.05	2.36
Lead angle		28.96	0.06	15.52	0.02	0.01	10.45	4.23	14.46
Tilt angle		18.67	0.20	0.98	0.13	0.00	7.57	0.61	9.42
MaxDstep		3.93	1.85	4.28	43.61	31.96	0.08	29.51	23.10
	2-Way interactions	21.89	3.08	26.79	25.57	12.17	10.21	22.56	28.89
Tool*Stepover		0.54	0.00	3.94	4.90	0.03	0.16	6.40	0.69
Tool*LeadAngle		0.00	0.05	0.75	0.12	0.08	0.01	0.06	0.02
Tool*TiltAngle		1.15	0.00	0.57	0.03	0.00	0.35	0.09	0.25
Tool*MaxDstep		0.01	1.24	0.14	13.44	0.01	0.54	4.77	2.71
Stepover*LeadAngle		2.15	0.08	10.71	0.02	0.00	1.02	2.86	0.26
Stepover*TiltAngle		14.55	0.07	0.00	0.01	0.02	5.51	0.00	4.73
Stepover*MaxDstep		0.17	0.86	1.39	6.63	12.03	0.12	5.39	16.20
LeadAngle*TiltAngle		3.08	0.77	0.19	0.39	0.00	2.41	0.03	2.62
LeadAngle*MaxDstep		0.16	0.00	0.77	0.00	0.00	0.04	0.25	0.04
TiltAngle*MaxDstep		0.06	0.01	8.33	0.01	0.00	0.05	2.72	1.37
	3-Way interactions	9.36	1.30	19.60	4.14	0.07	4.49	6.22	7.48
Tool*Stepover*LeadAngle		0.45	0.02	1.15	0.01	0.04	0.22	0.28	0.59
Tool*Stepover*TiltAngle		1.15	0.04	1.07	0.06	0.00	0.25	0.17	0.39
Tool*Stepover*MaxDstep		0.00	0.58	0.32	3.64	0.01	0.21	0.87	0.03
Tool*LeadAngle*TiltAngle		2.07	0.07	4.09	0.22	0.00	0.43	0.64	0.94
Tool*LeadAngle*MaxDstep		0.00	0.00	2.02	0.01	0.02	0.01	0.73	0.24
Tool*TiltAngle*MaxDstep		0.00	0.01	2.44	0.00	0.00	0.01	0.69	0.44
Stepover*LeadAngle*TiltAngle		5.34	0.54	0.98	0.01	0.01	3.17	0.22	3.55
Stepover*LeadAngle*MaxDstep		0.09	0.00	0.18	0.00	0.00	0.03	0.06	0.04
Stepover*TiltAngle*MaxDstep		0.14	0.03	4.09	0.03	0.00	0.10	1.00	0.35
LeadAngle*TiltAngle*MaxDstep		0.10	0.02	3.26	0.16	0.00	0.07	1.55	0.90
	Error	0.42	0.05	9.10	0.14	0.01	0.13	2.53	0.67
	Total	100	100	100	100	100	100	100	100

Table 3.7: Factorial regression analysis and ANOVA contributions of all model terms for sculptured surface (SS-2).

Benchmark sculptured surface SS-2

Objective		$\bar{\delta}$	$\bar{h}$	$\overline{stdev\delta}$	$\overline{stdevh}$	NoCLs	ME $\dfrac{}{\bar{\delta}+\bar{h}}$	$\dfrac{\overline{stdev\delta}}{\overline{stdevh}}+$	OF_{PO}
	(%) Contributions								
	Model	98.00	99.80	83.57	91.96	100.00	98.35	99.89	97.82
	Linear terms	90.09	98.46	25.77	61.39	87.75	92.08	74.16	58.66
Tool		7.50	0.54	3.51	7.62	0.00	6.06	10.32	15.88
Stepover		13.06	97.38	10.02	11.21	56.63	58.31	19.06	7.21
Lead angle		11.75	0.00	1.77	0.84	0.00	4.51	0.00	4.55
Tilt angle		44.72	0.11	2.12	0.64	0.55	20.58	0.01	16.57
MaxDstep		13.06	0.42	8.36	41.08	30.57	2.62	44.76	14.44
	2-Way interactions	6.47	0.80	31.12	19.11	12.12	4.69	22.73	35.77
Tool*Stepover		0.00	0.04	3.44	2.12	0.00	0.04	4.62	1.40
Tool*LeadAngle		1.25	0.00	2.64	1.31	0.00	0.51	0.00	0.68
Tool*TiltAngle		2.51	0.26	2.65	1.74	0.00	2.27	0.04	2.68

		$\bar{\delta}$	$\bar{h}$	$\overline{stdev\delta}$	$\overline{stdevh}$	NoCLs	ME	$\overline{\frac{stdev\delta}{stdevh}}$	OF_{PO}
Tool*MaxDstep		0.00	0.19	3.36	4.02	0.00	0.19	6.67	2.78
Stepover*LeadAngle		1.05	0.01	1.83	1.56	0.00	0.55	0.09	0.38
Stepover*TiltAngle		0.87	0.02	2.30	0.71	0.54	0.54	0.01	3.82
Stepover*MaxDstep		0.10	0.16	7.89	5.41	11.44	0.04	11.25	22.17
LeadAngle*TiltAngle		0.67	0.03	2.76	0.90	0.00	0.49	0.01	0.88
LeadAngle*MaxDstep		0.01	0.08	2.00	0.66	0.00	0.05	0.01	0.26
TiltAngle*MaxDstep		0.01	0.00	2.26	0.67	0.12	0.00	0.02	0.73
	3-Way interactions	1.45	0.54	26.68	11.46	0.13	1.59	3.01	3.39
Tool*Stepover*LeadAngle		0.18	0.00	2.66	1.63	0.00	0.07	0.03	0.08
Tool*Stepover*TiltAngle		0.07	0.17	2.70	0.93	0.00	0.33	0.01	0.62
Tool*Stepover*MaxDstep		0.10	0.07	3.31	0.80	0.00	0.00	2.82	0.84
Tool*LeadAngle*TiltAngle		0.23	0.13	2.76	0.79	0.00	0.45	0.02	0.68
Tool*LeadAngle*MaxDstep		0.48	0.08	2.70	1.12	0.00	0.51	0.00	0.69
Tool*TiltAngle*MaxDstep		0.00	0.02	2.67	1.51	0.00	0.01	0.02	0.05
Stepover*LeadAngle*TiltAngle		0.01	0.06	2.73	1.80	0.00	0.09	0.04	0.08
Stepover*LeadAngle*MaxDstep		0.02	0.02	2.05	1.33	0.00	0.00	0.03	0.01
Stepover*TiltAngle*MaxDstep		0.10	0.00	2.40	0.77	0.12	0.05	0.01	0.30
LeadAngle*TiltAngle*MaxDstep		0.25	0.00	2.69	0.77	0.00	0.07	0.02	0.04
	Error	2.00	0.20	16.43	8.04	0.00	1.65	0.11	2.18
	Total	100	100	100	100	100	100	100	100

Table 3.8: Factorial regression analysis and ANOVA contributions of all model terms for sculptured surface (SS-3).

Benchmark sculptured surface SS-3

Objective		$\bar{\delta}$	$\bar{h}$	$\overline{stdev\delta}$	$\overline{stdevh}$	NoCLs	ME $\frac{}{\bar{\delta}+\bar{h}}$	$\overline{\frac{stdev\delta}{stdevh}}$	OF_{PO}
	(%) Contributions								
	Model	99.97	99.91	99.82	99.71	100.00	99.92	99.43	99.40
	Linear terms	99.11	97.21	71.33	77.10	88.00	97.33	79.56	62.86
Tool		0.07	0.60	0.02	10.92	0.01	0.47	7.78	0.21
Stepover		0.00	95.07	34.70	7.25	26.41	32.35	70.92	0.01
Lead angle		0.01	0.00	0.19	0.02	0.00	0.01	0.32	0.32
Tilt angle		0.03	0.29	0.24	0.03	0.00	0.03	0.13	0.00
MaxDstep		99.00	1.24	36.19	58.88	61.58	64.47	0.40	62.32
	2-Way interactions	0.45	2.50	25.64	19.52	12.00	1.96	16.89	33.29
Tool*Stepover		0.18	1.55	0.07	3.22	0.00	1.20	1.73	9.06
Tool*LeadAngle		0.00	0.02	0.05	0.00	0.00	0.01	0.09	0.16
Tool*TiltAngle		0.01	0.02	0.45	0.05	0.00	0.02	0.78	0.69
Tool*MaxDstep		0.07	0.59	0.18	9.77	0.01	0.47	5.45	6.54
Stepover*LeadAngle		0.00	0.01	0.10	0.04	0.00	0.00	0.02	0.01
Stepover*TiltAngle		0.02	0.00	0.39	0.01	0.00	0.01	0.52	0.08
Stepover*MaxDstep		0.00	0.19	23.35	6.36	11.99	0.10	7.38	15.14
LeadAngle*TiltAngle		0.13	0.01	0.47	0.03	0.00	0.14	0.29	1.21
LeadAngle*MaxDstep		0.01	0.01	0.10	0.03	0.00	0.00	0.21	0.06
TiltAngle*MaxDstep		0.03	0.11	0.48	0.01	0.00	0.00	0.41	0.33
	3-Way interactions	0.41	0.21	2.85	3.09	0.00	0.63	2.98	3.24
Tool*Stepover*LeadAngle		0.00	0.01	0.02	0.04	0.00	0.00	0.09	0.03
Tool*Stepover*TiltAngle		0.01	0.01	0.49	0.00	0.00	0.01	0.60	0.46
Tool*Stepover*MaxDstep		0.23	0.04	0.91	2.86	0.00	0.28	0.27	0.26
Tool*LeadAngle*TiltAngle		0.00	0.01	0.03	0.05	0.00	0.01	0.14	0.20
Tool*LeadAngle*MaxDstep		0.00	0.01	0.01	0.00	0.00	0.01	0.02	0.00
Tool*TiltAngle*MaxDstep		0.01	0.00	0.02	0.04	0.00	0.01	0.09	0.00
Stepover*LeadAngle*TiltAngle		0.00	0.01	0.43	0.00	0.00	0.01	0.48	0.09
Stepover*LeadAngle*MaxDstep		0.00	0.01	0.05	0.04	0.00	0.00	0.00	0.00
Stepover*TiltAngle*MaxDstep		0.04	0.02	0.90	0.03	0.00	0.07	1.24	1.33
LeadAngle*TiltAngle*MaxDstep		0.11	0.10	0.00	0.04	0.00	0.23	0.05	0.86
	Error	0.03	0.09	0.18	0.29	0.00	0.08	0.57	0.60
	Total	100	100	100	100	100	100	100	100

Table 3.9: Factorial regression analysis and ANOVA contributions of all model terms for sculptured surface (SS-4).

Objective	$\bar{\delta}$	$\bar{h}$	$\overline{stdev\delta}$	$\overline{stdevh}$	NoCLs	$\dfrac{ME}{\bar{\delta}+\bar{h}}$	$\dfrac{\overline{stdev\delta}+}{\overline{stdevh}}$	OF_{PO}
Benchmark sculptured surface SS-4								
(%) Contributions								
Model	94.43	99.91	95.46	99.25	99.88	98.05	97.30	93.87
Linear terms	62.85	98.97	41.71	66.23	81.16	87.11	68.73	32.72
Tool	3.85	0.15	4.07	16.18	0.05	2.21	24.67	17.95
Stepover	3.51	98.38	17.56	7.85	40.54	71.22	27.05	5.37
Lead angle	0.01	0.10	5.74	0.09	0.00	0.06	3.14	3.19
Tilt angle	2.12	0.05	0.40	0.24	0.02	0.38	0.72	0.05
MaxDstep	53.36	0.29	13.93	41.88	40.55	13.23	13.15	6.16
2-Way interactions	17.40	0.76	35.27	26.04	18.45	6.58	21.14	54.38
Tool*Stepover	4.96	0.25	3.85	4.01	0.01	3.05	9.44	16.85
Tool*LeadAngle	0.26	0.00	4.39	0.00	0.00	0.12	1.81	0.86
Tool*TiltAngle	1.13	0.01	3.07	0.16	0.02	0.24	2.15	0.02
Tool*MaxDstep	0.80	0.39	3.13	10.36	0.01	0.01	3.45	0.47
Stepover*LeadAngle	4.06	0.00	4.22	0.34	0.04	1.41	3.31	0.03
Stepover*TiltAngle	0.28	0.03	0.68	0.03	0.01	0.23	0.13	0.02
Stepover*MaxDstep	0.80	0.01	11.96	10.13	18.24	0.37	0.59	33.96
LeadAngle*TiltAngle	2.55	0.02	0.57	0.28	0.03	0.59	0.00	1.55
LeadAngle*MaxDstep	0.62	0.02	3.23	0.72	0.08	0.10	0.12	0.17
TiltAngle*MaxDstep	1.92	0.01	0.18	0.01	0.02	0.45	0.14	0.44
3-Way interactions	14.18	0.18	18.48	6.98	0.27	4.36	7.43	6.77
Tool*Stepover*LeadAngle	3.57	0.03	3.77	0.15	0.05	0.85	2.46	0.03
Tool*Stepover*TiltAngle	0.34	0.01	2.87	0.07	0.02	0.20	0.67	0.00
Tool*Stepover*MaxDstep	1.48	0.04	2.96	6.05	0.04	0.77	1.41	0.72
Tool*LeadAngle*TiltAngle	2.96	0.01	0.32	0.14	0.02	0.83	0.00	2.10
Tool*LeadAngle*MaxDstep	0.19	0.01	2.64	0.34	0.05	0.03	0.23	0.07
Tool*TiltAngle*MaxDstep	1.06	0.00	1.92	0.03	0.02	0.32	1.07	0.09
Stepover*LeadAngle*TiltAngle	0.95	0.01	0.74	0.00	0.03	0.40	0.30	1.49
Stepover*LeadAngle*MaxDstep	0.97	0.03	2.81	0.00	0.01	0.16	1.17	0.12
Stepover*TiltAngle*MaxDstep	0.27	0.04	0.14	0.20	0.03	0.23	0.03	0.32
LeadAngle*TiltAngle*MaxDstep	2.39	0.02	0.30	0.00	0.02	0.57	0.08	1.83
Error	5.57	0.09	4.54	0.75	0.12	1.95	2.70	6.13
Total	100	100	100	100	100	100	100	100

It was reported in section 3.6.2 of this thesis that experiments were based on machining simulations conducted using a CAM system whilst two automation functions were developed and deployed to automatically provide computational results for chordal deviation and scallop height for each cutting point of a tool path. It was also mentioned that computational results for scallop height were compared to real-time deviation measurements taken on scallop curves of 3D CAM outputs for all sculptured surfaces examined, by applying virtual probing techniques. The following paragraphs report the results obtained by conducting different tests to verify the applicability of the automation functions responsible for automatically computing scallop height and chordal deviation.

To evaluate the consistency of scallop height analytical formula given in Eq.3.10, the results of analytical computations and virtual measurements were considered as two independent populations

with different size. The pairs of populations were individually examined for each benchmark sculptured surface to prove the assumption that there is no statistically significant difference between their means against the alternative which suggests difference, under the significance level of alpha 0.05. It should be mentioned that resulting means from analytical computations provided by the corresponding automation function give a true figure of the average since they derive from the entire populations of computational results and not from samples of them. The same cannot be claimed in the case of the populations of virtual measurements whose sizes vary significantly against those referring to computational results. However, the necessity to show whether computational results agree with experimental ones taken from virtually machined models (and to what extend) is of major importance since actual CNC machining is based on process planning that involves machining simulations in CAM environment. The *separate variance 2-sample t-test* (*non-pooled t-test*) was selected and applied under the assumption that there is no difference between the means of paired populations of analytical and experimental results against the alternative, considering the standard significance level in the literature, that of $\alpha=0.05$. Figures 3.8, 3.9, 3.10 and 3.11 depict the results of t-tests conducted for all benchmark sculptured surfaces, SS-1, SS-2, SS-3 and SS-4 respectively. The red line in the illustrations represents the significance level which is represented by p-value. According to descriptive statistics, magnitudes that exceed a p-value equal to 0.05 imply statistically insignificant results. On the contrary magnitudes equal to p-values of 0.05 or less dictate statistically significant results between objectives under comparison. It is observed for the benchmark sculptured surfaces tested that most of p-values do not reject the null hypothesis, indicating thus concrete evidences for statistically insignificant difference among the means of analytical computations and experimental measurements for scallop heights. Analytical results for scallop heights referring to the experiments conducted for surface (SS-1) and the corresponding 2-sample t-test were found to be in agreement to the percentage of 87.5%. As it can be seen in Fig.3.8, 4 out of 32 comparative populations' means had statistically significant differences which for the case of SS-1 it is interpreted to the percentage of 12.5%. In the case of the second surface (SS-2) the same results were found (Fig.3.9), whilst for the third sculptured surface (SS-3) the success in achieving statistically insignificant differences among the means of analytical computations and experimental measurements for scallop heights reached 81.25%. In this case, 6 out of 32 comparative populations' means had statistically significant differences, with a result equal to 18.75% (Fig.3.10). As regards the fourth sculptured surface (SS-4) all 32 comparative populations' means were found to have statistically insignificant differences (Fig.3.11). By considering all 128 (4*32) experimental runs for the overall estimation of scallop height, statistically significant differences among the means of analytical computations and experimental measurements for scallop heights span 14 experiments. This can be

73

given as a percentage equal to 10.94%. Based on these results the formula given in Eq.3.10 for computing scallop heights can be considered as being a quite reliable attribute. The magnitudes of p-values per benchmark sculptured surface are also summarized in Tables 3.10, 3.11, 3.12 and 3.13 for the separate variance 2-sample t-tests referred to sculptured surfaces SS-1, SS-2, SS-3 and SS-4 respectively. In the tables, the 1st column is assigned to the number of experiments, the 2nd to the population size of computational results for scallop heights, the 3rd to the population size of experimental measurements for scallop heights, the 4th to the means of the population size of computational results for scallop heights, the 5th to the means of population size of experimental measurements for scallop heights, the 6th and the 7th to their standard deviations respectively and finally the 8th column is assigned to the resulting p-values. Figs.3.8, 3.9, 3.10 and 3.11 depict also the correlation among analytical and experimental means of scallop heights.
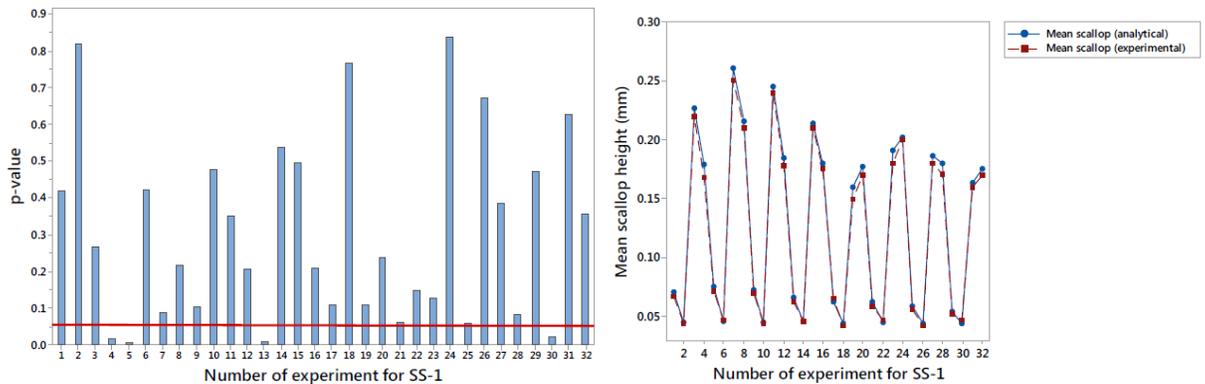
Figure 3.8: 2-sample t-test results for the statistical significance between analytical and experimental means of scallop heights for the benchmark sculptured surface SS-1.

Table 3.10: Detailed results of the 2-sample t-test for the benchmark sculptured surface (SS-1).

SS-1	N h (comp)	N h (exp.)	Mean h (comp)	Mean h (exp.)	StDev h (comp)	StDev h (exp.)	P-value
1	13927	1000	0.125	0.0176	0.0011	0.00056	0.419
2	13455	0015	0.0448	0.0438	0.0651	0.0161	0.819
3	3417	1000	0.227	0.2206	0.333	0.0175	0.264
4	3311	1000	0.179	0.1701	0.209	0.0172	0.015
5	13618	1000	0.075	0.0714	0.134	0.0176	0.005
6	13416	450	0.0458	0.0466	0.0674	0.0172	0.421
7	3345	1000	0.261	0.2498	0.371	0.0176	0.087
8	3301	1000	0.216	0.2104	0.251	0.0174	0.216
9	13825	1000	0.072	0.0699	0.132	0.0171	0.102
10	13328	120	0.0450	0.0438	0.0658	0.0180	0.476
11	3383	1000	0.246	0.2397	0.366	0.0176	0.350
12	3287	1000	0.185	0.1800	0.205	0.0175	0.204
13	13567	1000	0.065	0.0623	0.122	0.0175	0.010
14	13358	450	0.0453	0.0459	0.0668	0.0170	0.537
15	3341	1000	0.214	0.2102	0.310	0.0175	0.494
16	3287	1000	0.180	0.1752	0.206	0.0174	0.209
17	5081	1000	0.062	0.0650	0.117	0.0175	0.108
18	4930	0015	0.0438	0.0423	0.0606	0.0190	0.766
19	1242	1000	0.160	0.1497	0.221	0.0173	0.110
20	1214	1000	0.178	0.1707	0.207	0.0172	0.238

21	5002	1000	0.062	0.0587	0.115	0.0171	0.060
22	4909	450	0.0451	0.0469	0.0673	0.0171	0.149
23	1224	1000	0.191	0.1795	0.260	0.0172	0.128
24	1211	1000	0.202	0.2007	0.223	0.0175	0.837
25	5047	1000	0.058	0.0553	0.107	0.0176	0.058
26	4879	0035	0.0433	0.0421	0.0612	0.0164	0.671
27	1234	1000	0.187	0.1800	0.283	0.0171	0.387
28	1206	1000	0.180	0.1701	0.195	0.0172	0.081
29	4951	1000	0.0535	0.0525	0.0984	0.0174	0.470
30	4885	450	0.0441	0.0469	0.0633	0.0172	0.020
31	1218	1000	0.163	0.1603	0.227	0.0174	0.626
32	1212	1000	0.175	0.1698	0.198	0.0174	0.355
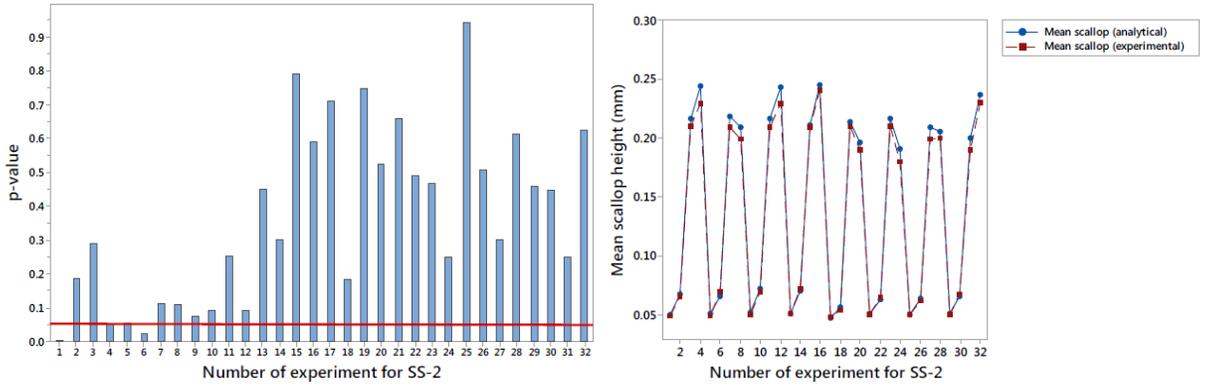
Figure 3.9: 2-sample t-test results for the statistical significance between analytical and experimental means of scallop heights for the benchmark sculptured surface SS-2.

Table 3.11: Detailed results of the 2-sample t-test for the benchmark sculptured surface (SS-2).

SS-2	N h (comp)	N h (exp.)	Mean h (comp)	Mean h (exp.)	StDev h (comp)	StDev h (exp.)	P-value
1	9466	1000	0.0505	0.0476	0.0824	0.0175	0.004
2	9812	800	0.0681	0.0661	0.133	0.0169	0.186
3	2111	800	0.217	0.2104	0.276	0.0174	0.290
4	2174	800	0.245	0.2300	0.364	0.0172	0.053
5	9435	800	0.0508	0.0488	0.0796	0.0175	0.056
6	9846	800	0.066	0.0691	0.125	0.0172	0.022
7	2096	800	0.219	0.2098	0.265	0.0169	0.110
8	2169	800	0.209	0.1995	0.283	0.0179	0.108
9	8736	800	0.0520	0.0501	0.0826	0.0174	0.076
10	8969	800	0.0722	0.0697	0.133	0.0171	0.091
11	1974	800	0.217	0.2098	0.268	0.0176	0.252
12	2011	800	0.244	0.2300	0.365	0.0178	0.093
13	8875	800	0.0515	0.0508	0.077	0.0174	0.449
14	9108	800	0.071	0.0721	0.132	0.0178	0.302
15	2005	800	0.211	0.2100	0.248	0.0178	0.791
16	2046	800	0.245	0.2410	0.373	0.0180	0.589
17	3609	800	0.0478	0.0483	0.0729	0.0171	0.712
18	3741	800	0.057	0.0541	0.107	0.0174	0.183
19	805	800	0.214	0.2108	0.272	0.0173	0.748
20	829	800	0.197	0.1903	0.279	0.0171	0.524
21	3581	800	0.0509	0.0502	0.0836	0.0173	0.660
22	3731	800	0.063	0.0648	0.121	0.0177	0.489
23	799	800	0.217	0.2105	0.255	0.0172	0.466
24	827	800	0.191	0.1804	0.270	0.0171	0.251

25	3341	800	0.0506	0.0505	0.0807	0.0170	0.941
26	3442	800	0.064	0.0621	0.123	0.0175	0.508
27	755	800	0.210	0.1997	0.261	0.0172	0.303
28	769	800	0.206	0.2002	0.309	0.0176	0.613
29	3393	800	0.0513	0.0502	0.0765	0.0173	0.459
30	3455	800	0.066	0.0676	0.124	0.0172	0.446
31	765	800	0.200	0.1903	0.240	0.0171	0.251
32	767	800	0.237	0.2308	0.359	0.0175	0.623
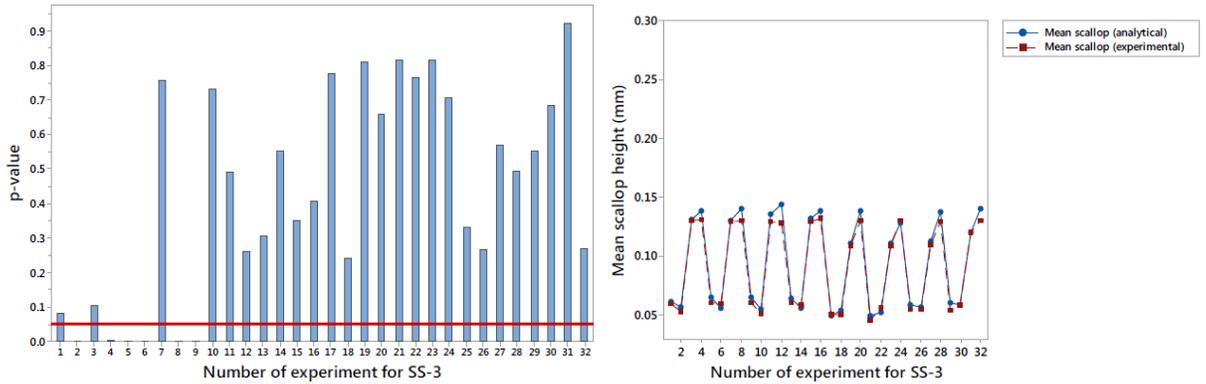
Figure 3.10: 2-sample t-test results for the statistical significance between analytical and experimental means of scallop heights for the benchmark sculptured surface SS-3.

Table 3.12: Detailed results of the 2-sample t-test for the benchmark sculptured surface (SS-3).

SS-3	N h (comp)	N h (exp.)	Mean h (comp)	Mean h (exp.)	StDev h (comp)	StDev h (exp.)	P-value
1	7054	500	0.0614	0.0591	0.0868	0.0176	0.080
2	6962	500	0.0570	0.0528	0.0588	0.0169	0.001
3	2778	500	0.139	0.1304	0.112	0.0179	0.104
4	2778	500	0.139	0.1315	0.112	0.0171	0.002
5	7013	500	0.0651	0.0600	0.0932	0.0173	0.001
6	6953	500	0.0554	0.0594	0.0536	0.0173	0.001
7	2785	500	0.130	0.1314	0.143	0.0171	0.758
8	2768	500	0.140	0.1308	0.116	0.0170	0.001
9	7193	500	0.0649	0.0607	0.0905	0.0170	0.001
10	7093	500	0.0551	0.0548	0.0518	0.0173	0.734
11	2872	500	0.136	0.1341	0.159	0.0172	0.491
12	2847	500	0.144	0.01417	0.121	0.0169	0.261
13	7221	500	0.0635	0.0622	0.0879	0.0172	0.305
14	7119	500	0.0561	0.0567	0.0535	0.0170	0.553
15	2884	500	0.132	0.1298	0.146	0.0170	0.350
16	2851	500	0.139	0.1372	0.105	0.0173	0.407
17	1377	500	0.0497	0.0502	0.0709	0.0178	0.778
18	1371	500	0.0534	0.0514	0.0569	0.0176	0.241
19	549	500	0.111	0.1093	0.139	0.0166	0.811
20	546	500	0.138	0.1358	0.126	0.0174	0.659
21	1397	500	0.0493	0.0488	0.0776	0.0174	0.818
22	1395	500	0.0519	0.0524	0.0508	0.0165	0.766
23	562	500	0.111	0.1091	0.153	0.0176	0.816
24	559	500	0.128	0.1301	0.110	0.0172	0.708
25	1489	500	0.0583	0.0560	0.0856	0.0175	0.330
26	1498	500	0.0568	0.0550	0.0562	0.0171	0.266
27	597	500	0.113	0.1097	0.130	0.0172	0.568
28	590	500	0.137	0.1340	0.117	0.0177	0.495
29	1549	500	0.0583	0.0592	0.0548	0.0174	0.553

30	1549	500	0.0583	0.0589	0.0548	0.0178	0.686
31	621	500	0.121	0.1201	0.140	0.0178	0.922
32	613	500	0.141	0.1355	0.118	0.0168	0.269
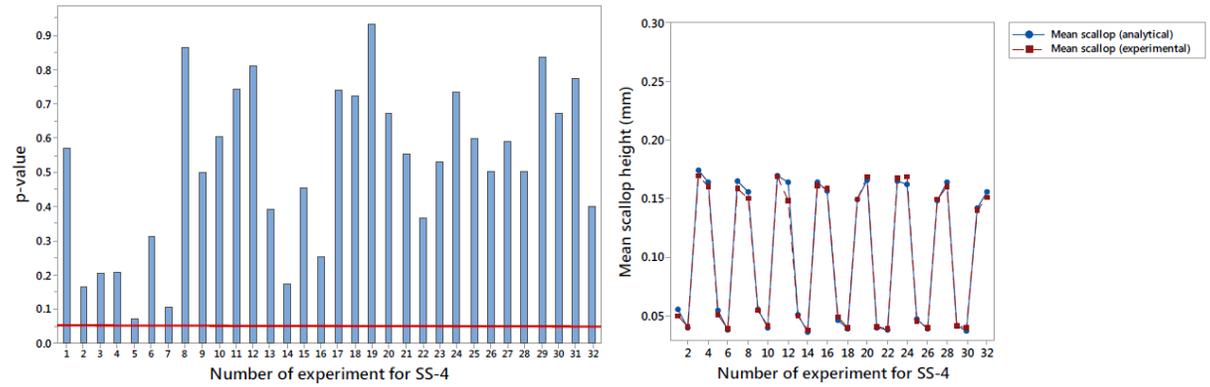
Figure 3.11: 2-sample t-test results for the statistical significance between analytical and experimental means of scallop heights for the benchmark sculptured surface SS-4.

Table 3.13: Detailed results of the 2-sample t-test for the benchmark sculptured surface (SS-4).

SS-4	N h (comp)	N h (exp.)	Mean h (comp)	Mean h (exp.)	StDev h (comp)	StDev h (exp.)	P-value
1	15828	500	0.0557	0.0551	0.0927	0.0173	0.572
2	14711	500	0.0398	0.0386	0.0515	0.0173	0.166
3	3931	500	0.175	0.1702	0.232	0.0179	0.204
4	3644	500	0.164	0.1606	0.164	0.0181	0.207
5	14950	500	0.0544	0.0525	0.0951	0.0171	0.070
6	14744	500	0.0384	0.0393	0.0497	0.0171	0.313
7	3705	500	0.165	0.1593	0.219	0.0174	0.103
8	3661	500	0.156	0.1555	0.157	0.0172	0.863
9	15625	500	0.0559	0.0551	0.0953	0.0177	0.499
10	14693	500	0.0402	0.0398	0.0529	0.0177	0.603
11	3900	500	0.170	0.1693	0.224	0.0174	0.744
12	3639	500	0.164	0.1637	0.165	0.0169	0.812
13	14909	500	0.0508	0.0499	0.0878	0.0181	0.393
14	14746	500	0.0363	0.0374	0.0435	0.0167	0.171
15	3697	500	0.164	0.1615	0.218	0.0171	0.455
16	3664	500	0.157	0.1542	0.160	0.0169	0.252
17	3444	500	0.0465	0.0470	0.0815	0.0170	0.742
18	3287	500	0.0392	0.0388	0.0503	0.0169	0.725
19	851	500	0.150	0.1494	0.221	0.0176	0.932
20	793	500	0.167	0.1642	0.167	0.0169	0.673
21	3546	500	0.0404	0.0396	0.0703	0.0173	0.552
22	3577	500	0.0379	0.0389	0.0470	0.0172	0.368
23	3705	500	0.165	0.1630	0.219	0.0177	0.530
24	852	500	0.163	0.1652	0.181	0.0172	0.736
25	3431	500	0.0472	0.0463	0.0833	0.0177	0.599
26	3284	500	0.0395	0.0387	0.0510	0.0175	0.503
27	855	500	0.148	0.1447	0.202	0.0171	0.593
28	790	500	0.165	0.1607	0.162	0.0171	0.504
29	3552	500	0.0417	0.0414	0.0720	0.0176	0.839
30	3567	500	0.0376	0.0381	0.0457	0.0169	0.673
31	861	500	0.142	0.1407	0.176	0.0180	0.774
32	852	500	0.156	0.1514	0.152	0.0179	0.400

Unlike scallop height which can be computed and measured directly on the machined surface, chord error can only be limited to analytical computations for estimating its magnitude. The attributes on which chord error $\delta_{i,i+1}$ is dependent are local curvature ρ_i and 3D distance (chord length) $L_{i,i+1}$ (see also Eq.3.4, 3.5 and 3.6). Both these geometric elements depend on the positions of the successive cutting points which in turn depend on the selected cutting tolerance set by the user during tool path planning. Based on this, chord error $\delta_{i,i+1}$ was automatically computed by using CATIA application programming interface (API) for all successive pairs of cutting points. To achieve successful computations for estimating chord error $\delta_{i,i+1}$ (chordal deviation) two private functions were developed in *Visual Basic®*, one for computing subsequent 3D distances (chord lengths) $L_{i,i+1}$ and one for computing subsequent local curvatures ρ_i. Note that for j number of cutting points, resulting chord lengths are $j-1$ since two cutting points designate one chord length.

To compute subsequent 3D distances (chord lengths) $L_{i,i+1}$ the j th cutting point was determined to be the first measurable reference and the $j+1$ cutting point was determined to be the second measurable reference. Thereby the automation property "*Get_Distance*" was applied to measure $L_{i,i+1}$ for all cutting points comprising the tool path. Local curvatures ρ_i were calculated with reference to three fundamental instances. The first refers to the main direction of cutting points which is the trajectory forward to feed. The second is the group of vectors normal to surface whilst the third is the angle between two subsequent normal vectors. Normal vectors are as much in magnitude as the number of cutting points and they were generated via the automation properties "*Create_Reference_Form_Object*", "*Add_New_Projection*" and "*Add_New_Line_PtPt*". The first automation property was implemented to set cutting points as references for creating a new collection of points. The second automation property creates these points as projections normal to surface under a predefined distance. The third automation property undertakes to connect all cutting points to their corresponding projected points from the collection to finally take the normal vectors. To compute the angle between each pair of normal vectors the "*Get_Angle_Between*" automation property of CATIA API was used. With reference to the normal vectors (let $\vec{n_i}$ and $\vec{n_{i+1}}$ be a pair) all angles $\theta_{i,i+1}$ were programmatically obtained by CAM software whereas local curvatures $\rho_{i,i+1}$ (mm⁻¹) were computed by passing Eq.3.5 to the corresponding private function. Chordal deviation $\delta_{i,i+1}$ (mm) was computed by passing Eq.3.6 to the same private function. Both private functions for 3D distances (chord lengths) $L_{i,i+1}$ and local curvatures ρ_i constitute parts of the integrated programming module

which is presented in Chapter 4. Several tests were conducted to evaluate the accuracy of results concerning the aforementioned computations obtained by the functions mentioned above. The results of 3D distances (chord lengths) and local curvatures were compared to manual measurements taken by examining arbitrary designed sculptured surfaces for which the automation functions were applied. Manual measurements were obtained by using the *surfacic curvature analysis* tool available to the *advanced free-form surface design* environment of *CATIA® V5 R18.* Figure 3.12 illustrates one of the arbitrary sculptured surfaces examined along with results manually and programmatically obtained for measuring 3D distances, Figure 3.13 shows the results obtained manually and automatically for the angles between generated normal vectors for the same surface and Figure 3.14 shows the manual results corresponding to local curvatures.
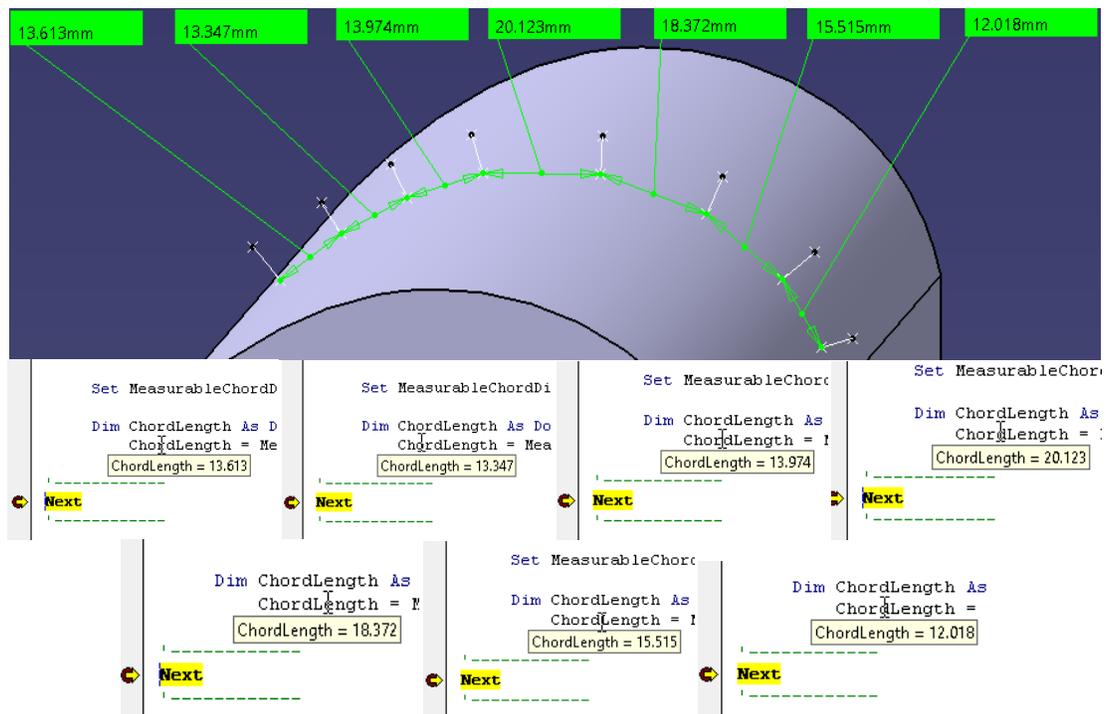
Figure 3.12: Manual test results for examining the accuracy of automation function developed for computing 3D distances (chord lengths).
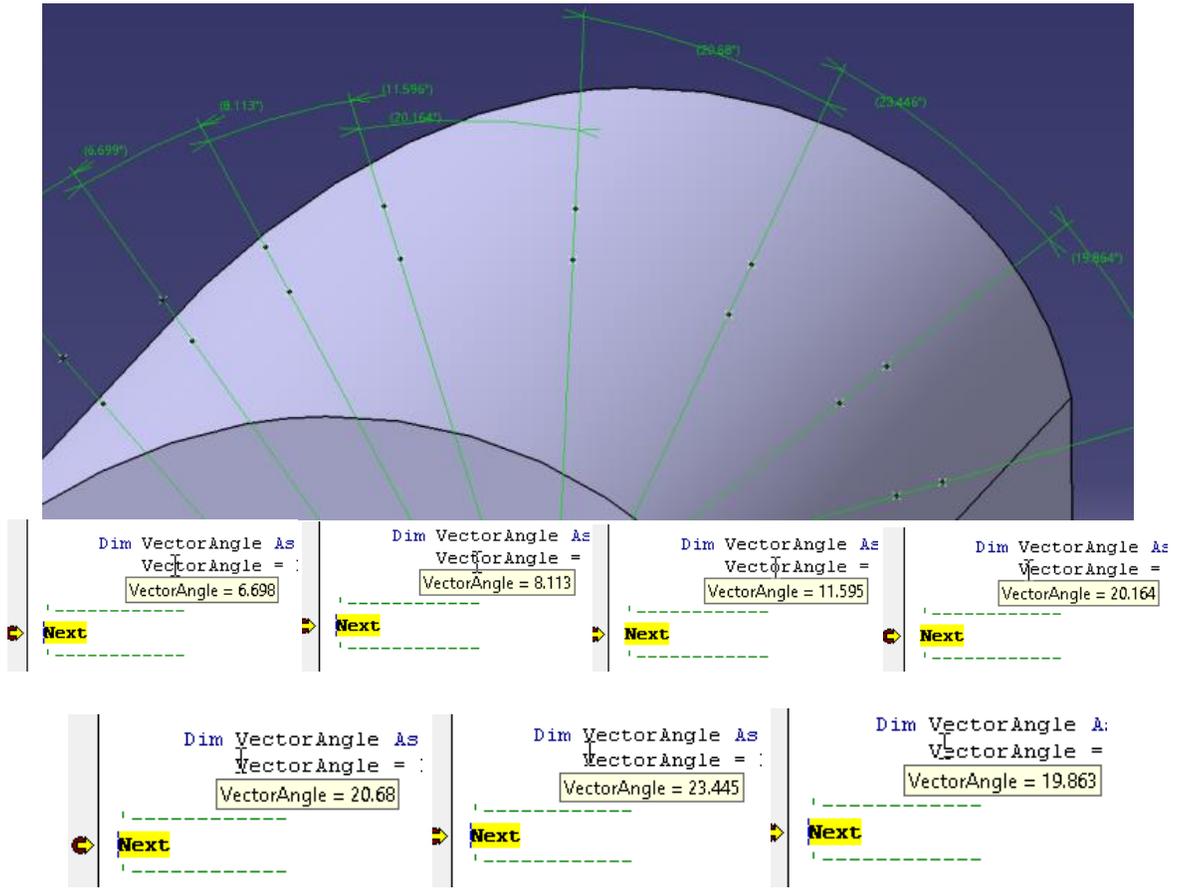
Figure 3.13: Manual test results for examining the accuracy of automation function developed for computing angles between normal vectors.
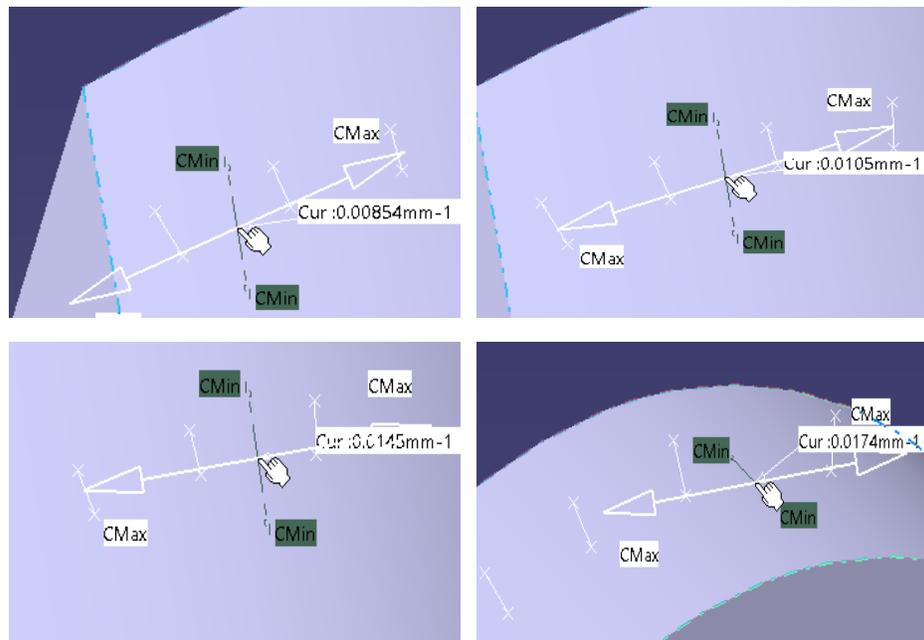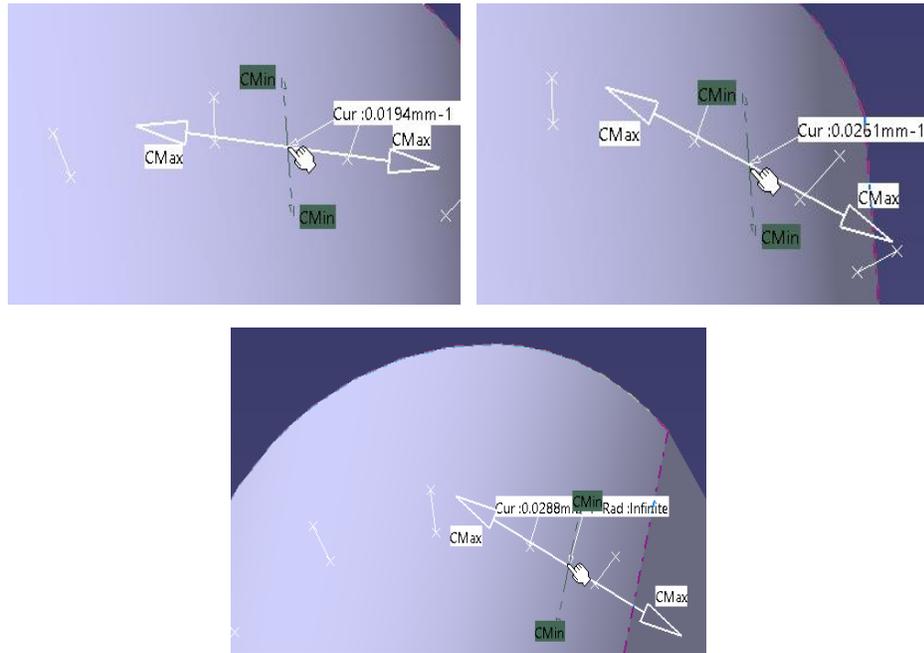
Figure 3.14: Manual test results for examining the accuracy of automatically computed local curvatures.

3.7   Conclusions

A global sculptured surface machining optimisation methodology should first focus on the appropriate formulation of the problem to be explored and then implement accurate elements with generic capabilities. Based on the results presented in this chapter, it can be deduced that all tool path parameters have significant effects on the quality characteristics set. Significance of their effects, changes in both magnitude and hierarchy when planning tool paths to machine different surfaces even if adopting the same cutting strategy. As much as it has been shown, it is neither safe nor profitable to try distinguishing the significance among tool path parameters according to the sculptured surface under investigation. Instead the same gravity should be given to all tool path parameters in order to ensure that the optimisation methodology will address the sculptured surface machining problem impartially, globally and stochastically.

The effects of the tool path parameters were studied using the two-level full factorial design of experiments as one of the available experiment design methods. The overall conclusion stemming from this study is the fact that corresponding results cannot be generalized for every case referring to the sculptured surface machining problem. However, the experiments conducted provide a good insight on the qualitative comparison between the different effects of tool path parameters on the criteria, with reference to the benchmark sculptured surfaces examined.

Experiments were conducted not only to study the behaviour of tool path parameters on the problem's criteria but also to evaluate the consistency of computed outputs given by analytical expressions for scallop height and chordal deviation estimation. The results obtained from the developed programming functions that involve the aforementioned analytical expressions were found to be in a very good agreement with those experimentally obtained by examining the benchmark sculptured surfaces. This success can be attributed to the consistency of the performance of CAM functions and to the objectives set for representing the problem through the Pareto criterion.

Based on the detailed experimental study presented for the tool path parameters as well as on the fact that its outputs lack generality as they depend on the sculptured surface geometry, the precision of tool path parameters' values when they are represented as binary strings (chromosomes), should be of the same number.

Chapter 4

Optimisation methodology for sculptured surface CNC machining

4.1 Introduction

This chapter presents the development stages followed to establish the optimisation methodology for solving the sculptured surface CNC machining problem. The methodology refers to the simultaneous 5-axis CNC machining since this technology is by far more superior to others implemented in manufacturing industry. The methodology is consisted of two parts. The first part is responsible for automating CAM software functions as well as for computing the optimisation criteria, machining error, machining error distribution and number of cutting points, as it has been mentioned in Chapter 3 (section 3.4). The second part undertakes the multi-objective optimisation process by implementing a virus-evolutionary genetic algorithm developed for this scope, regarding the problem definition reported in Chapter 3 (section 3.2). Such an algorithm has never been proposed to address the sculptured surface machining problem whilst its significant architecture and differentiation from other evolutionary algorithms has caught the interest for its development and implementation for solving the sculptured surface machining problem.

It is of great importance to mention that the proposed methodology has been developed such that any representation of the sculptured surface machining problem can be handled as long as the objective function is properly formulated, and its objectives ensure generic results. It is also crucial to report that the multi-objective virus-evolutionary genetic algorithm accounts for the methodology's successful implementation but doesn't represent the overall philosophy of stochastically optimising tool paths from multi-axis machining strategies as candidate solutions since any other variant of intelligent algorithms can be implemented instead. The usage of an evolutionary algorithm other than the virus-evolutionary genetic algorithm proposed in this thesis can be suitable provided that its components and functions are compatible with the automation part of the methodology. The same also goes for the function undertaking to automate CAM system's utilities and evaluate the optimisation criteria adopted in this research.

4.2 Fundamentals of genetic and evolutionary algorithms

Genetic algorithms (GAs) are stochastic functions for searching solutions in a potential solution domain that mimic the biological evolution of species according to the Darwinian evolution theory. Genetic algorithms operate on populations of potential solutions (individuals or candidate solutions) by following the principle of the "survival of the fittest" so that better solutions gradually are achieved to finally solve a problem. In each generation of potential candidates, a set of functions is applied referring to the selection of individuals in a population regarding their "fitness" for solving a problem and their following reproduction (crossover) to produce better offspring. This procedure leads to the "evolution" of populations consisted of candidate solutions (individuals) that adapt better to their "environment" than the individuals from which first came to be, as it occurs in natural adaptation. Individuals or candidate solutions are encoded as series of numerical characters known as "chromosomes" so that their genotype (the real values of chromosomes) can be unambiguously represented as variable numbers (phenotypes) that describe the problem at hand. The most usual representation for chromosomes is binary, however other schemes such as real and gray binary are available for usage. Figure 4.1 shows an example of a problem involving two independent variables x_1 and x_2 represented using binary encoding where the former variable comprises 10 binary digits whereas the latter comprises 15 binary digits.

$$1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1 \mid 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1$$

$$\underset{x_1}{\longleftrightarrow} \qquad \underset{x_2}{\longleftrightarrow}$$
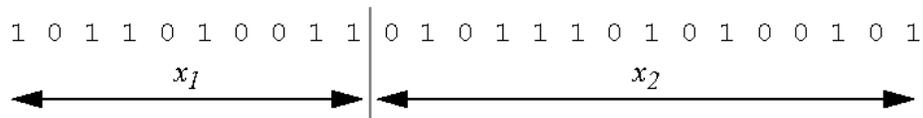
Figure 4.1: Binary representation for two independent variables, x_1 and x_2.

When the numerical sequence of a chromosome is examined alone few information can be obtained concerning the problem to be solved. In order to extract meaningful information concerning a candidate solution, its chromosome should be mapped to the corresponding phenotype. Nevertheless, the search for optimal solutions is based on the genotypes of candidates unless a real-value encoding scheme has been applied to represent chromosomes. When chromosomes are mapped to phenotypes their fitness can be evaluated. This is achieved by implementing a fitness function that expresses the performance (or contribution, or quality) of an individual to the problem's solution. This corresponds to the successful adaptation of an individual to survive to natural environment. Hence, a fitness function determines the prerequisites for selecting pairs of individuals to produce offspring. During the reproduction phase a fitness value is assigned to each individual

whilst this value is a result of a fitness function transformation. Fitness value is biased regarding the corresponding function transformation value of an individual. That is, individuals with higher or lower fitness will be favored for being selected depending on the optimisation task, maximisation or minimisation respectively. Once fitness values have been assigned to individuals, they can be selected for reproduction under a probability proportional to their fitness so that a new generation will be created. Genetic operators directly handle the genes of chromosomes assuming that the genetic code of some specific individuals produces fitted individuals. Crossover operator is applied for exchanging genetic material between pairs of individuals. The simplest crossover operator is the single-point crossover. According to its operational process the genetic material of two chromosomes is exchanged with reference to a given point assigned to both chromosomes. Crossover is not applied to all individuals in a population but only to those been selected under a given probability. Another important genetic operator is mutation and is applied regarding a predetermined probability to new individuals as a result of the reproduction process. Mutation simulates and prompts the genetic change of an individual's binary representation regarding a probability rule. When it comes to binary representation, mutation changes a bit for 0 to 1 and vice versa. Mutation operator is essential from the perspective of ensuring that the probability to explore a subspace of potential solutions is never zero. This technically prevents an algorithm from being converged to a local optimum (minimum or maximum) instead of the global optimum.

After crossover and mutation, the chromosomes are decoded (if necessary) the objective function is computed, the fitness value is assigned and individuals are selected for reproduction again, thus the evolution process continues until the next generation. As an outcome the average performance of individuals constituting a population is expected to be increased since fitted individuals remain and produce offspring whilst less fitted ones degrade. The genetic algorithm terminates the evolution process once the stopping criteria are met, i.e. maximum number of generations has been reached.

Genetic algorithms have significant differences compared to other conservative optimisation / problem solving methods. Most important differences are that GAs examine a population of solutions and not a single solution, they don't need derivative information or any other mathematical "knowledge" except from the objective function and fitness values that guide the GA towards the search direction, they apply probabilistic transition rules and not deterministic ones to end up with optimal results and finally they operate on encoded groups of parameters rather than on their phenotypes unless a real-value encoding scheme has been applied to represent chromosomes. It is important to note that genetic algorithms provide a number of potential solutions referring to a given optimisation problem whilst it is up to the user which one of the solutions available is to be selected.

4.2.1   Representation and initialization of population (candidate solutions)

The most common chromosome representation for candidate solutions is that of binary-encoded stings (Goldberg, 1989). Each variable is given as a binary string and all variables are linked to formulate a chromosome of a finite length. Recent research concerning encoding schemes to represent chromosomes suggests also the employment of several others such as real-value encoding scheme (or floating point – FP), Gray-binary encoding, Permutation encoding, etc. (Kumar, 2013). Encoding scheme selection plays key role to problem solving and optimisation processes when artificial intelligent algorithms are to be employed since it facilitates the functionality of genetic operators on their way to generate, evaluate and recombine individuals towards the convergence of optimal results. Encoding schemes provide the initial seeding and as such they should satisfy the requirements of providing efficient building blocks (patterns that describe subset of chromosomes with similar sections (Forrest and Mitchell, 1993, Holland, 2000)), as well as to follow the principle of minimal alphabets (Holland 1975). According to the principle of minimal alphabets, encoded patterns need to be the smallest possible to increase the possibility of maintaining similar schemes. This is based on the fact that by reducing cardinality of alphabets, increase in potential solutions is achieved, i.e. encoding using the binary range of (0,1) is better than one using all available letters from the alphabet (A,B,C,D………,Y,Z), however, selecting the proper encoding scheme is a problem-dependent task (Kumar, 2013, Jaggi et al. 2013). Binary encoding scheme is quite simple in terms of its employment and satisfies both principles stated above, that of efficient building blocks and that of minimal alphabets. To examine the behaviour of the two most often-implemented encoding schemes, binary and real, in the case of CNC machining problems, Krimpenis and Fountas (2016) performed experiments on a common CNC machining problem using two genetic algorithms, one with binary encoding and one with real-value encoding for chromosome representation. It was found that binary encoding was superior to real-value encoding.

Once the encoding scheme for representing chromosomes has been decided, the creation of the initial population of candidate solutions is the next step. This is usually achieved by using a random number generator capable of proposing uniformly distributed numbers with reference to a given bound. If a binary population is assumed to have *N* individuals whose chromosomes' length is *L* digits then *N x L* randomly distributed numbers around the given bound will be produced.

4.2.2 Objective and fitness functions

Objective function is applied to provide a metric for the performance of individuals in a problem's region. In the case of a minimisation problem best individuals are considered those with the lowest objective function value whereas in the case of a maximisation problem best individuals are those with high objective function values. Fitness function is a numerical representation of the objective function and transforms its value to a non-negative number. Fitness function ought to be more sensitive compared to the objective function in order to be able to detect and distinguish a better candidate solution from a good one when it comes to low differences in magnitudes. This is of great importance since there is a need to decide which partial solutions should be considered over others and thus guide the algorithm to an advantageous search direction in which the entire population should move towards (Chipperfield et al. 1994). Hence if the transformed fitness of an individual is $F(x_i)$ and its initial objective value is $f(x_i)$, then:

$$F(x) = g(f(x))$$
Eq. 4.1

where,

f: the fitness value prior to transformation,

g: transformation of initial objective/fitness value to a non-negative value,

F: the resulting fitness value after applying transformation.

Thereby, the fitness of each individual in a population $F(x_i)$ is defined as the fraction of the initial objective/value $f(x_i)$, over that of the entire population as shown in Equation 4.2.

$$F(x_i) = \frac{f(x_i)}{\sum_{i=1}^{N} f(x_i)}$$
Eq. 4.2

where,

N: the population size,

x_i: the phenotype (arithmetic value) of i_{th} individual.

4.2.3   Parent selection for reproduction

Parent selection is the process of defining the number of cases where a single individual is selected for reproduction and is related to the number of offspring that creates. Selection of individuals may be treated as a two-fold process involving the definition of the number of cases where the individual is selected and their conversion to a specific number of offspring. The former part of selection process deals with the transformation of rough fitness values to an estimation of an individual's probability to be selected and this is previously performed during fitness assignment. The latter part deals with the probabilistic selection (sampling) of individuals for reproduction based on their fitness when compared to others.

Baker (1987) presented three performance metrics for selection algorithms, bias, spread and efficiency. Bias designates the absolute difference between actual and expected probability of selecting an individual. Consequently, the optimal (zero) bias may be achieved when the probability of selecting an individual, equals to its expected number of selection. Spread refers to the bound (upper and lower) of the range of possible number of an individual's selections. If $S(i)$ is the actual number of selections for i_{th} individual, then the minimum range is considered that which theoretically introduces no bias Equation 4.3.

$$S(i) \in \left\{ \left\lfloor Lb_i \right\rfloor, \left\lceil Ub_i \right\rceil \right\}$$

Eq. 4.3

where,

$S(i):$ the expected number of selections for i_{th} individual,

$\left\lfloor Lb_i \right\rfloor:$ the lower bound of i_{th} individual's the selection range,

$\left\lceil Ub_i \right\rceil:$ the upper bound of i_{th} individual's the selection range.

Consequently, while bias is an indication of accuracy, the range of a selection method measures its consistency. The need for efficient selection methods is motivated by the necessity to maintain computation time of a GA within acceptable levels. It has been shown in the literature that the rest of algorithmic operations - except objective function evaluation - require computational time equivalent

to *N x L* or better. The selection algorithm must thus achieve zero bias while maintaining a minimum range and not leading to increased calculation time.

Selection process is based on the law of the "survival of the fittest". This process decides which individuals will have the opportunity to participate in reproduction to bequeath to next generations a part, or all of their characteristics. The main goal of selection process is to account for the exponential growth of elite individuals and thus - after several generations of reproduction - their prevalence. Without selection operator within the reproductive procedure, the genetic algorithm is equivalent to a system that performs a completely scattered search. There are several ways to implement the selection within a GA, yet, given that the size of the population from generation to generation does not change (at least in the basic genetic algorithm), any selection technique should somehow give a greater chance of reproduction to most competent individuals evaluated in the artificial environment. Reproduction may be expressed on an algorithmic basis in many ways from which the easiest and most prevalent of them is that of a forced roulette where each string of a population is represented as a part of the roulette in proportion to its performance (Goldberg, 1989).

To introduce the usage of the forced roulette, a population of four individuals comprising five digits each is created by casting a coin twenty times whilst their performance (objective function value) is evaluated as shown in Table 4.1.

Table 4.1. Example of forced roulette implementation to four 5-digit individuals.

No. string	String	Objective value	Performance (%)
A1	01101	169	14.0
A2	11000	576	50.0
A3	01000	064	05.0
A4	10011	361	31.0
Sum		1170	100.0

The overall summation of the performance of the four candidate solutions (strings) equals to 1170 whereas the percentage of each string in the overall population performance is shown in the last column of Table 4.1. Figure 4.2 illustrates the performance for each of the four individuals in this generation in terms of a percentage.
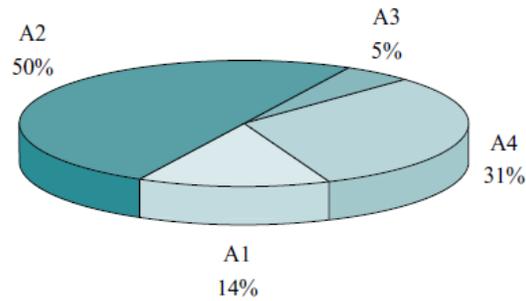
Figure 4.2: Forced roulette selection operation for a generation of four individuals.

Sum is a range of real values and is defined as the sum of objective values in the current population. The performance for each individual is then displayed one by one in a row within the closed range [0, *Sum*]. The size of each interval is proportional to the individual's fitness. In Figure 4.2 Individual A2 exhibits the highest fitness thus it occupies the largest roulette's portion. Similarly individual A3 has the lowest fitness thus it occupies the smallest portion in the roulette. A random number is generated within [0, *Sum*] range and the individual within the space of which the random number falls is selected for reproduction. This process is repeated until the required number of individuals has been selected. Known methods of forced roulette are stochastic sampling with replacement (SSR) and stochastic sampling with partial replacement (SSPR). In stochastic sampling with replacement (SSR) the interval size and probability of selection remain the same throughout the selection phase and individuals are selected according to the aforementioned procedure. SSR has zero bias but probably an infinite range. Any individual with a size greater than zero might completely occupy the next population. Stochastic sampling with partial replacement (SSPR) is based on SSR, yet, the portion of the roulette of a selected individual is updated. Each time an individual is selected, the magnitude corresponding to its portion is reduced by 1.0. If the magnitude of the portion becomes negative, it is set to 0.0. This provides an upper limit for the range. However, the lower limit is zero and the bias is higher than that of SSR.

Other selection schemes are steady-state and elitism. A conventional genetic algorithm (Cobb and Grefenstette, 1993) will create offspring from an old population's individuals by applying the genetic operators and offspring will be placed in a new population which in turn becomes the old one after the entire new population is created (Goldberg, 1989). Now, in an incremental (or steady-state) GA, the steady-state selection operation suggests that a single or few elite individuals are to be selected for creating new offspring whilst those with low fitness are to be removed from the population with offspring to take their place and the rest of population survives to next generation. This operation applies a replacement strategy from two available (deleting the oldest, deleting the weakest), to determine which individuals will be removed from the population (Whitley and Kauth, 1988).

Outstanding candidate solutions may be lost if crossover and mutation operations produce offspring weaker that their selected parents. There is a chance for a GA to discover again those individuals in subsequent populations of next generations, yet, with no guarantee. To ensure that such solutions will be preserved to next generation elitism has been proposed and implemented as a parent selection scheme to GAs (Yang, 2007, Chudasama et al., 2001, Romero-Hdz et al., 2016). Elitism copies small parts of fittest individuals and preserves them to next generation whilst it saves significant computational time referring to the effort needed for retrieving previously discarded partial solutions. Thereby eligible elite solutions may be selected as parents for producing offspring. Elitism may target to the preservation of a unique individual or few outstanding individuals as copies of best solutions to be included to the next generation.

4.2.3 Crossover

As a result of selection operation, the temporary population should pass through a mating procedure in order to produce offspring as it happens in nature. Selected individuals will be mated as pairs of two to produce new individuals (offspring). Despite that the characteristics of selected individuals may significantly affect the algorithm's convergence speed, mating is randomly performed. In each pair of two individuals a simple exchange of genetic material is done known as crossover. In a GA-EA crossover facilitates the exploration of new solution regions since it corresponds to the exchange of substrings between two individuals. Owing to its high contribution to optimisation process, several types of crossover operators is available to implement. However their suitability depends on the problem's properties. Most known crossover types are single-point crossover, multi-point crossover and uniform crossover.

Single-point crossover (or one-point crossover), a crossover point is randomly determined and the tails of the two selected individuals are swapped to give offspring with regard to the crossover point as shown in Figure 4.3 (a). Multi-point crossover generalizes single-point crossover where alternating portions are swapped to give offspring in (Figure 4.3 (b). In uniform crossover shown in Figure 4.3 (c), chromosomes of mated individuals are not divided to portions but a mask that assigns which chromosome locations referring to both parents will be recombined, is applied.

(a)

(b)

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

| 5 | 8 | 9 | 4 | 2 | 3 | 5 | 7 | 5 | 8 |

=>

| 0 | 1 | 2 | 4 | 2 | 3 | 6 | 7 | 8 | 9 |

| 5 | 8 | 9 | 3 | 4 | 5 | 5 | 7 | 5 | 8 |

(c)

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

| 5 | 8 | 9 | 4 | 2 | 3 | 5 | 7 | 5 | 8 |

=>

| 5 | 1 | 9 | 4 | 4 | 5 | 5 | 7 | 5 | 9 |

| 0 | 8 | 2 | 3 | 2 | 3 | 6 | 7 | 8 | 8 |
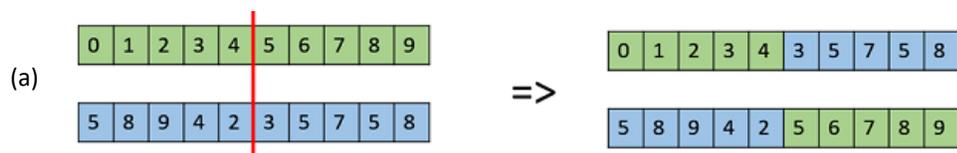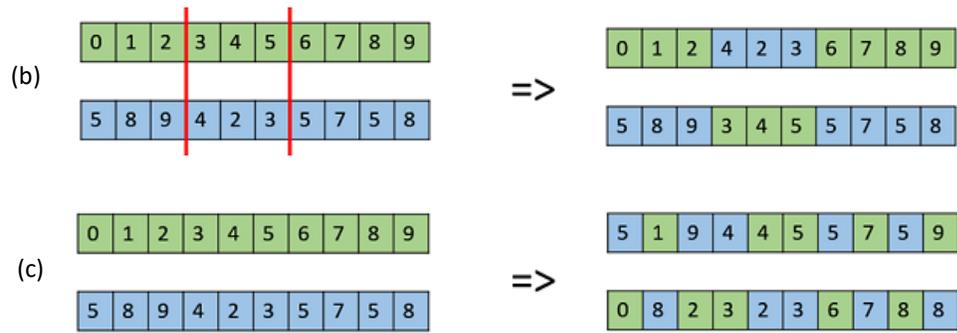
Figure 4.3: Crossover operator: (a) single-point crossover, (b) multi-point crossover, (c) uniform crossover.

Crossover operation redirects the search towards new local regions of the solution space and thus it facilitates the GA-EA to increase its chance of success. However the possibility of producing offspring worst than parents always exists, yet, the probability of multiplication for offspring reduces in next reproduction cycles owing to their poor performance and quality. Crossover is applied to GAs-EAs by using a certain probability. This probability depends on the optimisation problem whilst it is possible to dynamically change its magnitude during the algorithm's execution time. Crossover probability affects both execution time of an algorithm and its convergence speed. The larger the probability for crossover (i.e. p_c = 1.0) is, the smaller the search step is determined for the algorithm. This results to a beneficial search towards global optimum but under slow convergence speed. On the contrary if small values for crossover probability are determined fast convergence speed is a favored using large search step with the risk to bypass the region containing global optimum solution.

Each type of crossover comes with its own advantages and drawbacks, thus, selecting one is ultimately based on optimisation requirements and the properties of a GA-EA variant. Despite the sophisticated schemes of multi-point and uniform crossover they do not always attain optimal results when compared to those obtained using the single-point crossover (Mendes, 2013). Single-point crossover involves only one point where genetic exchange is determined for a pair of chromosomes thus it is less likely to damage building blocks (Goldberg, 1989). In contrast, multi-point crossover and uniform crossover impose multiple points where genetic exchange occurs, violating thus the requirement of maintaining subsets of chromosomes with similar sections.

4.2.4 Mutation

Mutation is a function that improves natural organisms and favors the evolutionary process. Like its role in natural environment, mutation introduces positive impact by preventing premature convergence of GAs-EAs, exploring new solution regions, as well as maintaining diversity in new populations of candidates. As such it is directly related to the "exploration" of the problem's solution

domain. Mutation suggests a digit's reversion (0 to 1 and vice versa) under a given probability (or percentage) p_m as substrings of chromosomes are copied from parent to offspring. It is essential that mutation probability p_m is maintained small enough, otherwise the algorithm will degenerate into an entirely scattered search (Goldberg, 1989). Mutation operates as a "safety precaution" mechanism in cases where selection and crossover operators gradually lose efficient genetic information. As a result, mutation redirects the search and ensures that no point in the solution domain will be excluded from the search process (Goldberg, 1989).

Known mutation schemes are single-point mutation (or bit flip) as described above, swap mutation, scramble mutation and inversion mutation, each of which is problem-dependent for its employment as it occurs to the rest of genetic operators previously presented. Bit flip mutation Figure 4.4 (a) is preferable when it comes to binary encoding for chromosomes. Swap mutation randomly selects two discrete positions in a chromosome and interchanges genes accordingly in Figure 4.4(b). Scramble mutation utilizes a subset of genes whilst their values are either scrambled or shuffled randomly and is preferable when it comes to permutation encoding representations for chromosomes in Figure 4.4 (c). Inversion mutation is similar to scramble mutation but instead of scrambling or shuffling the subset of genes a mere inversion of the entire string is performed in Figure 4.4(d).
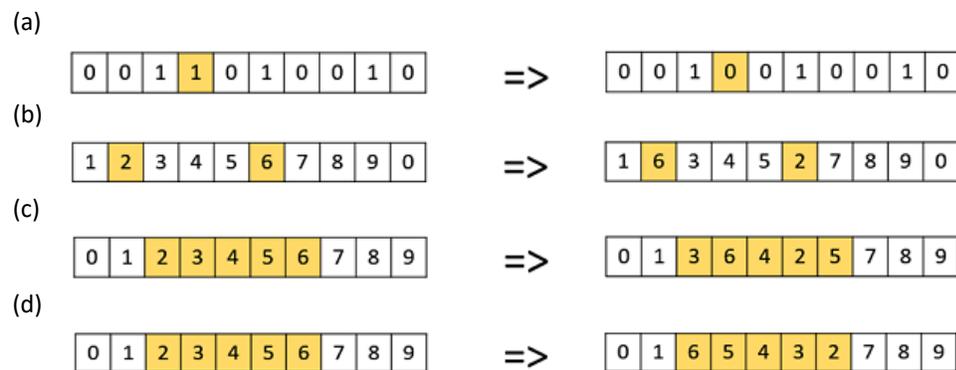
Figure 4.4: Mutation operator: (a) single-point mutation (bit flip), (b) swap mutation, (c) scramble mutation, (d) inversion mutation.

4.2.5 Reinsertion

Once a new population form selection and crossover of old individuals has been created the fitness value of offspring can be determined. If fewer individuals have been generated regarding the previous population's size then the fractional difference between the size of new population and that of the old one is known as generation gap (De Jong and Sarma, 1993). In order to avoid generation gap (which technically means reduction of information) and maintain the original population size,

new individuals should be reinserted to the old population. Similarly, if not all new individuals are to be used in the next generation or if more offspring are created than old individuals then a reinsertion method should be applied so as to determine which individuals will constitute the new population. The only advantage gained from the fact that no more offspring than individuals of the current population are created, is that computational time in each generation is gradually reduced especially in the steady-state GAs. The same goes also for the memory usage since fewer individuals are produced and eventually stored. Nevertheless these are of minor importance when compared to the preservation of efficient solutions (and probably global optimum) until convergence. As it has been reported in selection operation, two methods, -deleting the oldest and deleting the weakest- are available for determining the individuals that will be removed from the population. Regardless of the method to apply for maintaining the population size, individuals should hold adequate information in order to survive to subsequent generations until the GA-EA is terminated.

4.2.6   Termination

Since genetic algorithms are stochastic searching modules it is rather difficult to formulate one or more stopping criteria. Fitness values may not change in terms of their magnitudes for a given number of generations before an outstanding candidate solution is found, thus, the implementation of conservative stopping criteria becomes problematic. A common practice for a genetic algorithm is to stop its workflow after a predetermined number of generations have been evaluated and further examine the quality of best solutions with regard to the problem at hand. If no solution satisfies the requirements the genetic algorithm may either continue to further evaluate some generations or start from the beginning to conduct a new search for the global optimal solution.

4.3   Optimisation methodology description

The major goal of the methodology developed in this PhD thesis is to formulate globally optimal tool paths for the machining of parts comprising sculptured surfaces regarding their important machining parameters which are cutting tool, stepover, lead angle, tilt angle and maximum discretisation step. To achieve this goal, the methodology adopts specific concepts, the validity of which has been supported by the state of the art in available literature as well as modern approaches for industrial practices. Such concepts are bulleted below:

- Tool path parameters do not maintain the same impact/effect on quality objectives such as machining error, machining error uniformity and number of successive tool positions when it comes to different sculptured surfaces (as shown in Chapter 3).

- Product requirements in terms of precision machining, part quality and accuracy are to be satisfied during the finishing stage.

- Even though machining error and its corresponding uniformity are low in absolute magnitudes during the finishing stage, they can yield dramatic tool axis variations which in turn may affect the overall cutting tool trajectory and tool path smoothness.

- Subsequent tool axis variations in the case of 5-axis sculptured surface CNC machining are a common occurrence mainly due to the complexity of sculptured surfaces and different local curvatures.

According to the concepts stated above, variations of cutting tool trajectory towards feed direction are more likely to occur in those surface regions where local curvatures result to abrupt changes of cutting tool orientations and consequently effective cutting postures. In other words, those surface regions will be responsible for dramatically varying the overall cutting tool trajectory since they impose profound changes in cutting tool orientations either as absolute magnitudes or as a frequent phenomenon affecting either way resulting precision and surface quality of products. From a functional perspective the methodology's philosophy on its design and development has been such that:

- It can provide a generic environment for globally optimizing the sculptured surface machining problem and maintain quality in its generated outputs in any case of sculptured surface,

- It can provide flexibility to easily modify its objectives/components to adapt to specific applications and / or requirements,

- It can be practically viable, i.e., to consider the practices, conditions and systems currently implemented in production environments and be implemented requiring the least possible resources,

- It can ensure compatibility to cooperate with existing manufacturing systems and offer new capabilities or even extend/automate currently available ones using a familiar operational interface,

- It can incorporate the user's experience based on the principles of manufacturing, without preventing the user from making critical decisions.

From a macroscopic viewpoint, the methodology consists of two parts as it has been already mentioned: the first part is responsible to automate repetitive tasks concerning the process planning in terms of tool path generation as well as to evaluate the criteria set for optimisation. The second part constitutes the optimisation module which in turn involves several functions to achieve artificial evolution of candidate solutions to eventually achieve several globally optimal solutions for the

sculptured surface CNC machining problem. The selection of methods and approaches to establish the overall optimisation methodology has been based on widely accepted conclusions from academic and industrial perspectives, most of which have been already (or are about to be) presented and discussed in this thesis. The analytical description of the methodology is given in the following sections of the chapter.

4.3.1 Part I: CAM software automation function and criteria evaluation

The evaluation of criteria involved to the optimisation problem is achieved through an integrated programming application in which demanding functions from the perspectives of computational time and complexity of CAM software tasks justify its development. As it is the case of any modelling approach, this work has examined crucial tool path parameters for 5-axis surface machining to study their effects on the optimisation criteria set and represent the problem at hand. Further on it is mandatory to automate tool path parameters and any of their associated utilities in order to manage the feasible workflow of the entire optimisation methodology as well as the reduction of the computational time required to execute it. Note that tool path parameters should be involved to the overall optimisation process since they directly affect the final CNC program formulation whilst they constitute the only attributes for planning tool paths to machine sculptured surfaces. Even though the programming application has been developed to support the optimisation criteria presented in Chapter 3 the possibility of including other criteria or replacing the ones introduced with others like part production cost, working shift cost, etc., which are not as strictly related to the problem as those introduced. The integrated programming application (automation function) has been developed in *Microsoft® Visual Basic® for Applications* environment and has taken advantage of the "open" programming architecture (API) of *Dassault Systèmes CATIA® V5 R18.* The application automates the overall management of the aforementioned CAD/CAM system in order to extract appropriate data for the evaluation of 5-axis sculptured surface machining tool paths. The same application provides also a feedback to the optimisation algorithm concerning the results from the criteria computed, thus, playing the role of the objective function. In a rough description the application scans the project tree in a machining setup document active in the CAM interface. Once the cutting strategy containing the 5-axis surface machining tool path has been found, it is retrieved to access its parameters. Thereby, a candidate solution in its phenotypic form (real values for tool path parameters) occupies the "argument-passing" fields corresponding to each of the tool path's parameters. The tool path is then automatically computed to produce the associated CL file (APT source file) which is accessed to

import tool positions in the CAD environment containing the designed sculptured surface. With reference to the designed sculptured surface, the cutting strategy/tool path and tool orientations the criteria of machining error, machining error uniformity and number of cutting points are evaluated to obtain the objective values. These operations are repeated for all populations of candidate solutions and subsequent generations that the optimisation algorithm handles. The overall workflow of the programming application for CAM software automation and criteria evaluation is depicted in Figure 4.5.
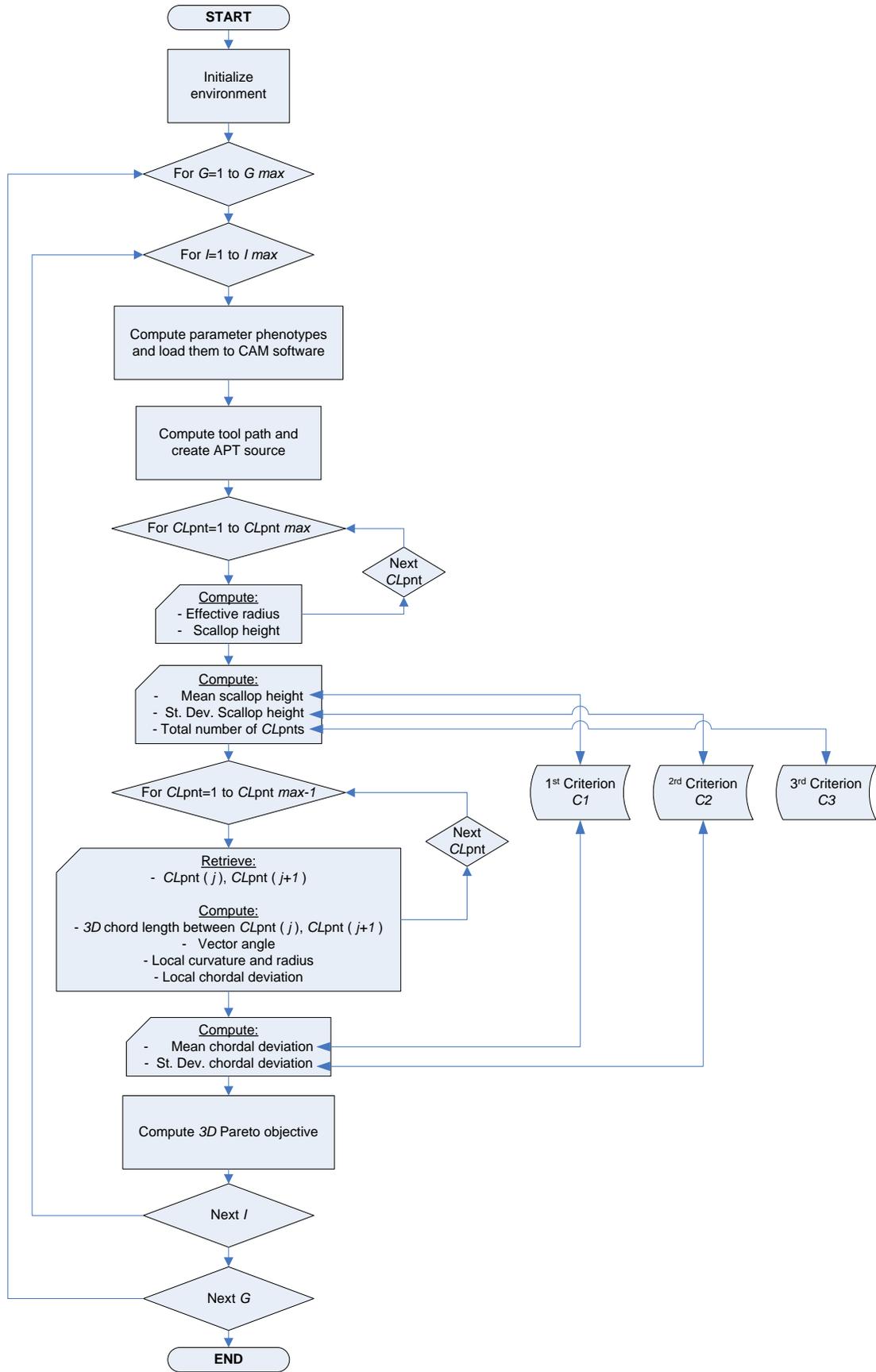
Figure 4.5: Overall workflow of the programming application (automation function) developed for automating CAM software functions and evaluating the optimisation criteria.

The programming application operates as an external public function which utilizes the necessary programmable references of *Dassault Systèmes CATIA® V5 R18* that deal with CAD and CAM automation utilities. It also involves three independent "public" functions called to operation according the application's workflow for managing the required automation elements to finally compute the optimisation criteria. The main core of the application declares the necessary variables as well as the objects to access and handle CAM software properties. In the application, the five tool path parameters which are cutting tool, stepover, lead angle, tilt angle and maximum discretisation step take values suggested by the optimisation algorithm via argument passing, i.e. "stepover.value = phenotype(i,2)." This means that the phenotypic value to be assigned for the second tool path parameter - which is stepover-, will be the binary-encoded gene in the chromosome of i_{th} individual after its mapping to the corresponding phenotype (real number). It is reminded at this point that during machining simulation experiments presented in Chapter 3 tool path parameters were conservatively fed with values for the automated computation of optimisation criteria without the necessity of this argument-passing technique. Based on this philosophy the phenotypic values provided by the optimisation algorithm are assigned to tool path parameters according to their order in a chromosome (candidate solution).

The first tool path parameter refers to the cutting tool which may be any tool, flat end-mill, ball end-mill or filleted end-mill in the case of 5-axis sculptured surface CNC machining. A cutting tool database in the form of "IF-ELSEIF-END" has been properly coded to change cutting tools' configurations according the case whilst the phenotypic value for cutting tool parameter is always an integer. The size of tool database in terms of its number of items can be as large as required, however, testing a large group of cutting tools adds to computational time since the number of experimental scenarios increases accordingly. Each cutting tool in the database is associated to its main geometrical configurations involving nominal (cutting) diameter, body diameter, corner radius, cutting length and overall length. In addition, applicable feeds and speeds are also associated to each of the cutting tools included to the database. Thus, if for example the optimisation algorithm suggests for evaluation the 3rd cutting tool in the database, the dedicated argument-passing field for the cutting tool which is "SelectedCuttingTool=phenotype(i,1)" takes number 3 as the phenotypic value (the tool's index) and based on the configurations of that tool, the entire cutting tool's 3D model and its feeds/speeds are automatically updated in CAM software.

The second tool path parameter refers to stepover and may be determined by setting a direct distance value, or by setting an overlap distance between consecutive passes, or by giving a percentage ratio in terms of the selected cutting tool's nominal diameter. The third determination

has been preferred against the first and the second stepover computation settings since it provides safety in terms of the final calculated distance with regard to cutting tool's diameter. Thus, if a cutting tool has been selected, based on its index from the database mentioned, i.e. tool no.3, and that tool has nominal diameter D_n=8 mm then a phenotypic value for stepover, i.e. "60" given to its associated argument-passing field (i.e. "SelectedStepOver=phenotype(i,2)") would render an actual stepover distance equal to 4.8 mm. Similarly for another cutting tool with D_n=16 mm the value of "60" would render an actual stepover distance equal to 9.6 mm. Lead angle, tilt angle and maximum discretisation step parameters take their phenotypic values suggested by the optimisation algorithm as arguments in their associated programming fields, "LeadAngle.Value=phenotype(i,3)", "TiltAngle.Value=phenotype(i,4)" and "MaxDstep.Value=phenotype(i,5)".

Once all "argument-passing" fields of tool path parameters are fed with suggested phenotypic values the tool path is automatically computed using the programming object available to *Dassault Systèmes CATIA® V5 R18* open API property, "GetTrajectoryEndPointCoord(EndPoint)". This property is used to retrieve the coordinates of the last cutting point in a tool path but as an advantageous side-effect, it computes the entire tool path to do it so. Further on the application executes the three public-declared functions as routines embedded to the main application's function. These functions are GenAPT, ToolPositionsXYZIJK and ComputeObjectives for generating the APT source code, retrieving the cutting tool's positions and computing the optimisation criteria respectively.

A prerequisite for GenAPT to be executed is to first compute the tool path with regard to the values for tool path parameters as suggested by the optimisation algorithm in the form of phenotypic values. Thereby the function takes into account the number of setups in the active process planning document as well as the number of manufacturing programs. In the case of a single machining setup with a single manufacturing program the function directly extracts CL data associate to that program based on the tool path. The function practically calls the post-processor engine that can be programmatically deployed using "ManufacturingAPTGenerator" API object. This object provides two properties, "InitFileGenerator" and "RunFileGenerator" so as to initialize the post-processing engine for the computed tool path and eventually extract the data required for building a complete NC program. At this point the post-processing engine does not account for a specific type of a CNC controller in order to translate APT commands (CL data) to ISO code (or G-code) since the task is to retrieve geometrical information concerning the tool positions rather than execute a complete ISO code using a typical 5-axis CNC machine tool. The workflow of public function GenAPT to produce the APT source file for further activities related to the overall process of computing the optimisation criteria is illustrated in Figure 4.6.
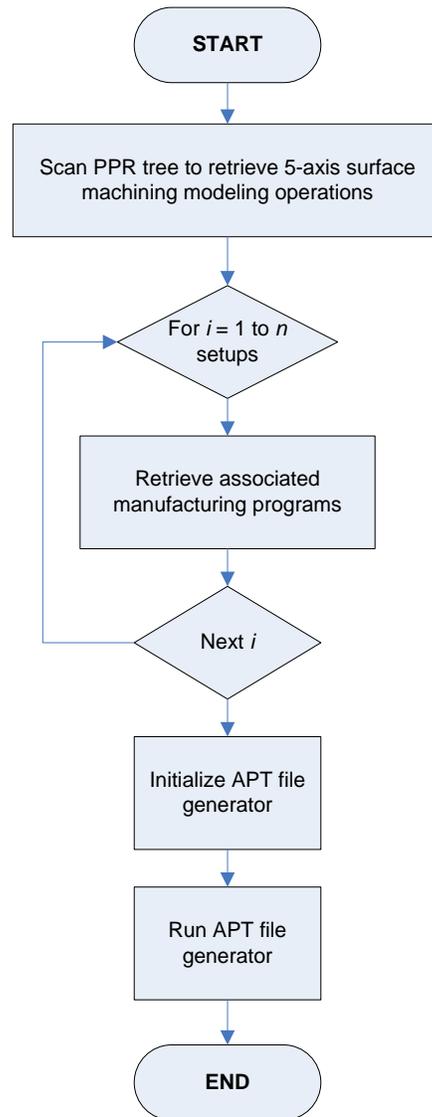
Figure 4.6: Workflow of GenAPT function developed for the automatic generation of APT source files in relation to computed tool paths.

The second public-declared function "ToolPositionsXYZIJK" undertakes to scan the APT source file (CL data) generated using the previous public-declared function and keep all tool positions with regard to their coordinates (X, Y, Z) and cutting tool orientations (I, J, K). The APT source file is accessed and a FOR-NEXT loop is assigned to sequentially read each block of the APT source file so as to track the tool position. The blocks of APT source file will include also the APT commands for miscellaneous and preparatory functions (M and G codes) as well as motion commands, tool change commands, spindle rotation, tool inclination mode for 5-axis machining, etc. The workflow of "ToolPositionsXYZIJK" public-declared function is shown in Figure 4.7.
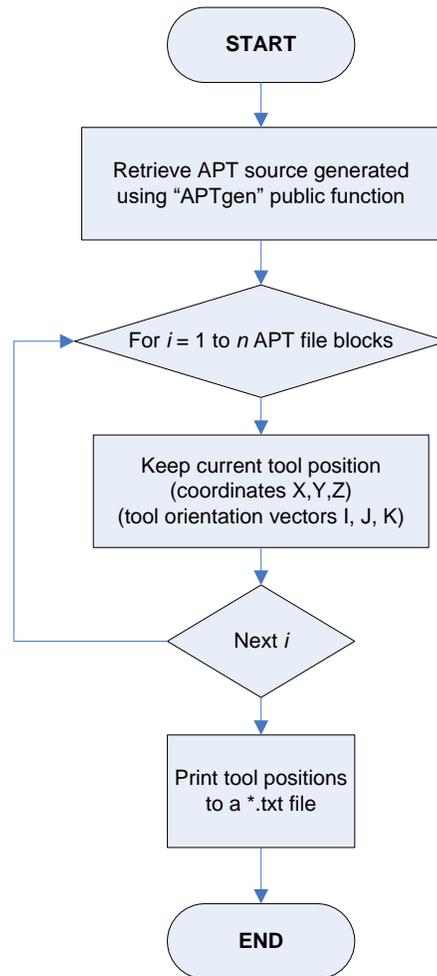
Figure 4.7: Workflow of ToolPositionsXYZIJK function developed for the automatic retrieval of tool positions.

The third public-declared function "ComputeObjectives" handles further the coordinates and tool vectors stored from the previous public-declared function to finally compute the optimisation criteria. The function initializes the input of meaningful tool positions to CAD environment with respect to the original model and its reference axis system if the latter is also the reference coordinate system in the machining setup document. As it has been reported each tool position is described by its coordinates (X, Y, Z) and cutting tool orientations (I, J, K). In order to transform each tool position to a dimensional entity to be imported to CAD environment the automation object "HybridShapeFactory" has been deployed along with its corresponding property of creating points from coordinates, "AddNewPointCoord". Vector components (I, J, K) have been taken into account to calculate inclination (rotary axis lead and tilt) angles for 5-axis CNC machining. When it comes to inclination angles X, Y and Z refer to the linear axes whilst A, B and C refer to rotary axis angles according to the machine tool configuration. While this is not of major importance for computing crucial geometrical

entities such as local distances and machining errors, it is mandatory when it comes to the final post-processing to generate the appropriate CNC code according to the type and kinematics of CNC machining center to be used for actual cutting. However to ensure consistency during the validation of the methodology presented in this thesis, the formulas referring to a 5-axis CNC machine tool with a profiling (tilting) spindle head were adopted to compute tool positions (rotary axes angles from tool vector components) in Cartesian space and with regard to the machining reference system. In such a 5-axis CNC machine tool configuration, *A* axis is the primary mechanical rotation axis whilst *B* (or *C*) is the secondary mechanical rotation axis. This implies that the secondary angle (tilt angle a_T) is computer after primary angle (lead angle a_L) since the secondary axis' positioning depends on the primary axis' orientation. For the case of 5-axis CNC machine tool configuration with a profiling (tilting) spindle head the formula to compute the inclination angles is given in Equation 4.4. For the case of dual-rotary machine tool tables and trunnions configurations an inverse mathematical relation in terms of computing inclination angles is adopted since a rotary machine tool table rotates the part and not the cutting tool as it is in a tilting spindle head 5-axis CNC configuration. Warkentin et al. (2001) report the classification of several types of 5-axis CNC machine tool configurations as well as their mathematical properties to compute tool orientations.

$$\alpha_L(\deg) = \left(-Atn\left(\frac{J}{K}\right) \right)$$

$$a_T(\deg) = Atn\left(\left(\frac{I}{K}\right) \times \cos\left(a_L\right) \right)$$

Eq. 4.4

The function continues with the sequential computation of effective radii in each cutting point based on the selected cutting tool type and is associated geometry. Having computed effective radii, local scallop heights are then computed. The computations for each successive cutting point imported to CAD environment are executed using a "FOR-NEXT" loop. Based on the overall number of cutting points examined to compute effective radii and local scallop heights the third optimisation criterion (which is the number of cutting points itself) is straightforwardly evaluated and stored. A part of the first optimisation criterion which is the mean machining error is also examined from the summation of local scallop heights and consequently the mean scallop height for the entire tool path. With regard to these attributes the standard deviation of scallop height which is a part of the second optimisation criterion (machining error distribution via its standard deviation) is also computed. At this point another "FOR-NEXT" loop is executed by the function in order to examine cutting points as pairs and calculate their associated local 3D distances (chord lengths), local curvatures and finally local chordal deviations (chord errors). Further on, mean chordal deviation which is a part of the first

optimisation criterion (mean machining error) is computed with respect to the sum of local chordal deviations whilst standard deviation of this magnitude is also computed and constitutes a part of the second optimisation criterion. By computing these instances all three optimisation criteria are fully determined as well as the Pareto criterion for the multi-objective optimisation. The mathematical relations to compute the aforementioned magnitudes have been presented and reported in Chapter 3. The workflow of "ComputeObjectives" public-declared function is shown in Figure 4.8.
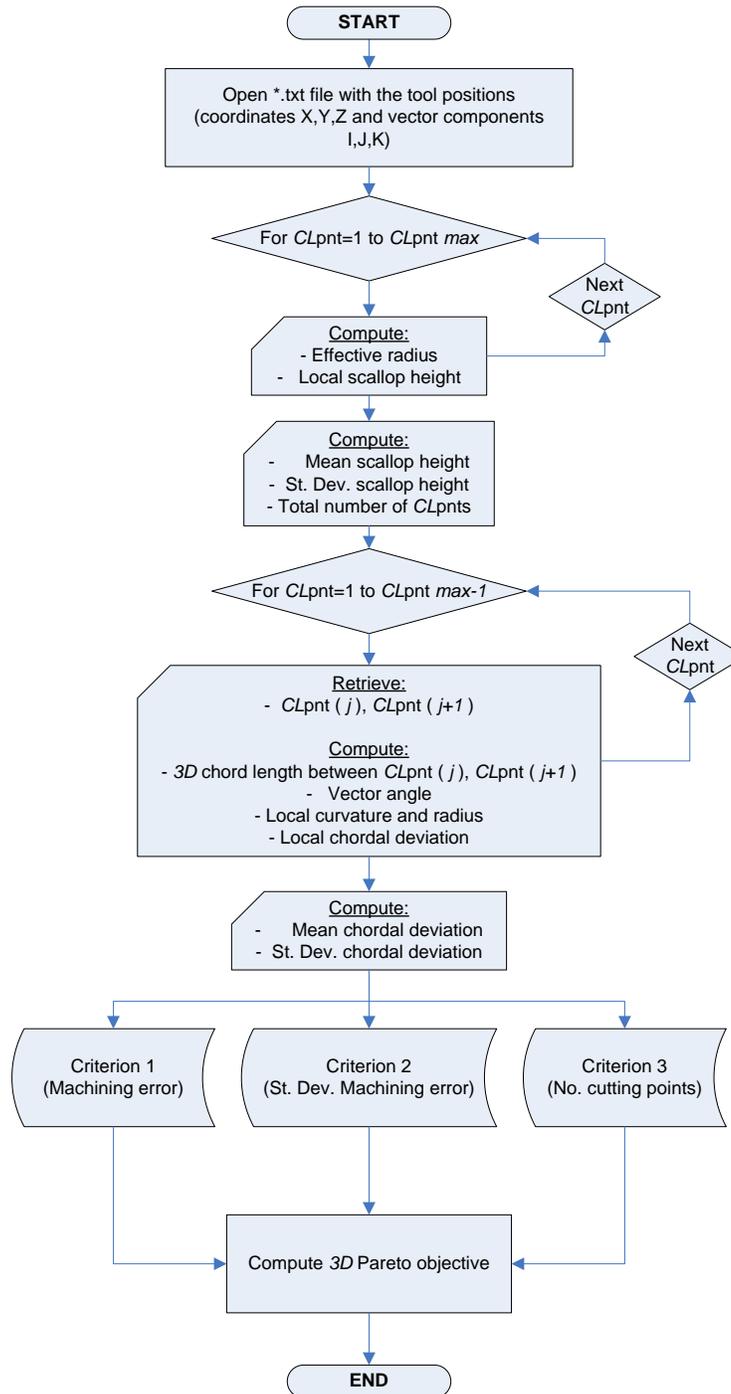
Figure 4.8: Workflow of ComputeObjectives function developed for the automatic retrieval of tool positions.

4.3.2   Part II: Multi-objective virus-evolutionary genetic algorithm (MOVEGA)

The second part of the optimisation methodology presented in this thesis develops and deploys a multi-objective virus-evolutionary genetic algorithm (MOVEGA) to search for globally optimal results in terms of the 5-axis tool path parameters investigated and corresponding optimisation criteria that formulate the problem. By incorporating the CAM software automation function reported in the previous section to MOVEGA, it is possible to evaluate each tool path chromosome (candidate solution) through objective and fitness functions. The ultimate goal is to attain a minimum machining error as uniform as possible (minimize standard deviation for low error distribution) for tool paths with the lowest number of cutting points at the same time.

As it has been mentioned, genetic algorithms are based on the concepts of natural selection and survival of the fittest individuals according to Darwin's evolution theory. The major task of natural selection is to create better offspring regarding the characteristics of their ancestors. With the progress of molecular biology several evolutionary theories other than Darwin's natural selection have been proposed. Therefore, computer science benefits from such new evolutionary theories to realize their key mechanisms and develop intelligent heuristics so as to facilitate engineering problem solving under the essential perspective of optimisation. The most important aspect that draws the interest of researchers worldwide to propose and create new intelligent heuristics or enhance already existing ones is the problem of premature convergence or local stagnation/trapping. The reason why genetic algorithms are prone to premature convergence is that proportional selection for mating individuals may increase not only efficient schemata but also inefficient ones whilst, increasing robust schemata is a fundamental research objective for building reliable evolutionary algorithms to improve searching abilities when it comes to engineering optimisation.

As pure stochastic search systems, evolutionary algorithms are inevitably based on the concept of natural selection inheriting thus the benefits but also the drawbacks characterizing it. Fortunately, evolutionary theories such as the virus theory of evolution (Anderson 1970) suggest that natural selection may not be always responsible for the evolution of species. The virus theory of evolution lies thoroughly on the concept suggesting that *viral transduction* is a major mechanism for transferring DNA segments across species (Anderson 1970). Viral transduction represents the mechanism of the genetic modification that occurs to bacteria by genomes taken from other bacteria through a bacteriophage. Most viruses can cross species' bounds whilst they can straightforwardly be transmitted from phylum to phylum among individuals. This means that viruses can pass over their genome to a population as *horizontal propagation*. In addition, a viral genome may exist in germ cells, thus, it can be transferred from generation to generation as *vertical inheritance*. The term "viral

intelligence" has been given by Fountas et al. (2017a, 2018) for the first time in the related literature and has been based on the fact that viral individuals might as well act as intelligent, sophisticated information carriers ("hill climbers") capable of providing the necessary local information to formulate optimal tool paths for sculptured surface machining since tool paths are represented in the form of binary chromosomes (see chapter 3). The following subsections present the features comprising the infrastructure of the multi-objective virus-evolutionary genetic algorithm (MOVEGA) for addressing the generalized sculptured surface CNC machining optimisation problem as it has been formulated in this PhD thesis. The MOVEGA incorporates the following functions:

- Initialization of candidate solutions (tool path chromosomes)
- Objective function computation
- Ranking function
- Fitness function computation
- Selection function
- Crossover function
- Mutation function
- Viral infection function

4.3.2.1 Initialization of candidate solutions (tool path chromosomes)

Initialization process involves the generation of randomly formulated tool path chromosomes represented in binary encoding. The process of creating randomly formulated tool path chromosomes is achieved through the usage of a random generator of numbers uniformly distributed to the applicable ranges of tool path parameters based on the user's inputs (upper and lower levels). Initialization process is to be performed if no previous evaluation has been preceded. In the case where an evaluation process has been previously conducted, initialization process is bypassed and the methodology considers the last best population of candidate solutions as it has been emerged from the previous optimisation process. Binary-encoded tool path chromosomes are mapped to their phenotypes to obtain the real values associated to their parameters. Thereby the phenotypes of tool paths consist of five numbers, each, corresponding to a single tool path parameter. Cutting tool type which is the first tool path parameter is of an integer form whereas the rest of parameters, stepover, lead angle, tilt angle and maximum discretisation step are of double form, i.e. decimal values are allowed. The attributes associated to the initialization process are stored to *.dat files. Thus, "population.dat" file is assigned to store the binary-encoded population of tool path chromosomes,

"phenotype.dat" file is assigned to store the phenotypes of the binary-encoded population of tool path chromosomes and a general file that serves as a repository file namely "variablelog.dat" is assigned to store the values for parameters accompanied to their associated values for optimisation criteria. The role of the last *.dat file mentioned ("variablelog.dat") is crucial to the overall evaluation process for candidate solutions since it accounts for the prevention of evaluating identical candidate solutions by the intelligent algorithm. This means that the algorithm retrieves "variablelog.dat" file and if an identical evaluation with reference to the values proposed for tool path parameters is found the candidate solution is bypassed and the next is evaluated to save computational time. It should be mentioned that the algorithm's activity for reading the file and identifying whether a set of tool path parameter values (candidate solution) retrieved from "variablelog.dat" has already been evaluated, requires much less computational time than that required to load them to the machining strategy for the tool path and execute the function for computing the optimisation criteria. The workflow of initialization process is illustrated in Figure 4.9.
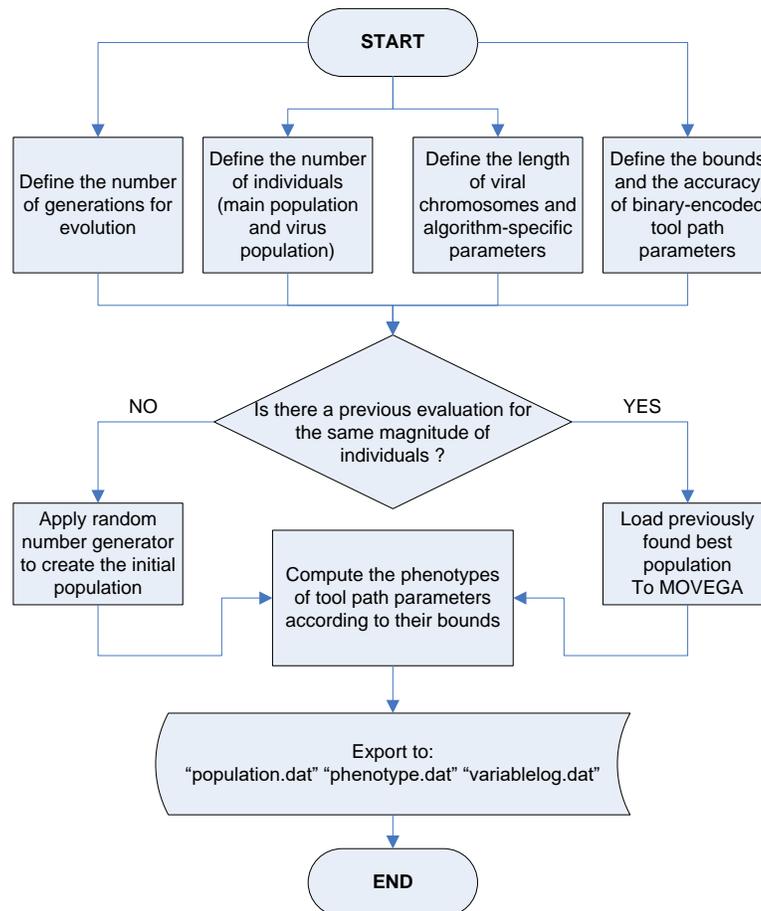
Figure 4.9: Workflow of MOVEGA's initialization process.

Initialization process imposes the need of maintaining data structures using series of arrays for binary-encoded tool path chromosomes, their number of accuracy bits and their locations in chromosome strings, as illustrated in Equations 4.5, 4.6 and 4.7 respectively.

$$C_{Pop}^{init} = \begin{bmatrix} \left[Tlp\Pr m_{1,1} \right]_{N_b^1} & \left[Tlp\Pr m_{1,2} \right]_{N_b^2} & \left[Tlp\Pr m_{1,3} \right]_{N_b^3} & \left[Tlp\Pr m_{1,4} \right]_{N_b^4} & \left[Tlp\Pr m_{1,5} \right]_{N_b^5} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \left[Tlp\Pr m_{n,1} \right]_{N_b^1} & \left[Tlp\Pr m_{n,2} \right]_{N_b^2} & \left[Tlp\Pr m_{n,3} \right]_{N_b^3} & \left[Tlp\Pr m_{n,4} \right]_{N_b^4} & \left[Tlp\Pr m_{n,5} \right]_{N_b^5} \end{bmatrix}$$

Eq. 4.5

$$N_b^{Tlp\Pr m_{i,j}} = \begin{bmatrix} \left[N_b^1 \right]_{1,j} & \left[N_b^2 \right]_{1,j} & \left[N_b^3 \right]_{1,j} & \left[N_b^4 \right]_{1,j} & \left[N_b^5 \right]_{1,j} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \left[N_b^1 \right]_{5,j} & \left[N_b^2 \right]_{5,j} & \left[N_b^3 \right]_{5,j} & \left[N_b^4 \right]_{5,j} & \left[N_b^5 \right]_{5,j} \end{bmatrix}$$

Eq. 4.6

$$C_{Pop}^{init} = \begin{bmatrix} \left[Lgth_i \right] & u_i & l_i \\ \dots & \dots & \dots \\ \left[Lgth_n \right] & u_i & l_i \end{bmatrix}$$

Eq. 4.7

In Equations 4.5, 4.6 and 4.7, C_{Pop}^{init} is the initial population of tool path chromosomes, $Tlp\Pr m$ a tool path's parameter, $N_b^{Tlp\Pr m}{}_{i,j}$, the number of bits required for the accuracy of each tool path parameter $Tlp\Pr m$ for the i_{th} tool path chromosome of the j_{th} population and $Lgth_i$ is the i_{th} tool path's chromosome length. Hence, considering a given tool path parameter $Tlp\Pr m_i$, its corresponding domain $D_{Tlp\Pr m_i} = [Ub_i, Lb_i]$ and the i_{th} tool path's chromosome length $Lgth_i$ the following expression has been used for converting binary-encoded values to real-encoded ones (Chipperfield et al. 1994).

$$TlpPrm_i = Lb_i + fnc(BinStr) \times \frac{Ub_i - Lb_i}{2^{Lgth_i} - 1}$$

Eq. 4.8

where, Ub_i, Lb_i are the parameter's upper limit and lower limits, whereas $fnc(BinStr)$ is a function developed to return the decimal values for binary-encoded schemes depending on the accuracy requirements (Holland 1975, Holland 2000). The decision whether a population of candidate solutions exists as a result of a previous optimisation process or not has been coded using "flag" statements

commonly available to *Microsoft® Visual Basic® for Applications* environment. Thus, if a population of solutions already exists and is stored to "population.dat" file, a coded flag statement "popFlag = 1" is valid and suggests that the algorithm is to be seeded with the last best population, otherwise, the initialization process is normally executed with the random number generator to produce a new population (flag statement "popFlag = 0").

4.3.2.2 Evolution

The main function for the optimisation process execution (evolution) is depicted in Figure 4.10.
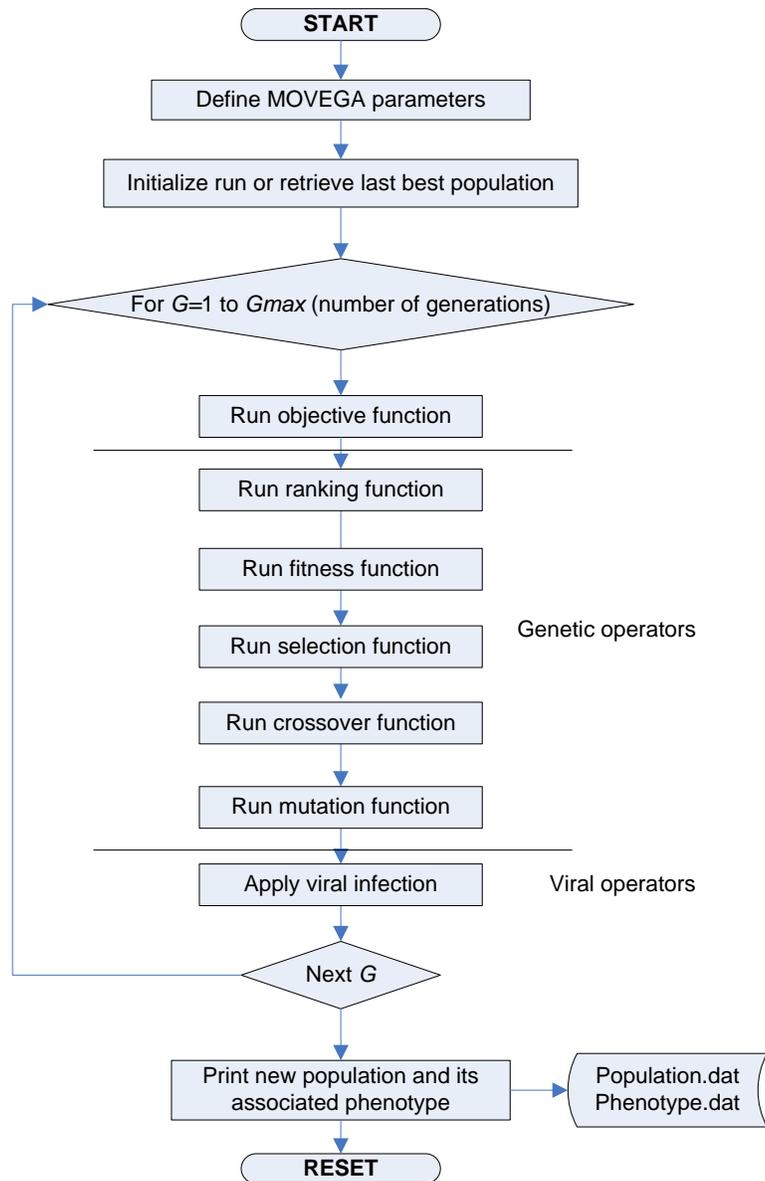
Figure 4.10: Workflow of MOVEGA's evolution process.

The first step in the process is to determine meaningful values for the algorithm-specific parameters, i.e., number of generations, number of individuals in a population, number of viruses in a virus population, etc. In each generation the following functions are subsequently executed:

- Objective function computation
- Ranking function
- Fitness function computation
- Selection function
- Crossover function
- Mutation function
- Viral infection function

Once the aforementioned function embedded to the main function (evolution) have been executed the results concerning the candidates solutions of the final population in terms of chromosomes and corresponding phenotypes are stored to "population.dat" and "phenotype.dat" files whilst the values for the three optimisation criteria and the values for the 3D Pareto optimisation criterion are stored to "3D-Pareto_values.dat" file. The values for tool path parameters accompanying the results of "3D-Pareto_values.dat" are also stored to "variablelog.dat" file. The stopping criterion for the MOVEGA is the maximum number of preset generations.

4.3.2.3 Objective function computation

Objective function computation is performed by implementing the integrated programming application presented in section 4.3.1. By taking into account that the entire integrated programming application returns four values, three for the optimisation criteria and one for the Pareto-optimal expression it is suggested that it constitutes the objective function. Thus, the integrated programming application developed in *Dassault Systèmes CATIA® V5 R18* open API architecture, executes an evaluation process per candidate solution (tool path chromosome) which examines. The expression for the Pareto-optimal non-dominated set of solutions has already been given in Chapter 3 (Equation 3.16) and is noted again for easy reference (Equation 4.9).

$$OF_{PO} = \sqrt{\left(\overline{\delta}+\overline{h}\right)^2 + \left(\overline{stdev\delta}+\overline{stdevh}\right)^2 + \left(\sum CL\right)^2}$$

Eq. 4.9

4.3.2.4 Ranking

The resulting values of objective function computation are transferred to ranking function to prioritize them regarding their objective scores. Ranking is performed in ascending order regarding the optimisation criteria values since all three criteria are to be minimized. The workflow of ranking function for objective values is illustrated in Figure 4.11.
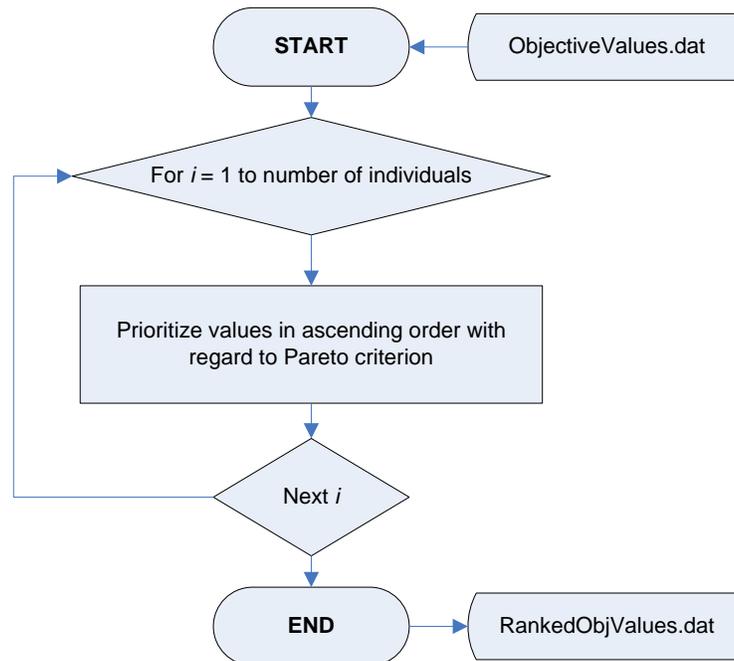
Figure 4.11: Workflow of ranking function.

4.3.2.5 Fitness function computation

The workflow of fitness function computation is depicted in Figure 4.12.
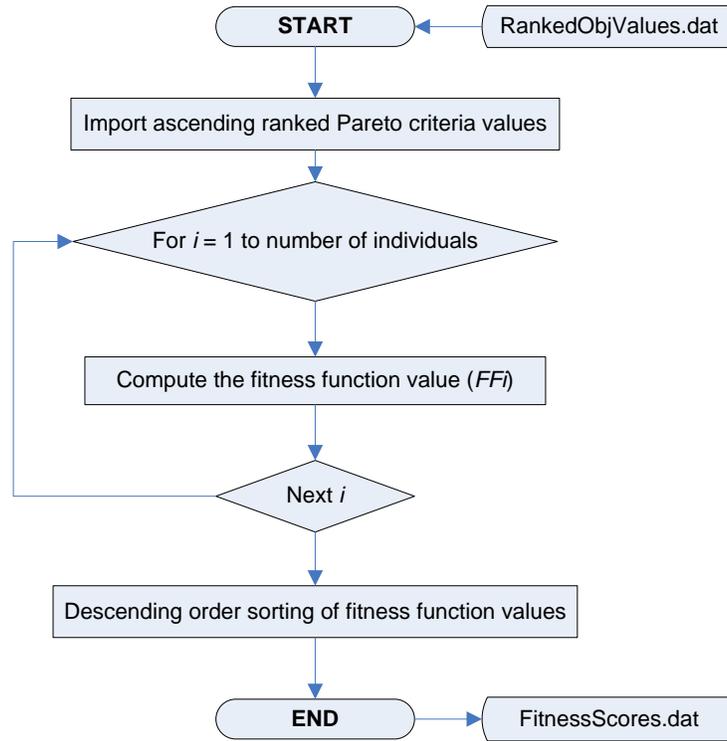
Figure 4.12: Workflow of fitness function.

According to the hierarchy performed by ranking function, fitness evaluation is conducted for each individual in MOVEGA's generated population of candidate solutions. Elite individuals in the population are considered those with the lowest values in the ranking function's hierarchy (ascending ranking). Fitness function has been determined to transform ranked objective values so that elite individuals will come up with a significantly higher fitness scores. The function computes the sum of ranking values for the entire population whilst the fitness score FF_i for each individual i is computed according to Equation 4.10.

$$FF_i = S_{RF_i} \times \exp(-2 \times RF_i)$$ Eq. 4.10

where:

S_{RF_i} :   The sum of ranked values for the entire population,

RF_i :   The ranked value of each individual i

An exponential mathematical expression has been adopted to make profound differences in the values obtained by the ranking function by magnifying their result and ultimately favor best solutions. It should be noted that differences in terms of ranked values for objectives are very small, usually

they are noticeable after the second decimal. The individuals are finally stored along with their corresponding fitness scores to a dedicated file ("fitnessScores.dat") and they are prioritized in descending order.

4.3.2.6 Selection

The workflow of selection function is shown in Figure 4.13. With reference to the fitness scores provided by fitness function, selection function is executed next to select individuals for reproduction based on their fitness scores. Stochastic sampling with partial replacement (SSPR) has been implemented as the main mechanism for selecting individuals (see section 4.2.3). The function imports the fitness scores corresponding to individuals and then computes the fitness cumulative sum for the entire population $Fitsum$. Based on this result the selection range is created with lower bound equal to zero and upper bound equal to $Fitsum$, hence, [0, $Fitsum$]. Individuals are then subsequently located according to their fitness score, i.e. [0,Fit1}, {Fit1,Fit2} ,….,{FitN-1, $Fitsum$] whilst a random generator is applied to provide values within this range. Random generator creates as many random values as the individuals to be selected for crossover. Random values will belong to one of the above sub-ranges that represent the individuals to be selected. With reference to fitness scores elite individuals are particularly favored so that elitist behaviour is maintained during the optimisation process of MOVEGA.
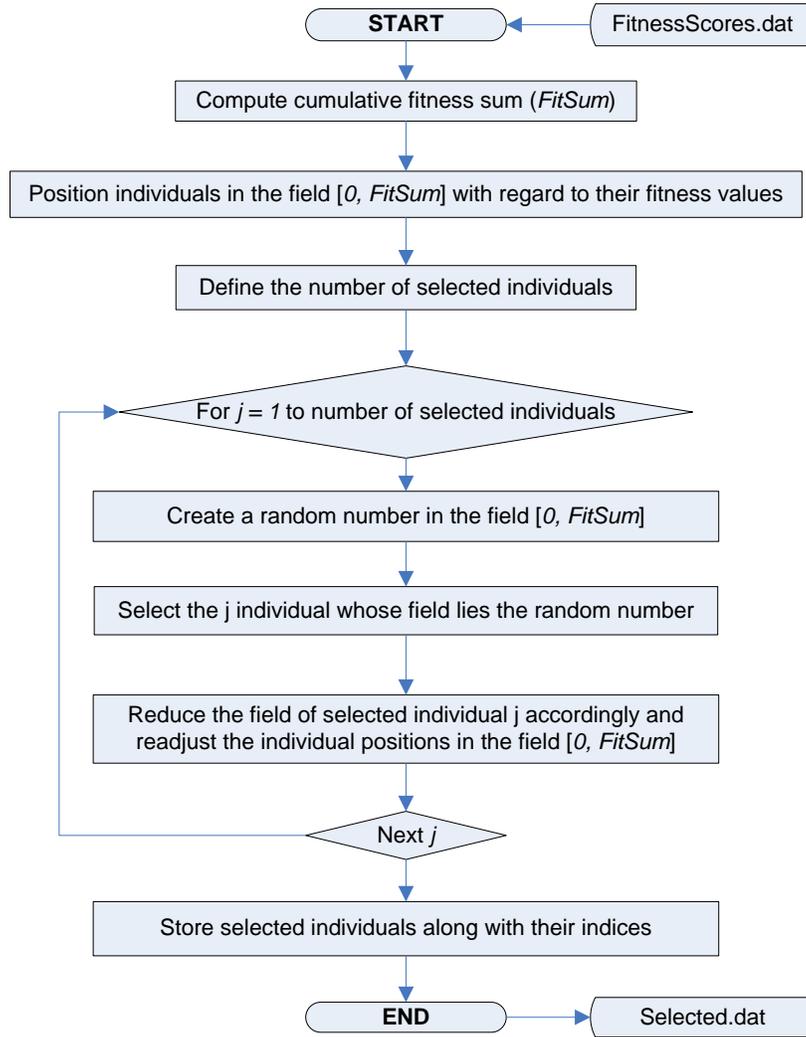
Figure 4.13: Workflow of selection function.

Once an individual has been selected, both the selection range $[0, Fitsum]$ and the individual's sub-range are restructured using the expression given in Equation 4.11 so as to prevent the repetitive selection of this particular outstanding individual.

$$Fit_i = Fit_i - \frac{Fitsum}{2N}$$

Eq. 4.11

In Eq. 4.11 Fit_i is the fitness function score of i_{th} individual, N is the population's size (total number of individuals in a population) and $Fitsum$ is the cumulative sum of the entire population. Selected individuals are stored to "Selected.dat" file to be further handled by crossover function.

114

4.3.2.7 Crossover (mating)

The method adopted to develop the function for crossover is one-point crossover owing to the advantages mentioned in section 4.2.3. The workflow of crossover function is shown in Figure 4.14. The pairs of individuals are randomly selected to produce offspring.

```
                    START  ◄──────  Selected.dat
                      │
        Input selected individuals (parents)
                      │
           Define crossover positions
                      │
   From the selected parents, randomly select pairs to mate
                      │
        For k = 1 to number of pairs of individuals
                      │
   Mate the pairs with regard to crossover positions to create
                   potential offspring
                      │
                    Next k
                      │
   Insert parents and offspring into new population with twice
                   the original size
                      │
      Call objective function to evaluate the new population
                      │
      Call ranking function to prioritize the new population
                      │
         Select the elite individuals to include in the new
                      population
                      │
                     END  ──────►  Population.dat
```

Figure 4.14: Workflow of crossover function.

After the execution of crossover function all individuals – parents and offspring – are imported to a new candidate population (individual pool) whose size is twice the original population's size. For the new population size, the objective function is computed and individuals are ranked accordingly by implementing ranking function. With reference to their hierarchy, N best individuals are deterministically selected where N the original number of individuals initially created to constitute the first population. Finally, these individuals are mutated to formulate the new population of candidate solutions (tool path chromosomes).

4.3.2.8 Mutation

Mutation operator involves the determination of the number of individuals to be mutated (NIM), the number of genes (variables) on which mutation is to be applied (NGM) as well as mutation rate $MutRate$. However, an outstanding individual being found in the population should be preserved and remained unchanged in order to avoid the loss of elite inheritance. The mutation operator imports the population of candidate solutions as a result of crossover and randomly selects offspring for mutation. For the selected offspring, pointers are assigned indicating the locations in their chromosomes for which the bits are to be switched from "0" to "1" and vice versa. Mutation rate $MutRate$ starts with a relatively high value to maintain diversity for exploration whilst it gradually reduces using a linear expression based on the preset initial value for $MutRate$ and the number of generations, to prevent the algorithm from the scattered random search (section 4.2.4). An adaptive mutation scheme is thus implemented (Thierens 2002). The relation adopted to set and gradually reduce mutation rate $MutRate$ with regard to the number of generations G is given in Equation 4.12, where $MutRate_0$ is the mutation rate for the individuals of the 1st generation and $MutRate$ the gradually reduced mutation rate for later generations. The operator scans the selected chromosome's bits and mutates every bit with mutation probability P_m equal to 0.5%.

$$MutRate = MutRate_0 - 0.005 \times G \qquad \text{Eq. 4.12}$$

The functions used for computing the number of individuals to be mutated (*NIM*) and the number of genes (variables) (*NGM*) on which mutation is applied are given in Equations 4.13 and 4.14 respectively.

$$NIM = Round\left(\frac{Mutind - 0.005 \times G}{100} \times i_{max}\right) \qquad \text{Eq. 4.13}$$

$$NGM = Round\left(\frac{Mutgen - 0.05 \times \dfrac{G}{j_{max}}}{100} \times j_{max}\right)$$

Eq. 4.14

where, i_{max}, j_{max} are the numbers of individuals and genes (variables) respectively, *Mutind* is the initial percentage (%) of selected individuals in the population for which mutation is to be applied, *Mutgen* is the initial percentage (%) of selected genes of individuals in the population for which mutation is to be applied, *Round* is a mathematical operator to round decimals to the closest integer and finally G is the number of generation in the algorithm. If the number of individuals (NIM) and the number of genes (variables) (NGM) to be mutated occur less than 1, they intentionally set equal to 1 to ensure that at least one individual and one of its bits will be mutated. Thereby the resulting population will be the next generation's first population to be evolved via crossover, mutation, viral infection and so on. The workflow of mutation function is shown in Figure 4.15.
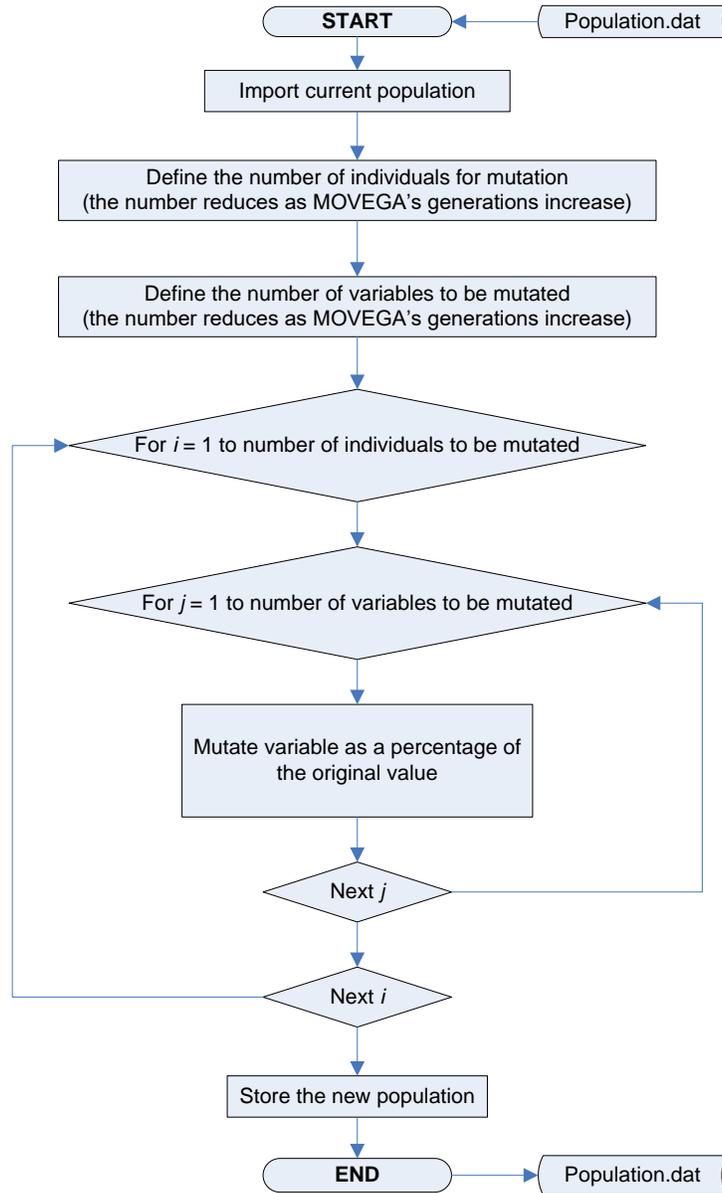
Figure 4.15: Workflow of mutation function.

4.3.2.9 Viral infection

Initial virus individuals are created as a fraction of the main population's magnitude after evaluating the fitness of all individuals existing to the main population. Then the MOVEGA performs both targeted as well as random selection of individuals to infect. The former selection is decided upon the elite of few outstanding individuals whereas the latter is normally decided upon the rest of the individuals through a probability, to sustain an unbiased selection procedure. Successfully infected individuals are appeared as offspring whilst should their fitness has been improved, they occupy the position of their ancestors, hence, replacing them. As a consequence, these individuals are to survive

in the next generation. The initial population of tool path "chromosomes", C_{Pop}^{init} is randomly generated and then *transduction* operation is applied to both fitted and randomly selected individuals to create the population of viruses V_{Pop}^{init}. The viruses are stored in binary representation to a related archive ("virus_population.dat"). A virus created by transducing from the i_{th} chromosome of the j_{th} population is denoted as Vrs_{ij}. Substrings being cut represent viruses' chromosomes whose length is denoted as $VrsLgth_i$. Locus $i = 1$ is the starting point from which $VrsLgth_i$ length will be determined and Locus i_{max} is the ending point. These two loci are randomly determined and are constrained to the original host's chromosome length $Lgth_i$. The chromosome length ($Lgth_i$) of individuals in the main population is constant, whereas the length of virus individuals ($VrsLgth_i$) extends as the evolution process continues $\left(VrsL\,g\,th_i = V_{strlength}\max \right)$. The index of the population where selected individuals have been attacked for infection is given in the archive "infected_host_population.dat" whilst "virus_phenotype.dat" archive includes the phenotypes of the individuals as candidates for being attacked to be infected. Finally, the objective values for viruses are printed to "virusobjvalues.dat" archive.

Transduction and *reverse transcription* are the main procedures of viral infection. As it has been mentioned, *transduction* operation is applied to individuals to create the population of viruses. Viruses Vrs_{ij} attack to infect individuals, using *reverse transcription* for overwriting their own substrings to a randomly selected segment of individuals' Idv_j strings. The indices of both Vrs_{ij} and Idv_j are declared in advance in order to perform the subsequent replacement of selected binary digits according to the predetermined references. Transduction and reverse transcription operators are depicted in Figures 4.16 and 4.17 respectively. Figure 4.18 depicts an infected individual.
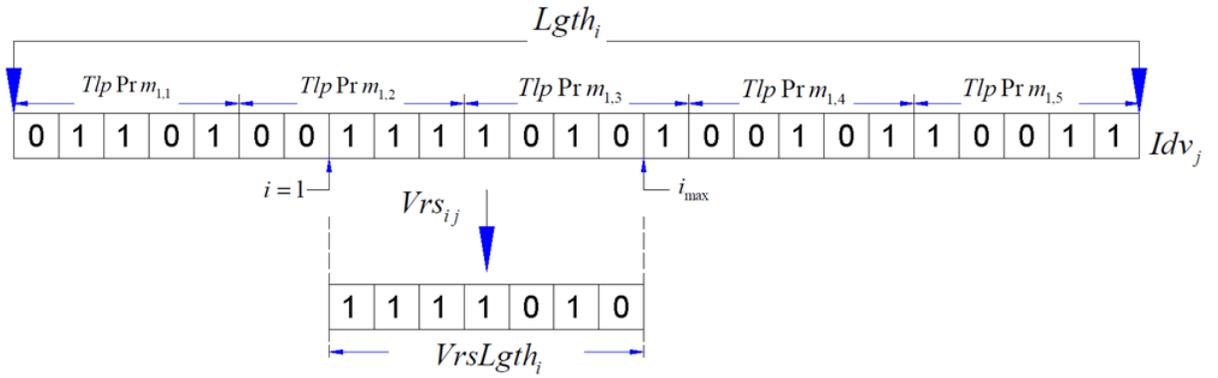
Figure 4.16: Transduction operation for the creation on a virus individual.
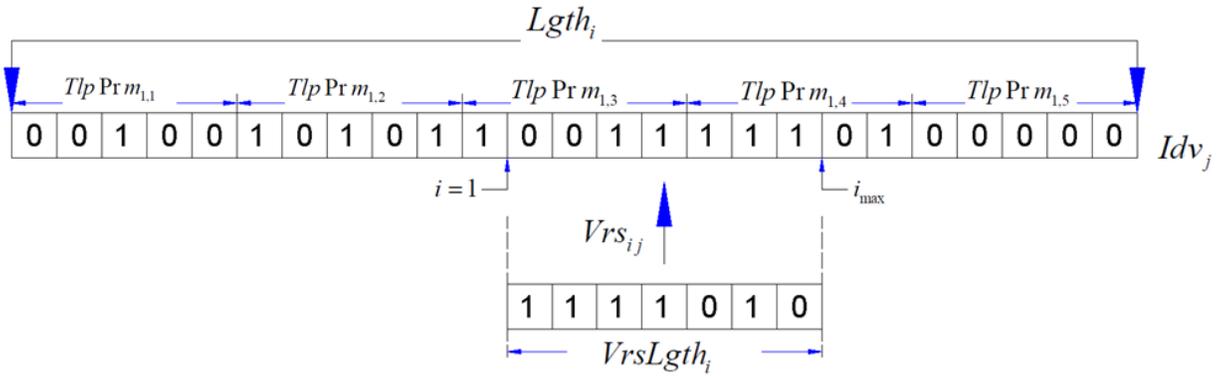
Figure 4.17: Reverse transcription operation for infecting an individual with a virus.
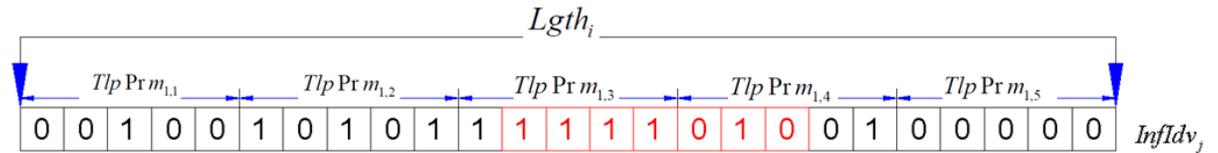
Figure 4.18: Infected individual after the reverse transcription operation performed by the virus.

Assessment of virus individuals is performed using their fitness scores denoted as $FitVrs_{i,j}$ reflecting their infection strength. This fitness is computed after the successful infection of Idv_j by Vrs_{ij} as Equation 4.15 indicates:

$$FitVrs_{i,j} = FitInfIdv_j - FitIdv_j$$ Eq. 4.15

The value obtained by Equation 4.15 is the difference between the two fitness values of individual Idv_j before and after its infection by Vrs_{ij}. Given that Vrs_{ij} might infect more than a single individual (let S be the set of infected individuals) then $FitVrs_{i,j}$ reveals the improvement of fitness values of all infected individuals and is as Equation 4.16 determines:

$$FitVrs_i = \sum_{j \in S} FitVrs_{i,j}$$

Eq. 4.16

A virus Vrs_{ij} has a maximum viral infection rate $V_{\inf Rate} \max$ for controlling the number of viral infections satisfying the condition $1 \leq V_{\inf Rate} \max \leq 10$. As a result the number of reverse transcriptions a single virus is to perform depends on its viral infection rate. However the maximum viral infection rate $V_{\inf Rate} \max$ is related also to its fitness value $FitVrs_{i,j}$ under the notion that the higher the fitness $FitVrs_{i,j}$ the higher the $V_{\inf Rate} \max$. Equation 4.17 gives the relation employed to the algorithm so as to correlate the aforementioned viral infection parameters and control $V_{\inf Rate} \max$ with regard to the virus fitness $FitVrs_{i,j}$. In Equation 4.17, $a(>0)$ is a constant coefficient for improving or degrading $V_{\inf Rate} \max$ parameter with regard to either the positive or the negative results for the fitness of a virus Vrs_{ij}.

$$V_{\inf Rate} \max_{i,G+1} = \begin{cases} (1+a) \times V_{\inf Rate} \max_{i,G} \\ (1-a) \times V_{\inf Rate} \max_{i,G} \end{cases}, \begin{array}{l} FitVrs_i \geq 0 \\ FitVrs_i < 0 \end{array}$$

Eq. 4.17

Every virus Vrs_{ij} is accompanied also to its corresponding life force indicating its contribution through successful infections to the main population. The life force of a virus Vrs_{ij} is presented as $VrsLiforce_{i,G}$ where i is the index of the virus Vrs_{ij} and G the current generation. $VrsLiforce_{i,G}$ is also dependent from the fitness of a virus Vrs_{ij} and is compared to the one obtained by the virus Vrs_{ij} in a previous generation. If its value is negative, then a new *transduction* operation is applied by the virus Vrs_{ij} to change its scheme by randomly selecting an individual. Otherwise Vrs_{ij} cuts a partially new substring form one of the successfully infected individuals for its own benefit from the

121

evolutionary viewpoint. The magnitude of $VrsLiforce_{i,G}$ parameter is computed in each generation with regard to an important indicator which is the virus life reduction rate $rate\downarrow V_{life}$ satisfying $0.001 \le rate\downarrow V_{life} \le 1.0$. Hence, maximum viral infection rate $V_{inf\,Rate}\max$ and virus life reduction rate $rate\downarrow V_{life}$ are related through the relation presented in Equation 4.18.

$$VrsLiforce_{i,G+1} = rate\downarrow V_{life} \times VrsLiforce_{i,G} + FitVrs_i$$

Eq. 4.18

The parameters, $V_{inf\,Rate}\max$ and $VrsLiforce_{i,G}$ are initialized in MOVEGA as $V_{inf\,Rate}\max_{init} = V_{inf\,Rate}\max_{i,0}$, $VrsLiforce_{i,0} = 0$. The operation of partial transduction in the case where $VrsLiforce_{i,G} > 0$ is depicted in Figure 4.19 with reference to transduction and reverse transcription operations depicted in Figure 4.16 and 4.17 above.
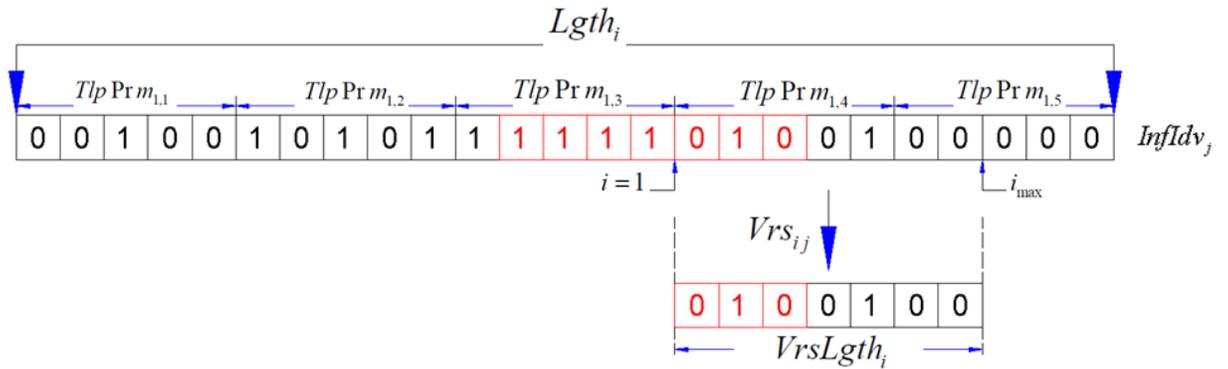
Figure 4.19: Partial transduction operation for changing the virus scheme.

With reference to the results obtained by conducting several algorithmic experiments and research work on the application of the MOVEGA to the generalized sculptured surface CNC machining problem for optimizing it, its overall contribution and as well as the features giving the added value to the problem's solutions, have been recognized and are summarized as follows:

- Co-evolution among candidate solutions (main individuals) and viruses ("partial" information carriers) enables MOVEGA to efficiently question new solutions owing to horizontal propagation which is beneficial for local information handling and vertical inheritance. As a result, global search is facilitated with the aid of local data as well.

122

- Information exchange for guiding the search is facilitated through the deployment of viral "agents" as local information carriers whilst their algorithm-specific parameters allows for the self-adaptive change of searching ratio, switching from local to global search and vice versa regarding the current status of evolution progress.

- Viral operators prevent the search from being locally trapped since they rapidly propagate schemata and maintain high genetic diversity in the population of candidate solutions.

Figure 4.20 depicts the workflow of viral infection in the MOVEGA and Figure 4.21 depicts the overall workflow of the methodology proposed for optimizing the generalized sculptured surface CNC machining problem.
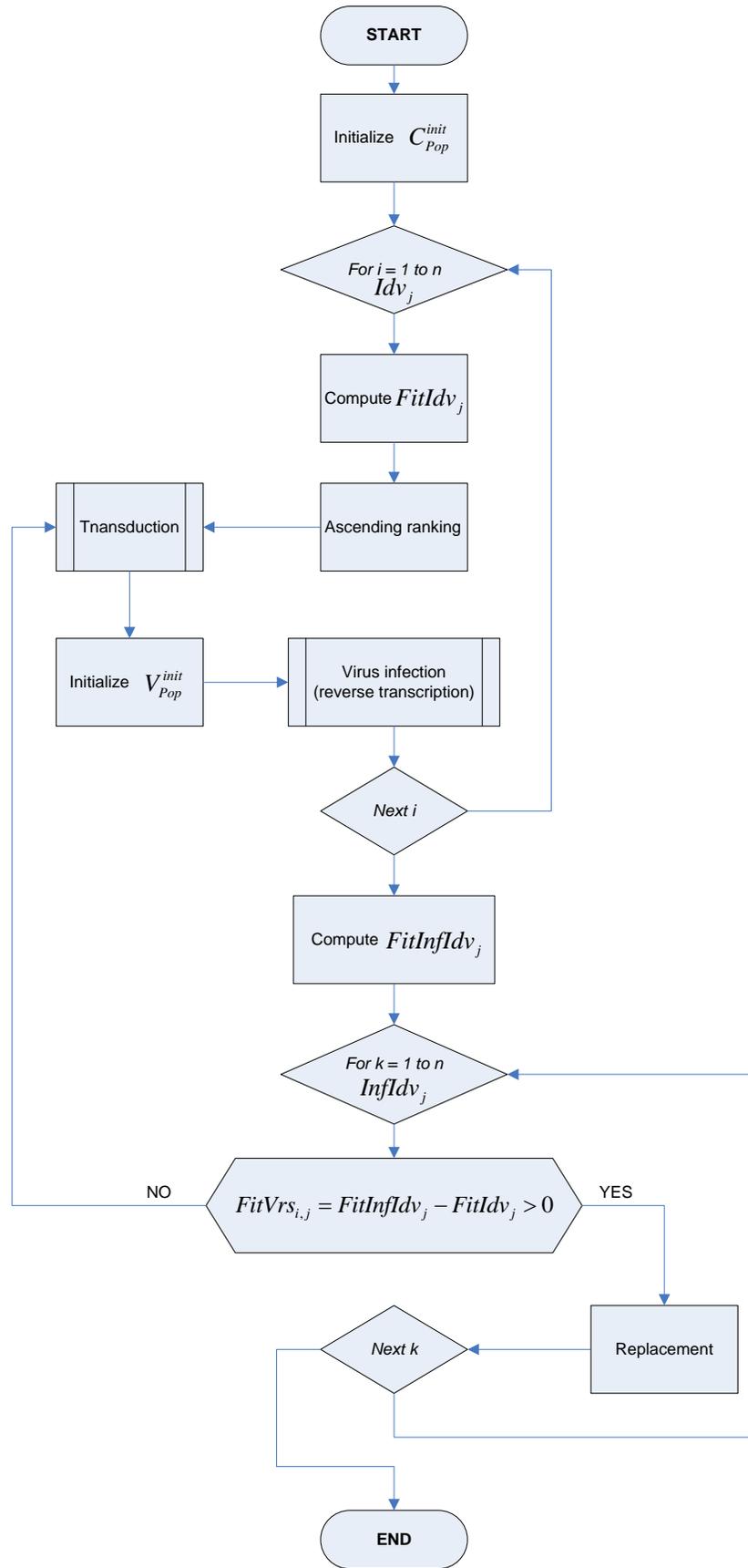
Figure 4.20: Workflow of viral infection after the evaluation of main population's individuals.
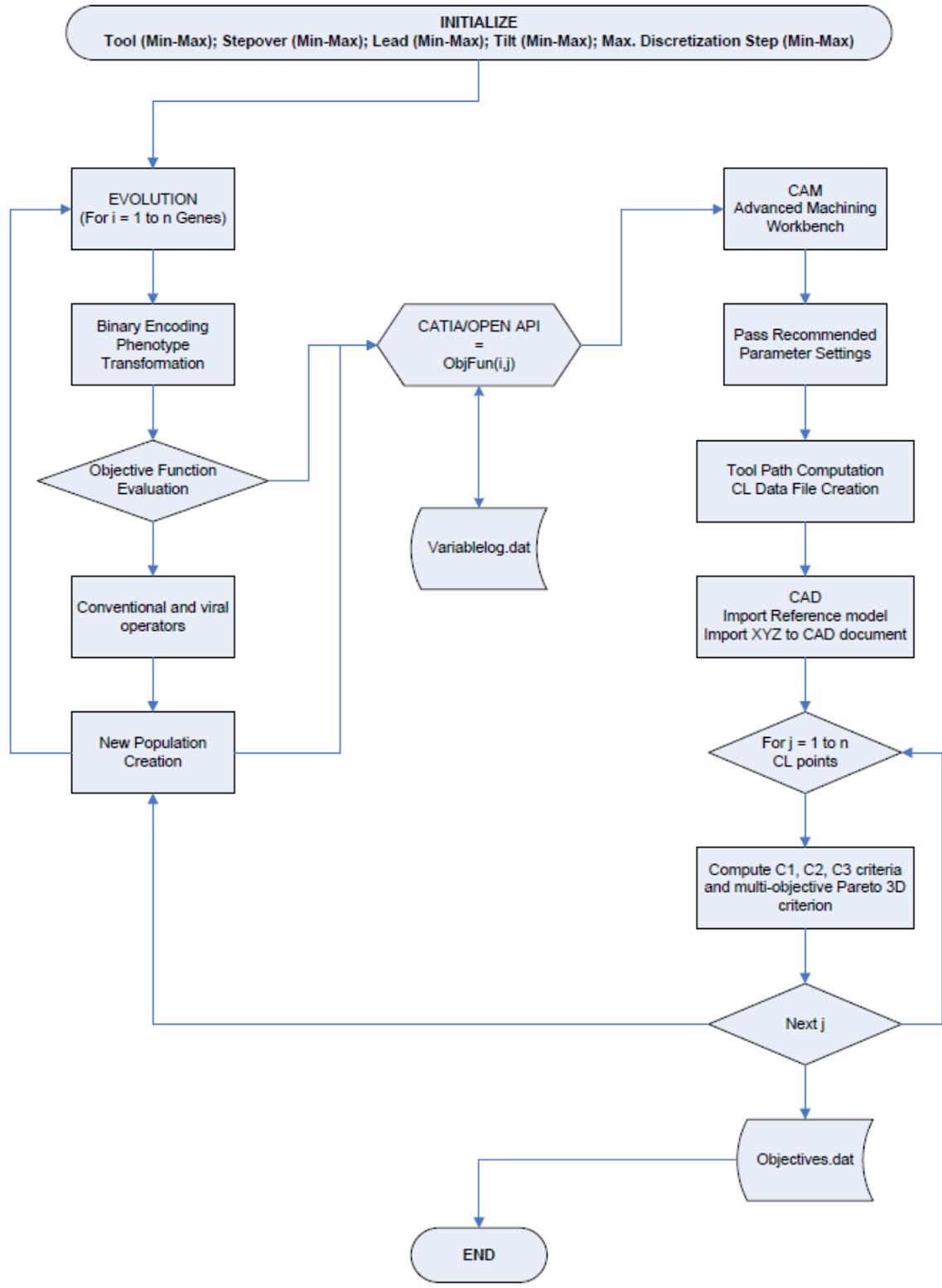
Figure 4.21: Overall workflow of the proposed methodology for optimizing the generalized sculptured surface CNC machining problem.

The overall performance of stochastic algorithms is heavily dependent on the proper selection of algorithm-specific parameters. To make the MOVEGA to reach its full potentials a thorough study in terms of its functional behaviour, efficiency of attaining optimal solutions and repeatability of accurate results needs to be conducted to further investigate the capabilities of adjusting optimal settings for its algorithm-specific parameters. The next chapter presents such a study having the aims and the objectives mentioned above, with the proposed algorithm to handle a generalized sculptured surface CNC machining optimisation problem with reference to a benchmark sculptured part and a systematic approach for designing machining simulation experiments.

Chapter 5
Algorithm-specific parameters identification for multi-objective virus-evolutionary GA

5.1   Introduction

Major scope of this chapter is to examine the efficiency of the multi-objective virus-evolutionary genetic algorithm (MOVEGA) on solving the sculptured surface machining problem as well as to identify the most beneficial (if not the optimal) settings related to its algorithm-specific parameters presented in Chapter 4. Towards achieving this goal, a response surface experimental design was established to systematically determine a series of sculptured surface machining experiments applied on a benchmark surface and to obtain results for examining performance attributes such as convergence speed, coverage (distribution) of non-dominated solutions in Pareto fronts and quality of optimal solutions. Response surface methodology (RSM) has been selected to design machining experiments to estimate interactions among quadratic and high-order effects, to estimate the curvature and overall shape of the response surface under investigation. The major objective formulating the response surface is the multi-objective criterion as it has been presented so far, expressing the machining error, the machining error uniformity and the number of tool path points. In order to generate meaningful results for investigation towards the goal of optimally adjusting MOVEGA's settings, tool path parameters were determined such that profound differences among performance attributes would occur. The experimental methodology involves:

- the experimental setup,
- the conduction of machining simulation experiments with different algorithm-specific parameters,
- the experimental results interpretation, and
- the final decision making as regards the algorithm's parameter settings and the conduction of confirmation experiments.

5.2 Design of response surface methodology experiments for algorithm-specific parameter tuning

Response surface methodology (RSM) was first introduced by Myers 1971 and it has been employed to study and optimise the process parameters for numerous engineering problems ever since. RSM is a multivariate experimental design methodology that involves fundamental mathematical and statistical approaches for determining empirical models to characterise a problem's navigation domain and further optimise its corresponding process parameters with regard to either a single or multiple responses. Such empirical models correlate the independent variables which are the process parameters to the response. To generate a reliable empirical model to study a problem at hand, a set of experiments were conducted so as to collect their corresponding outputs and evaluate them properly to fit the model in relation to experimental outputs. In RSM two experimental design approaches are the most often-used, "Box-Behnken" and "central composite design-CCD" (Box and Hunter 1957). Box-Behnken approach offers the advantage of establishing experiments with a reduced number of runs when it comes to three factors (independent variables), however, in the case of four or more factors the number of experiments required for fitting a model increases dramatically compared to that of CCD approach. A significant advantage of CCD against Box-Behnken approach is that CCD augments an embedded factorial design which is very important for examining the main effects as well, along with interactions.

In a typical CCD the number of experiments required involve the standard 2^k factorial runs having their origin at the center, *2k* axial (or star) points in a distance a from the center so as to create the quadratic terms and *n* points as replicates at the center for estimating the experimental error, where *k* is the number of parameters. Axial points are selected such that they ensure rotatability, that is, the empirical model's variance in terms of its prediction is constant at all points equidistant from the design's center (Box and Hunter 1957). Moreover axial points provide screening analysis and readability to check the variance of the empirical model and is fixed at all points equidistant from the design center (Behera et al. 2018) The total number of experiments required for the CCD with 2^k factorial points, *2k* axial points and *n* points-replicates is thus determined by Equation 5.1.

$$N_{CCD} = 2^k + 2 \times k + n$$

Distance a is $a = \left(2^k\right)^{0.25}$.

The results obtained by the CCD experiments are further processed using statistical analysis and response surface regression in order to generate the empirical model for correlating independent parameters to responses by fitting them using a 2nd order polynomial relation presented in Equation 5.2.

$$Y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_{ii}^2 + \sum_{i=1}^{k} \sum_{i \neq j=1}^{k} \beta_{ij} x_i x_j \qquad \text{Eq. 5.2}$$

where Y represents the responses, k is the overall number of independent variables, β_0 is a constant term and *i, ii, ij* accompanying β are the coefficients for linear, quadratic and interaction terms respectively. Finally, x_i, x_j represent the coded levels for the independent parameters.

The MOVEGA was implemented to optimise the 5-axis surface machining tool path for a benchmark sculptured surface other than those already presented in the thesis. Major objectives of the application of MOVEGA to this benchmark sculptured surface were:

- To examine its operational behaviour under different settings in terms of its algorithm-specific parameters,
- To determine the effect of MOVEGA's algorithm-specific parameters and to identify to what degree they influence its stochastic evaluation performance,
- To find the optimal or at least a semi-optimal set of parameter values for tuning MOVEGA and study its enhanced performance regarding the set of non-dominated solutions.

The benchmark sculptured surface was a bi-cubic Bezier surface determined by a 4x4 array of control points (Choi and Banerjee 2007). In order to establish a meaningful problem domain, the range of values for 5-axis tool path parameters were examined in advance through preliminary experiments so that results would yield profound differences and variation to further investigate the effect of MOVEGA's algorithm-specific parameters. For all parameters the number of accuracy digits to formulate the tool path chromosomes for algorithmic evaluations was fixed to 20. Table 5.1 summarizes the 5-axis tool path parameters and their range of values whereas Figure 5.1 illustrates the benchmark part and the tool path applied.

Table 5.1: 5-axis tool path parameter values corresponding to Bi-cubic Bezier benchmark surface for algorithmic evaluations.

Bi-cubic Bezier surface	Levels	Tool	Step over (%D)	Lead angle (deg)	Tilt angle (deg)	MaxDstep (mm)
	Low	D16-Rc0	30	10	0	2
	High	D16-Rc4	45	20	5	8
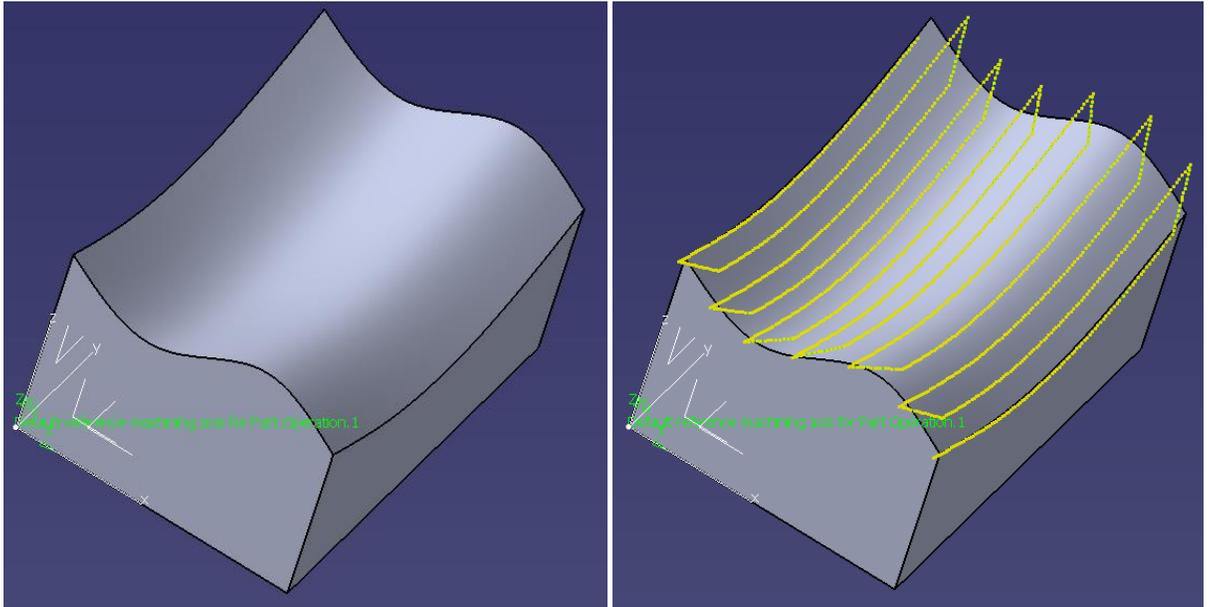
Figure 5.1: Experimental Bi-cubic Bezier benchmark sculptured part and 5-axis machining tool path.

The response surface experiment established had 31 runs according to the relation given in Eq. 5.1, consisting of 16 factorial points, 8 star points and 7 replicates to estimate the experimental error. Hence, the MOVEGA handled the sculptured surface machining optimisation problem for the aforementioned benchmark surface with different algorithm-specific parameter settings as per the 31 runs in the CCD design. The algorithm-specific parameters of MOVEGA are the virus population size V_{pop}, the maximum length of a virus chromosome $V_{strlength} \max$, the virus life reduction rate $rate \downarrow V_{life}$ and maximum infection rate $V_{\inf Rate} \max$ as presented in Chapter 4. To facilitate curvature examination in the experimental outputs concerning the MOVEGA performance and its parameter effects the parameters were assumed to be continuous.

To study the overall performance of MOVEGA in solving the sculptured surface machining optimisation problem and examine the effect of algorithm-specific parameters (viral operators) a set of indices was investigated, diversity, spacing (coverage), convergence speed, as well as best values of non-dominated solutions. Diversity exhibits the amount of success in terms of the adaptation of populations (candidate solutions) to changing environments whilst a high value for diversity is generally preferable. Spacing or coverage indicates the spread distribution within non-dominated solutions existing in a Pareto-front. Convergence speed shows the accuracy of obtained solutions and is represented through the objective function's results in relation to the number of function evaluations. During the objective function's evaluations, a characteristic convergence curve (slope) is

formulated indicating the speed of an algorithm's clustering towards the optimal result ending up with an asymptotic straight line parallel to the function's evaluations showing the end of the search. The narrower the convergence slope occurs, the higher the convergence speed is. Best and worst values of non-dominated solutions depend of the optimisation problem's nature (maximization or minimization) whereas the average of all non-dominated solutions should be close to the optimal result (either minimum or maximum) to indicate the low variance among the set of solutions regarding the optimal one. The number of non-dominated solutions in the related repository (archive) provides also an important performance index. In general, a wide range of solutions is preferred to facilitate final decision making. Table 5.2 summarizes the factors of the CCD response surface experiment established along with their upper and lower levels of parameter values and Figure 5.2 illustrates this design for k = 4 factors.

Table 5.2: Experimental algorithm-specific parameters and corresponding levels for the RSM-CCD design of experiments.

Levels	V_{pop}	$V_{strlength}\,\mathrm{max}$	$rate \downarrow V_{life}$	$V_{\inf Rate}\,\mathrm{max}$
Low	2 (1/5 C_{pop})	10	0.001	1 (10%)
High	10 (= C_{pop})	40	1.000	10 (100%)
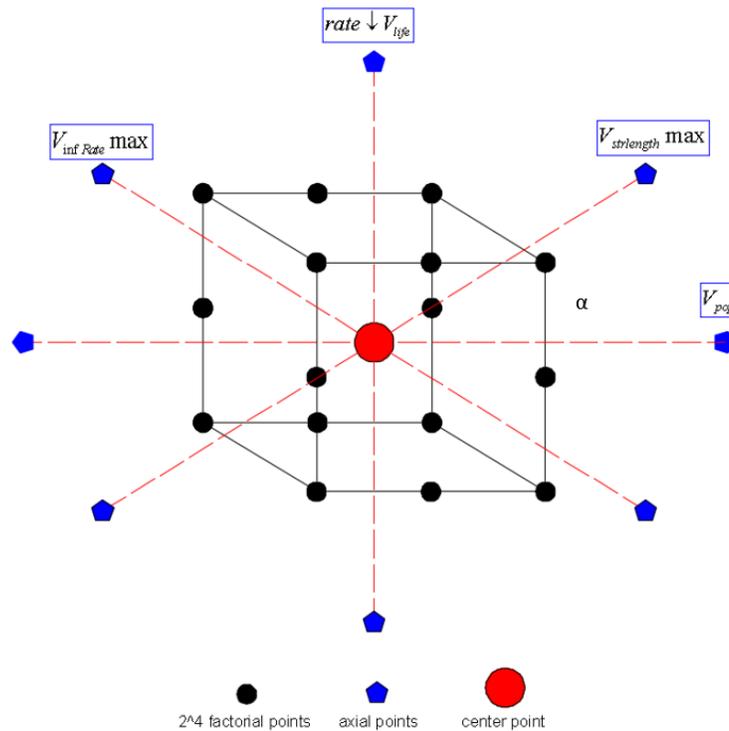
Figure 5.2: Graphical illustration of the CCD response surface design.

5.3   Results and analysis

The 31 experiments were randomly conducted using the intelligent tool path optimisation environment presented in the thesis. In all experiments the MOVEGA evaluated the problem via the automation function presented in Chapter 4, section 4.3.1, with constant parameters such as the number of generations, number of candidates in a population, single-point crossover scheme and single-point mutation scheme. Thus, the number of generations was set equal to 5 and the number of candidates in a population equal to 10. Table 5.3 summarizes the results from the RSM-CCD experiment with regard to the 3D Pareto criterion (minimal) along with the normalized individual criteria, machining error (C_1), machining error distribution (C_2) and number of cutting points (C_3).

Table 5.3: Experimental results of the RSM-CCD design of experiments referring to individual criteria and 3D Pareto criterion.

a/a	Algorithm-specific parameter levels				Pareto3D	C_1	C_2	C_3
	V_{pop}	$V_{strlength}$ max	$rate \downarrow V_{life}$	$V_{\inf Rate}$ max				
Factorial points								
1	2	10	-0.001	1.0	0.301983	0.048	0.201	0.220
2	10	10	-0.001	1.0	0.291034	0.029	0.087	0.276
3	2	40	-0.001	1.0	0.297922	0.047	0.114	0.271
4	10	40	-0.001	1.0	0.300670	0.024	0.114	0.277
5	2	10	-1.000	1.0	0.286510	0.040	0.182	0.217
6	10	10	-1.000	1.0	0.304125	0.023	0.116	0.280
7	2	40	-1.000	1.0	0.307945	0.084	0.183	0.233
8	10	40	-1.000	1.0	0.297509	0.024	0.144	0.259
9	2	10	-0.001	10	0.307419	0.058	0.189	0.235
10	10	10	-0.001	10	0.280594	0.020	0.174	0.219
11	2	40	-0.001	10	0.294101	0.045	0.114	0.267
12	10	40	-0.001	10	0.251180	0.017	0.120	0.220
13	2	10	-1.000	10	0.309630	0.055	0.127	0.277
14	10	10	-1.000	10	0.267812	0.048	0.150	0.216
15	2	40	-1.000	10	0.308370	0.048	0.126	0.277
16	10	40	-1.000	10	0.273437	0.042	0.149	0.225
Axial points								
17	5.6	25	-0.500	5.5	0.297933	0.035	0.104	0.277
18	6.4	25	-0.500	5.5	0.294029	0.025	0.190	0.223
19	6	23.5	-0.500	5.5	0.305419	0.032	0.115	0.281
20	6	26.5	-0.500	5.5	0.296163	0.030	0.100	0.277
21	6	25	-0.450	5.5	0.304467	0.025	0.207	0.221
22	6	25	-0.550	5.5	0.263750	0.033	0.146	0.217
23	6	25	-0.500	5.05	0.272615	0.034	0.157	0.220
24	6	25	-0.500	5.95	0.260985	0.035	0.137	0.219
Center points								
25	6	25	-0.500	5.5	0.296652	0.032	0.101	0.277
26	6	25	-0.500	5.5	0.290789	0.035	0.182	0.224
27	6	25	-0.500	5.5	0.284134	0.040	0.175	0.220
28	6	25	-0.500	5.5	0.282278	0.030	0.176	0.218
29	6	25	-0.500	5.5	0.295910	0.042	0.113	0.270
30	6	25	-0.500	5.5	0.294771	0.043	0.088	0.278
31	6	25	-0.500	5.5	0.294648	0.034	0.192	0.221

5.3.1 Main experimental observations

According to Table 5.3 the lowest (best scores) values for 3D Pareto criterion are achieved by determining a relatively large number of viruses, i.e., approximately half of the number of individuals of the main population or equal to the number of main population's individuals. Experimental runs 12, 24 and 22 indicate 3D Pareto values equal to 0.251180, 0.260985 and 0.263750 respectively. For these results the number of viruses are 10, 6 and 6 respectively whilst the rest of parameters, $V_{strlength} \max$, $rate \downarrow V_{life}$ and $V_{\inf Rate} \max$ seem to balance accordingly with reference to the number of viruses. It is observed that for a large number of viruses i.e. 10 the increased length in their chromosome strings favors the result. The same also goes for the infection rate in the case of a large number of viruses in the virus population. As it is evident by these results, life reduction rate of viruses $rate \downarrow V_{life}$ doesn't need to be rapidly reduced for prompting the algorithm to proceed on new transductions towards finding new efficient schemes since many viruses already contain an adequate amount of genetic information for improving candidate solutions. The results reported in experimental runs 22 and 24 which are results from axial points, indicate that it is possible to obtain good outputs with fewer viruses (i.e. V_{pop} = 6) provided that the rest of parameters are adjusted accordingly. The maximum length of viral chromosome strings $V_{strlength} \max$ is the ¼ (25 bits) to the overall main population's chromosome length which is 100 bits. A reduced number of virus individuals, seems to point out the necessity of performing new transductions through the virus life reduction rate $rate \downarrow V_{life}$ which is equal to -0.5 and -0.55 for these two axial point experimental runs.

The maximum infection rate $V_{\inf Rate} \max$ in the case of fewer viruses is reduced from 10 (100%) to 5.5 or 5.95->6.0 (55% or 60%) to achieve minimised objective values. Thus, it can be advocated that "infectivity" is, in a way, dependant to the number of viruses (V_{pop}).

As regards the individual criteria, machining error (C_1), machining error distribution (C_2) and number of cutting points (C_3) it is observed that their magnitudes span normalized ranges from 0.020 to 0.084, 0.087 to 0.207 and 0.216 to 0.281 respectively. From the combinations of results for the three criteria it is clear that a trade-off exists among machining error, its uniformity and number of cutting points. The lowest (best) value for C_1 is observed in experimental run 10 (factorial run) suggesting the maximum number of viruses, minimum number of chromosome string length, minimum virus life reduction rate and maximum viral infection rate. On the contrary the maximum

(worst) value for C_1 is observed in experimental run 7 (factorial run) suggesting the lowest number of viruses, maximum number of chromosome string length, maximum virus life reduction rate and minimum viral infection rate. The lowest (best) value for C_2 is observed in experimental run 2 (factorial run) suggesting the maximum number of viruses, maximum number of chromosome string length, minimum virus life reduction rate and minimum viral infection rate. On the contrary the maximum (worst) value for C_2 is observed in experimental run 21 (axial point run) determining 6 viruses, 25 bits for the maximum number of chromosome string length, -0.45->-0.4 (40%) maximum virus life reduction rate and 5.5 (55%) viral infection rate. However the same set of algorithm-specific parameters with different virus life reduction rate (i.e. from -0.45 to -0.55) are capable of improving the 3D Pareto result where C_1 = 0.033, C_2 = 0.146 and C_3 = 0.217. The lowest (best) value for C_3 is observed in experimental run 14 (factorial run) suggesting the maximum number of viruses, minimum number of chromosome string length, maximum virus life reduction rate and maximum viral infection rate. On the contrary the maximum (worst) value for C_3 is observed in experimental run 19 (axial point run) determining 6 viruses, 23.5->24 bits for the maximum number of chromosome string length, -0.5 (50%) maximum virus life reduction rate and 5.5 (55%) viral infection rate. Further investigation on the effects of algorithm-specific parameters has been experimentally identified with the aid of convergence diagrams by considering the number of function evaluations. The latter is determined using the relations given in Equations 5.3 and 5.4. Eq. 5.3 determines the number of function evaluations ($fncEvals$) with regard to the number of individuals in the main population (C_{pop}), the number of viruses (V_{pop}) and the number of generations (G) whereas Eq. 5.4 determines the number of function evaluations ($fncEvals$) by considering only the number of individuals in the main population (C_{pop}) and the number of generations (G).

$$fncEvals = \left[\left(C_{pop} + 2 \times C_{pop} \right) + \left(V_{pop} \times 2 \times G \right) \right] \times G \qquad \text{Eq. 5.3}$$

$$fncEvals = \left(C_{pop} + 2 \times C_{pop} \right) \times G \qquad \text{Eq. 5.4}$$

It is evident that the MOVEGA requires theoretically 3 times the computational cost when compared to the same algorithm without deploying the viral operators, yet, this magnitude is deemed of minor importance since the function checks for individuals identical to those previously evaluated and whose objective function result has been already printed in the *variablelog*.dat file. Moreover, new technologies referring to hardware and computer systems allow for a significant reduction of the required computational cost.

With reference to the results obtained for different numbers of function evaluations (Eq. 5.3) the MOVEGA's performance in terms of the convergence speed and optimal result was examined. By considering the population of viruses as a fraction of the main population's size which is 10 individuals, convergence curve was investigated referring to its slope and number of function evaluation where the optimal point was reached. To exhibit fast convergence, the slope should be the narrowest possible with its corresponding asymptotic straight line to indicate an early convergence to the lowest point. Figure 5.3 depicts the convergence results for 2 viruses as a fraction of the main population size (V_{pop} =2 = $\frac{1}{5}C_{pop}$) and 2 viruses suggest 250 function evaluations according to Eq. 5.3.
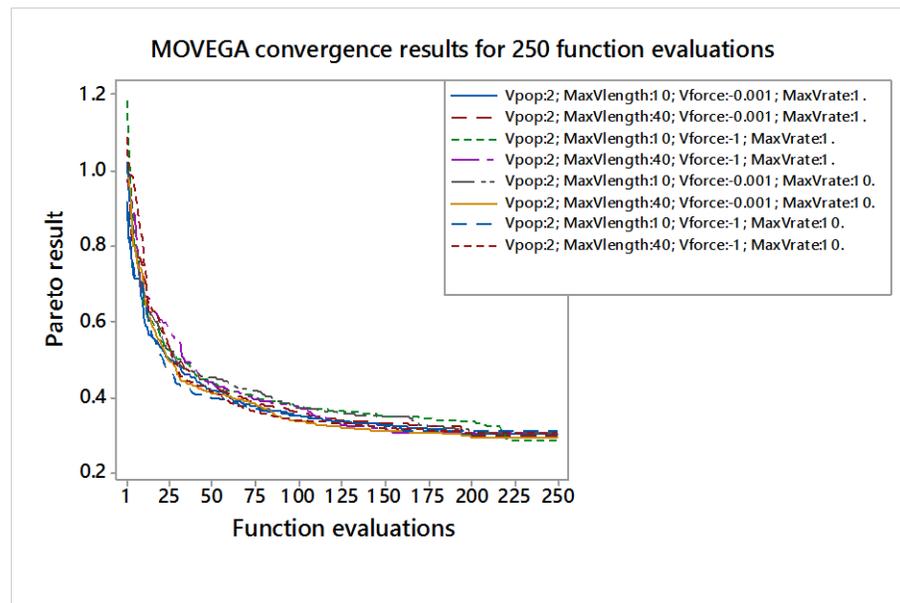
Figure 5.3: Convergence results for factorial runs (2 viruses-250 function evaluations).

By looking at the eight factorial experimental runs with V_{pop} =2, (Table 5.3) it can be seen that the algorithm's performance in terms of the convergence slope improves when determining $V_{strlength}$ max =10, $rate \downarrow V_{life}$ = -1.0 (virus life reduction rate) and $V_{inf\,Rate}$ max =10 (100% viral infection rate). On the other hand, fast convergence to lowest point does not necessarily mean that the latter is guaranteed. This result is given by the 13th experimental run (7th result in the corresponding diagram). For this result the final point is equal to 0.309630 reached in the 168th function evaluation. The best score (0.286510) is noticed in the 5th experimental run, 226th function evaluation (3rd result in the corresponding diagram) where $V_{strlength}$ max =10, $rate \downarrow V_{life}$ = -1.0 and $V_{inf\,Rate}$ max =1.0 (10% viral infection rate) are determined. For this set of 250 function evaluations (V_{pop} =2) the earliest

convergence (155th evaluation) has been given by the 7th experimental run (4th result in the corresponding diagram) with a final point equal to 0.307945. This result determines $V_{strlength}$ max =40, $rate \downarrow V_{life}$ = -1.0 and $V_{inf\,Rate}$ max =1.0 (10% viral infection rate). The results suggest that fast convergence in terms of both narrow slope and fewer function evaluations may be achieved yet, a trade-off between convergence speed and best score seems to exist. By considering the latter index as the "accuracy" the results so far agree with the literature regarding the fact that speed and accuracy in algorithmic experiments are contradictory attributes (Mirjalili et al. 2017).

Figure 5.4 depicts the convergence results for 10 viruses as an equal magnitude of the main population size (V_{pop} =10 = C_{pop}) and 10 viruses suggest 650 function evaluations according to Eq. 5.3.
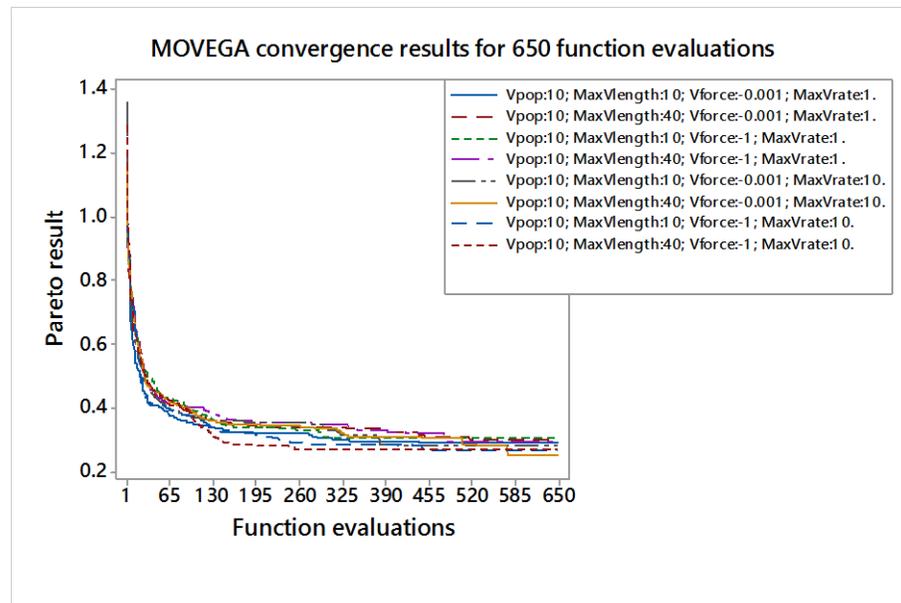
Figure 5.4: Convergence results for factorial runs (10 viruses-650 function evaluations).

By looking at the rest eight factorial experimental runs with V_{pop} =10, (Table 5.3) it can be observed that the algorithm's performance in terms of the convergence slope improves when determining $V_{strlength}$ max =40, $rate \downarrow V_{life}$ = -1.0 and $V_{inf\,Rate}$ max =10 (100% viral infection rate). At an early stage the previous assumption that the final result is favored by determining an increased length (i.e. 40 bits) for the viral chromosome string in the case of V_{pop} =10 is proved to be valid. At the same time the higher level of viral chromosome string length ($V_{strlength}$ max) parameter seems to facilitate both the narrow slope needed for fast convergence (16th experimental run - 8th result in the corresponding

diagram) and fast convergence speed. For this result the output is equal to 0.273437 obtained in 256th function evaluation. However the lowest point which is equal to 0.251180 is obtained for $V_{strlength}$ max =40, $rate \downarrow V_{life}$ = -0.001 and $V_{inf\ Rate}$ max =10 (100% viral infection rate) and obtained in 584th function evaluation (12th experimental result - 6th result in the corresponding diagram). The results in this case clearly indicate the trade-off between speed and accuracy to virus life reduction rate parameter. $rate \downarrow V_{life}$ = -0.001 facilitates best score whilst $V_{inf\ Rate}$ max = -1.0 which controls the number or viral infections, facilitates speed (slope and fast convergence) as opposed to the case of V_{pop} =2. A common characteristic for V_{pop} = 2 and V_{pop} =10 is that in both cases acceptable solutions may be attained in later function evaluations indicating the avoidance of local stagnation, despite the lower convergence speed.

The two cases examined so far (V_{pop} = 2 and V_{pop} =10) conclude the investigation of factorial points in the RSM-CCD design generally suggesting that when it comes to the employment of only few viruses i.e. V_{pop} = 2, local search is not facilitated and the problem's solution is mainly based on global search through the performance of conventional operators (crossover and mutation). Thus, a significant number of transductions to create new viruses is normally required ($rate \downarrow V_{life}$ = -1.0) in order to update the scheme and have the chance to escape from local trapping. In the case of only a few viruses, a significant magnitude of viral infections are needed to increase schemata and maintain the diversity in the population and "infectivity" should be high for these viruses ($V_{inf\ Rate}$ max =10, 100% viral infection rate). On the contrary a lot of viruses, i.e. V_{pop} = 10, immediately propagate efficient schemes supporting local search and avoiding local trapping, whilst fewer transductions are needed to update the scheme ($rate \downarrow V_{life}$ = -0.001).

Figures 5.5, 5.6, 5.7 and 5.8 illustrate the convergence results of MOVEGA for the parameter settings suggested by the axial points of the RSM-CCD experimental design. These results allow for the effect investigation in the form of a parametric study since the investigation can be conducted by examining the effect of variation of a single parameter on the result at a time while maintaining the same settings for the rest of parameters. Fig. 5.5 shows the effect of number of viruses in the virus population V_{pop} for V_{pop} =5 (400 function evaluations) and for V_{pop} =7 (500 function evaluations).
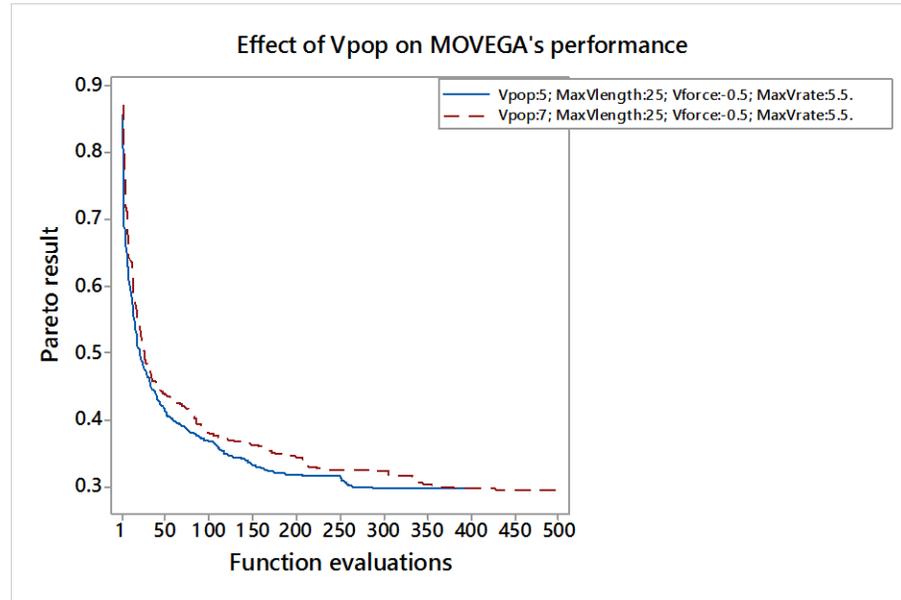
Figure 5.5: Effect of population of viruses (axial runs with 5 and 7 viruses-400 and 500 function evaluations) in the multi-objective Pareto result.

The result given in Figure 5.5 corresponds to 17th and 18th experimental runs (axial points) and clearly shows that by implementing a higher number of viruses it is improved. The Pareto result for V_{pop} =5 is 0.297933 whereas for V_{pop} =7 is 0.294029. However, the former result was obtained at 293 function evaluation out of 400 and the latter was obtained at 441 function evaluations out of 500. With reference to these indications it is asserted that the "near optimal" result of 0.297933 with V_{pop} =5 has been obtained at 73.25% of the total number of function evaluations whilst the lower "near optimal" result of 0.294029 with V_{pop} =7 has been obtained at 88.20% of the total number of function evaluations. The result of 0.294029 is 1.34% to that of 0.297933 which seems to be insignificant. However, by considering the individual criteria formulating the Pareto result it can be observed form 17th and 18th experimental runs that the result of 0.294029 corresponds to C_1 =0.025, C_2 =0.190 and C_3 =0.223 and the result of 0.297933 corresponds to C_1 =0.035, C_2 =0.104 and C_3 =0.277. Based on these outputs the former Pareto result (0.294029) outperforms the latter (0.297933) by reducing the result of C_1 at 28.57%, as well as the result of C_3 at 19.50%, yet, at the expense of C_2 criterion that results 45.26% worst.

Figure 5.6 shows the effect of viral chromosome string length $V_{strlength}$ max to the result of Pareto criterion for V_{pop} = 6 (450 function evaluations) as per the indications of the RSM-CCD design and its corresponding axial points.

138

Effect of MaxVlength on MOVEGA's performance

Legend:
— Vpop:6; MaxVlength:24; Vforce:-0.5; MaxVrate:5.5.
– – Vpop:6; MaxVlength:27; Vforce:-0.5; MaxVrate:5.5.

Y-axis: Pareto result
X-axis: Function evaluations
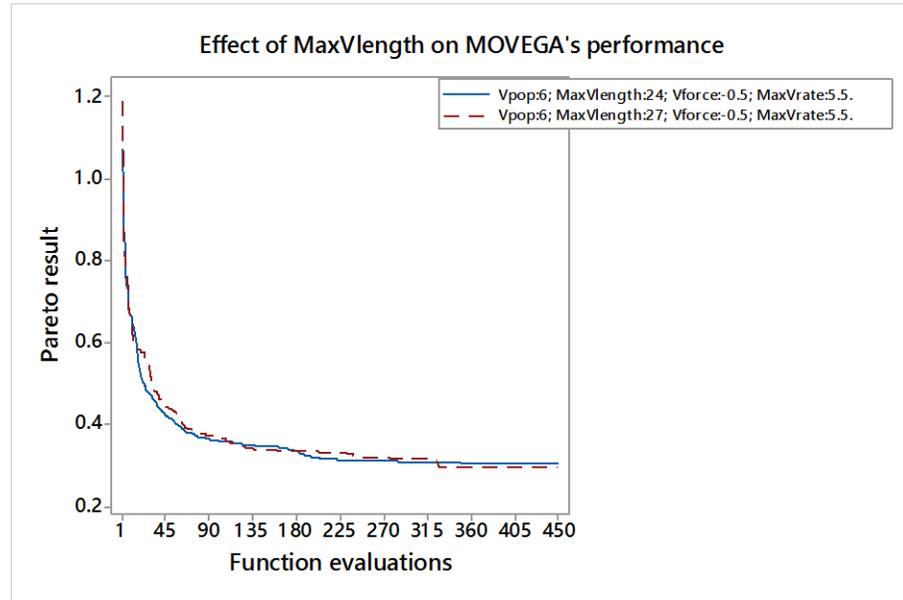
Figure 5.6: Effect of viral chromosome string length (number of bits) in the multi-objective Pareto result.

From Figure 5.6 and Table 5.3 (experimental runs 19 and 20) it is deduced that a higher variable number of bits in chromosome strings of viruses provides more beneficial schemes towards the convergence to an optimal result. The 19th experimental run results to a Pareto solution equal to 0.305419 obtained at 350 function evaluation whilst the 20th experimental run results to a solution equal to 0.296163 obtained at 341 function evaluation. A first observation at least when it comes to 450 function evaluations, (V_{pop} = 6) is that more bits in the chromosome string of a virus individual benefit both optimal result and convergence speed. The Pareto solution obtained using 27 bits in the chromosome strings of viruses is 2.96% better that the one obtained using 24 bits. As regards the individual criteria all three values for C_1, C_2 and C_3 result as better at the amounts of 6.25%, 13.04% and 1.42% respectively.

Figure 5.7 shows the effect of virus life reduction rate $rate \downarrow V_{life}$ to the result of Pareto criterion for V_{pop} = 6 (450 function evaluations) as per the indications of the RSM-CCD design and its corresponding axial points.
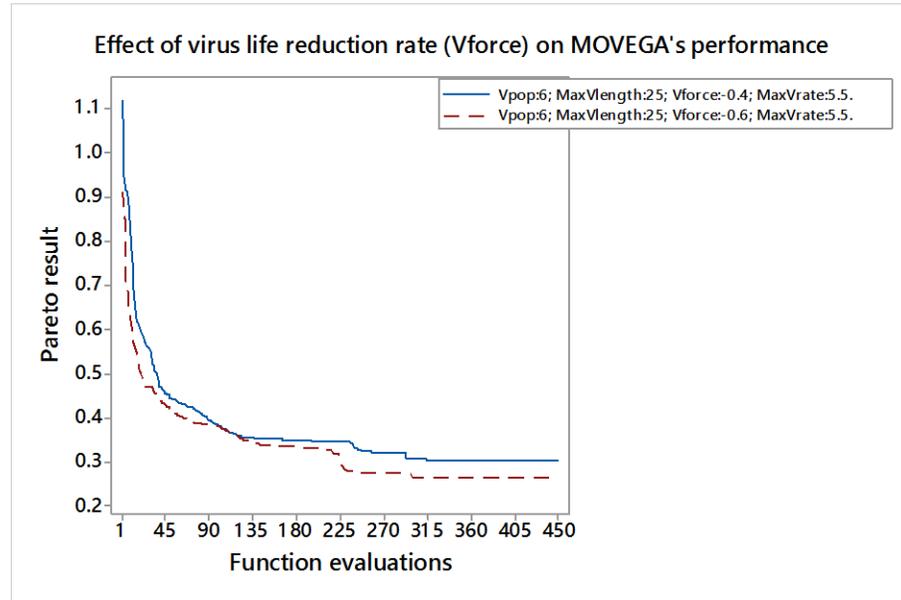
Figure 5.7: Effect of virus life reduction rate in the multi-objective Pareto result.

Figure 5.7 clearly illustrates that virus life reduction rate $rate \downarrow V_{life}$ is beneficial when it is at an intermediate level. The results for Pareto criterion correspond to 21st and 22nd experimental runs (Table 5.3). The result with the lower virus life reduction rate ($rate \downarrow V_{life}$=-0.4) gives a result for Pareto criterion equal to 0.304467 obtained at 340 function evaluation whereas the result with the higher life reduction rate ($rate \downarrow V_{life}$=-0.6) gives a result for Pareto criterion equal to 0.263750 obtained at 299 function evaluation. The gain obtained in terms of the "best" final point is equal to 13.49%. As regards the individual criteria two out of three values (C_2 and C_3) result as better at the amounts of 29.47% and 1.81% respectively whereas the value corresponding to C_1 criterion ends up as worst at the amount of 24.25%. At least for the 450 function evaluations (V_{pop} = 6 according to the RSM-CCD axial points) higher reduction rates for the life of virus individuals are advantageous for both lower final point and convergence speed as evident from Fig. 5.7. This implies that transduction should be prompted to operate to increase local search capabilities in the algorithm when it comes to a population of viruses half to that of the main population of candidate solutions. Another important observation based on Fig. 5.7 for the effect of $rate \downarrow V_{life}$ parameter is that a similar trend is exhibited throughout the entire convergence process for the settings, -0.4 and -0.6.

Figure 5.8 shows the effect of viral infection rate $V_{inf\ Rate}$ max to the result of Pareto criterion for V_{pop} = 6 (450 function evaluations) as per the indications of the RSM-CCD design and its corresponding axial points.
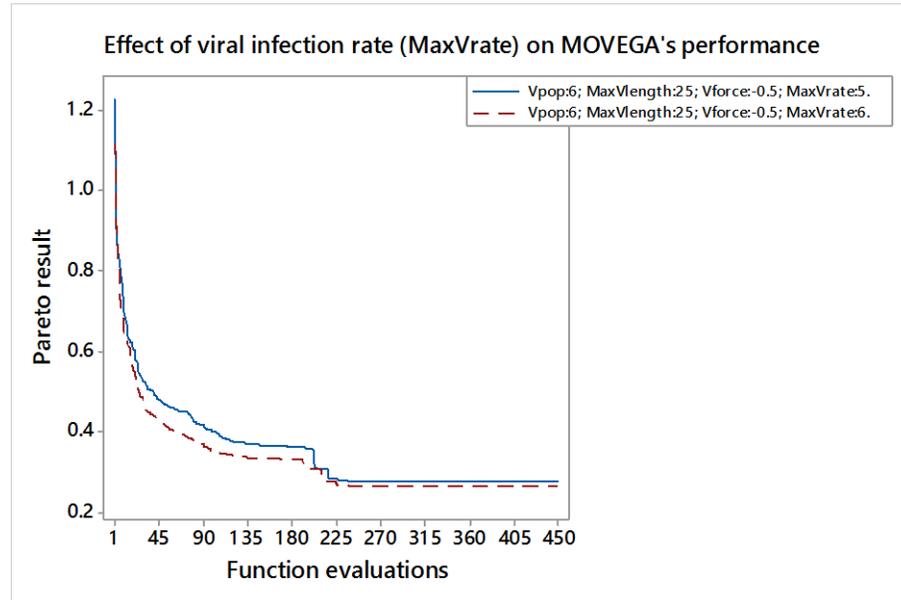
Figure 5.8: Effect of viral infection rate in the multi-objective Pareto result.

The two different viral infection rates for the same number of viruses (V_{pop} = 6, 450 function evaluations) and the same settings for the rest of algorithm-specific parameters suggest significant differences for both convergence speed as well as quality of the result. As it is suggested in Fig. 5.8 and Table 5.3 the increase of viral infection rate advantageously shifts the algorithm's behaviour in terms of convergence slope and final point whereas the effect seems to maintain its trend. For $V_{inf\,Rate}$ max = 5 (23th experimental run) the result corresponding to Pareto criterion equals to 0.272615 whereas for $V_{inf\,Rate}$ max = 6 (24rd experimental run) the result corresponding to Pareto criterion equals to 0.260985. Their percentage difference equals to 4.27%. As regards individual criteria, two out of three values (C_2 and C_3) exhibit better outputs (12.74% and 0.46% respectively) whereas the value corresponding to C_1 criterion occurs as 2.86% worst. The result obtained in 23rd experimental run for $V_{inf\,Rate}$ max = 5 was reached after 280 function evaluations whilst the one obtained in 24th experimental run for $V_{inf\,Rate}$ max = 6 was reached after 286 function evaluations. In general the results at this state for V_{pop} = 6, $V_{strlength}$ max = 25 and $rate \downarrow V_{life}$ =-0.5, no significant changes in terms of arithmetic magnitudes seem to be experienced thus the clear and concise effect cannot be determined. Nevertheless from Fig. 5.8 it is evident that $V_{inf\,Rate}$ max highly contributes to algorithm's general performance towards convergence.

To prove stability and repeatability of non-dominated solutions as well as the optimal final point the results obtained for the center points (replicates) of the RSM-CCD design were statistically examined.

Statistical analysis involved the investigation for the hypothesized difference between standard deviations and/or variances of two non-dominated solution-sets taken from the outputs of experimental results of center points (replicates). This investigation needs to be done owing to the stochastic nature of evolutionary strategies - such as GAs/EAs – and their randomness characteristic which might display them as unreliable optimisation techniques (Hoos and Stótzle 2004). This false impression for evolutionary strategies comes as a strong assumption based on the difference among results usually found when trying multiple independent algorithmic runs (Kirkpatrick, 1984). However, owing to genetic operators, stochastic techniques avoid premature stagnation to local optima as opposed to deterministic techniques. Thus, the statistical study for these results is necessary to explore repeatability and reliability of MOVEGA in the quality of results when the latter is executed iteratively for a finite number of experimental tests (i.e. as many as the center points) for the same settings of its algorithm-specific parameters. Figure 5.9 shows the convergence results of MOVEGA for the same parameter settings suggested by the replicates (center points in cube) of the RSM-CCD experimental design.
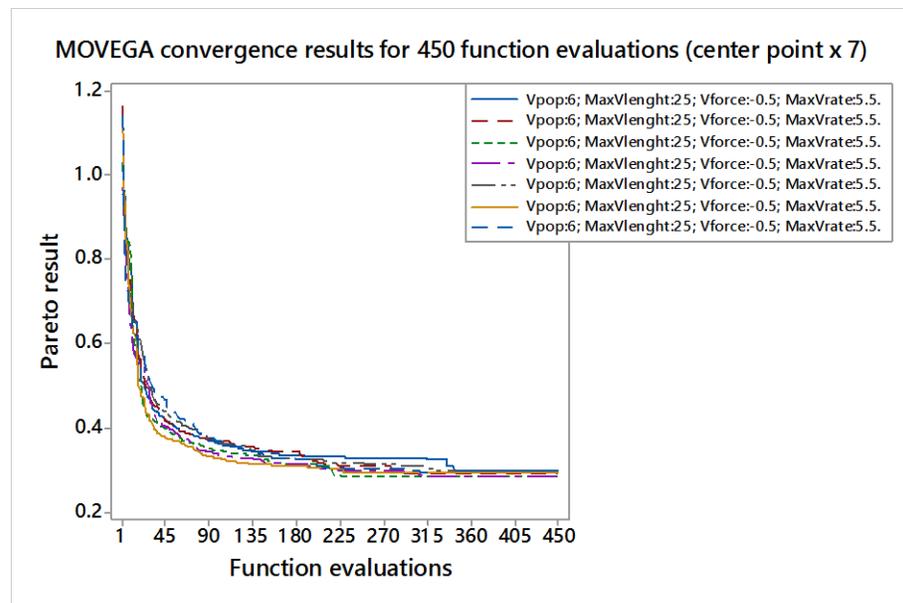
Figure 5.9: Convergence results for replicates – center points x 7 (6 viruses-450 function evaluations).

For the replicates of the RSM-CCD experiment 450 function evaluations have been determined (V_{pop} = 6) whilst the rest of parameters have the settings as per the RSM-CCD design dictates (Table. 5.3) for the experimental runs 25 to 31 ($V_{strlength}$ max = 25, $rate \downarrow V_{life}$ = - 0.5, $V_{inf\ Rate}$ max = 5.5). As evident from Fig. 5.9, six out of seven evaluations have quite similar trends in terms of the

convergence curve with the only exception observed to the first experimental test (25th experimental run according to Table 5.3).

In the statistical analysis conducted, the non-dominated solution sets where considered as independent sample data with the same magnitude (450 non-dominated solutions per set with reference to the number of function evaluations). Thereby, the non-dominated solutions where examined as pairs of two independent sets where the ratio of their standard deviations as well as variances is assumed to be equal to 1 according to their confidence intervals for the assessment of practical significance of the results. Statistical outputs were based on both Bonett's and Levene's methods to ensure reliability of results for any kind of data distribution (normal, non-normal, skewed and/or heavy-tailed, etc.). p -value interpretation is of immense importance so as to judge significance level which is denoted by alpha $\alpha = 0.05$. The value of alpha equal to 0.05 implies that a 5% risk may exist on having significant differences among results. As regards $\alpha = 0.05$ two assumptions are made. The former suggests that the ratio of standard deviations and/or variances is statistically significant ($p \leq \alpha = 0.05$) thus the null hypothesis H_0 is rejected and the conclusion is that the ratio of standard deviations (or variances) differs from the hypothesized one. Usually the hypothesized ratio R_H equals to 1 as a default value. The latter assumption suggests that the ratio of standard deviations or variances is not statistically significant ($p > \alpha = 0.05$) thus the null hypothesis H_0 is accepted and the conclusion is that there is enough evidence to support that difference the ratio of standard deviations (or variances) is statistically insignificant.

Figures 5.10a to 5.10f summarise outputs from the statistical tests conducted (2-variance ratio test) for the pairs of non-dominated solutions for the center points' replicates. In the outputs the confidence intervals for the ratios and the confidence intervals for variances are also presented. In the graphs the upper box refers to the estimated values (blue dots in the parallel lines) and confidence intervals for Bonett's and Levene's tests for the variance ratios with reference to the null hypothesis $H_0 = 1$ denoted by the red line vertical to the parallel lines. The lower box refers to the estimated values (blue dots in the parallel lines) and confidence intervals for Bonett's and Levene's tests for the variances referring to the null hypothesis $H_0 = 1$. As it is evident from Figs. 5.10a-f significant overlaps among the results of Bonett's test and those of Leven's test are observed suggesting similarity and insignificant differences among the results of non-dominated solutions. Even thought p -values for the tests do not agree in their magnitudes, they all above the critical level of

alpha ($p > \alpha = 0.05$). Table 5.4 gives the results from the statistical analysis with reference to Figures 5.10a to 5.10f.

(a) Test and CI for variances: 1stCP and 2ndCP
Ratio = 1 vs Ratio ≠ 1

(b) Test and CI for variances: 1stCP and 3rdCP
Ratio = 1 vs Ratio ≠ 1

(c) Test and CI for variances: 1stCP and 4thCP
Ratio = 1 vs Ratio ≠ 1

(d) Test and CI for variances: 1stCP and 5thCP
Ratio = 1 vs Ratio ≠ 1

(e) Test and CI for variances: 1stCP and 6thCP
Ratio = 1 vs Ratio ≠ 1

95% CI for Variance(1stCP) / Variance(6thCP)

Bonett
Levene

0.5 1.0 1.5 2.0 2.5 3.0

Bonett's Test P-Value 0.911
Levene's Test P-Value 0.208

95% CI for Variances

1stCP
6thCP

0.0050 0.0075 0.0100 0.0125 0.0150 0.0175

(f) Test and CI for variances: 1stCP and 7thCP
Ratio = 1 vs Ratio ≠ 1

95% CI for Variance(1stCP) / Variance(7thCP)

Bonett
Levene

0.50 0.75 1.00 1.25 1.50 1.75 2.00

Bonett's Test P-Value 0.906
Levene's Test P-Value 0.193

95% CI for Variances

1stCP
7thCP

0.0050 0.0075 0.0100 0.0125 0.0150

Figure 5.10: *2-variance* ratio test results for the pairs of non-dominated solutions of replicates.

Table 5.4: *2-variance* ratio test results for the pairs of non-dominated solutions of replicates.

Pairs	N:450	1stCP-2ndCP	1stCP-3rdCP	1stCP-4thCP	1stCP-5thCP	1stCP-6thCP	1stCP-7thCP
StDev		0.099-0.104	0.099-0.100	0.099-0.093	0.099-0.101	0.099-0.097	0.099-0.101
Variance (V)		0.010-0.011	0.010-0.010	0.010-0.009	0.010-0.010	0.010-0.009	0.010-0.010
95% CI for Variances		[0.006,0.015] [0.007,0.017]	[0.006,0.015] [0.006,0.016]	[0.006, 0.015] [0.006, 0.013]	[0.006,0.015] [0.007,0.015]	[0.006,0.015] [0.006,0.016]	[0.006,0.015] [0.007,0.016]
StDevs ratio		0.957	0.989	1.066	0.985	1.020	0.982
Variances ratio		0.917	0.979	1.137	0.970	1.040	0.964
CI for StDev ratio	*Bonett*	[0.689,1.330]	[0.710,1.389]	[0.770, 1.463]	[0.716,1.318]	[0.723,1.505]	[0.710,1.344]
	Levene	[0.641,1.079]	[0.708,1.233]	[0.762, 1.309]	[0.678,1.151]	[0.897,1.688]	[0.646,1.091]
CI for Variance ratio	*Bonett*	[0.475,0.770]	[0.505,1.929]	[0.593, 2.142]	[0.513,1.738]	[0.523,2.264]	[0.504,1.806]
	Levene	[0.411,1.164]	[0.501,1.519]	[0.581, 1.714]	[0.459,1.324]	[0.804,2.849]	[0.417,1.191]
p -value	*Bonett*	0.785	0.948	0.683	0.917	0.911	0.906
	Levene	0.167	0.623	0.996	0.362	0.208	0.193

The results reported so far present an image of the MOVEGA's functional behaviour in terms of its reliability and quality of results. However, to extract rigorous conclusions for these attributes a deeper examination of non-dominated solutions quality is needed as well as the investigation of main effects and interactions from the statistical analysis. The following sub-sections report the results interpreted with reference to the main effects / interactions from regression analysis as well as those obtained from the Pareto fronts corresponded to the RSM-CCD design presented.

5.3.2 Main effects and interactions

To ensure rigorous conclusions about the effects of MOVEGA's algorithm-specific parameters on its evaluation/optimisation performance regression analysis was performed to create a reliable model capable of providing statistical results to model the curvature of data and further optimise the critical response which is the 3D Pareto result per algorithm-specific parameter settings. Regression analysis was based on selections about the order of the terms, the cross predictors, the interactions as well as the method of forward selection of terms to exclude those that have no significant effect on the objective whilst keeping the influential ones. To study the main effects and interactions among algorithm-specific parameters, the main effects plot and the normal plot of standardized effects corresponding to ANOVA outputs where generated. The curvature was examined after employing the ANOVA model to fit the experimental results of the RSM-CCD experiment and generating the corresponding contour plots. The main effects plot and the normal plot for the standardized effects are shown in Figure 5.11. Such plots have already been used and interpreted also for the tool path parameters discussed in Chapter 3.
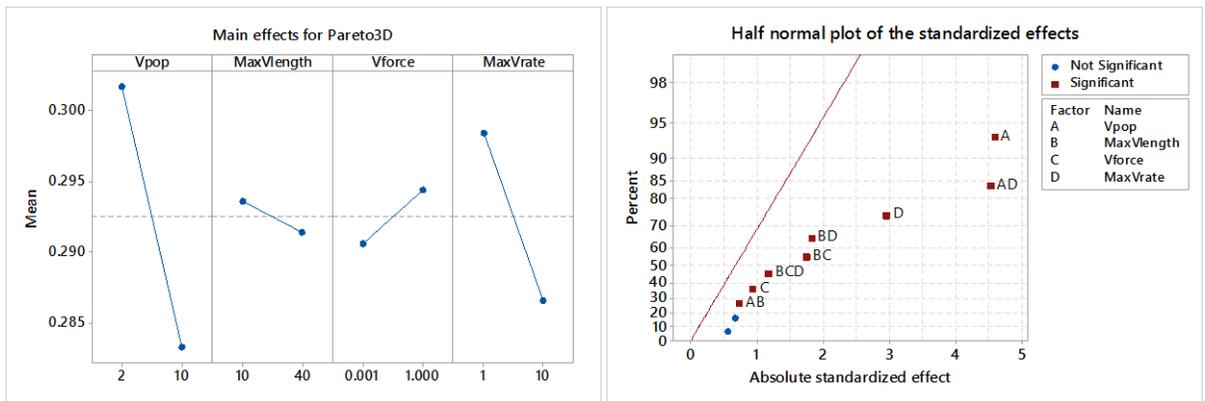
Figure 5.11: Main effects and standardized effects plots.

By examining the main effects plot it is observed that the mean of Pareto 3D is reduced for V_{pop} =10 compared to the usage of significantly fewer viruses (V_{pop} =2). $V_{strlength}$ max also reduces the mean of Pareto 3D if it is adjusted to its high level, 40 bits for the virus chromosome length. The virus life reduction rate $rate \downarrow V_{life}$ should be at its lower level to offer the opportunity for the viruses to infect a large number of individuals from the main population whereas viral infection rate $V_{inf\,Rate}$ max reduces the mean of Pareto 3D when set to its high level to facilitate local search. These observations are in agreement with those made during the analysis of experimental results for the RSM-CCD experimental design established. The half normal plot of the standardized effects indicates that

strong interactions exist among the MOVEGA's algorithm-specific parameters. Terms further from zero suggest statistically significant effects for the $a = 0.05$ significance level. V_{pop} and $V_{\inf Rate}$ max have the strongest effect as individual linear terms whereas their product V_{pop} x $V_{\inf Rate}$ max as well as $V_{strlength}$ max x $rate \downarrow V_{life}$ and $V_{strlength}$ max x $V_{\inf Rate}$ max exhibit the strongest effect among 2-way interactions. According to ANOVA, the contribution of linear terms, 2-way interactions, 3-way interactions and error in the model was found equal to 47.54%, 42.71%, 2.12% and 7.62% respectively. The highest significance in terms of influence on the "Pareto 3D" response was exhibited by V_{pop} ($p-$value = 0.006) followed by the interaction V_{pop} x $V_{\inf Rate}$ max ($p-$value = 0.006). $V_{\inf Rate}$ max alone has an important significance ($p-$value = 0.031). The overall significance of linear terms in the suggested model has a $p-$value equal to 0.030 whereas the overall significance of 2-way interactions has a $p-$value equal to 0.041. The suggested model is capable of explaining the variability of algorithm-specific parameters to the response of Pareto 3D at R^2 = 92.38%.

The curvature of experimental results of the RSM-CCD design is shown in Figure 5.12 in the contour plots. The scales for Pareto 3D value ranges accompanying the plots assist to identify the regions for recommending beneficial values of algorithm-specific parameters in order to minimise the Pareto 3D response. By examining Fig. 5.12a it is observed that 7 to 10 viruses are needed to optimise the final result referring to a main population of 10 individuals. At this point no significant effect seems to be noticeable for $V_{strlength}$ max . Fig. 5.12b agrees with Fig. 5.12a in terms of the number of viruses and further suggests that this number of viruses should have a low life reduction rate ($rate \downarrow V_{life}$). From Fig. 5.12c it is clear that the optimal results may be obtained with a large number of viruses and a maximised viral infection rate, yet, there is a large region where good results may be obtained for any number of viruses provided that a suitable value for maximum viral infection rate ($V_{\inf Rate}$ max) will be set since there is a region where the optimal result is dramatically deteriorated. Fig. 5.12d provides information for the simultaneous effect between maximum viral infection rate ($V_{\inf Rate}$ max) and virus life reduction rate ($rate \downarrow V_{life}$). It is clearly shown that an inversely proportional relation exists among $V_{\inf Rate}$ max and $rate \downarrow V_{life}$ parameters to balance virus life reduction rate and maximum infectivity according to the number of viruses in the virus population V_{pop} . Fig. 5.12e suggests that a number of bits in the chromosome strings of viruses ($V_{strlength}$ max) from 25 to 40

could be advantageous only if the virus life reduction rate ($rate \downarrow V_{life}$) is to be kept at low levels, i.e., from 10% to 35-40%. It is further suggested that a virus life reduction rate from 10% to 35% could contribute to the optimisation process when viruses operate as 25-bit to 40-bit substrings of their hosts. Finally, Fig. 5.12f suggests that the result is optimised for viruses that possess large substrings i.e. 35 to 40 bits, yet, with maximised infectivity.
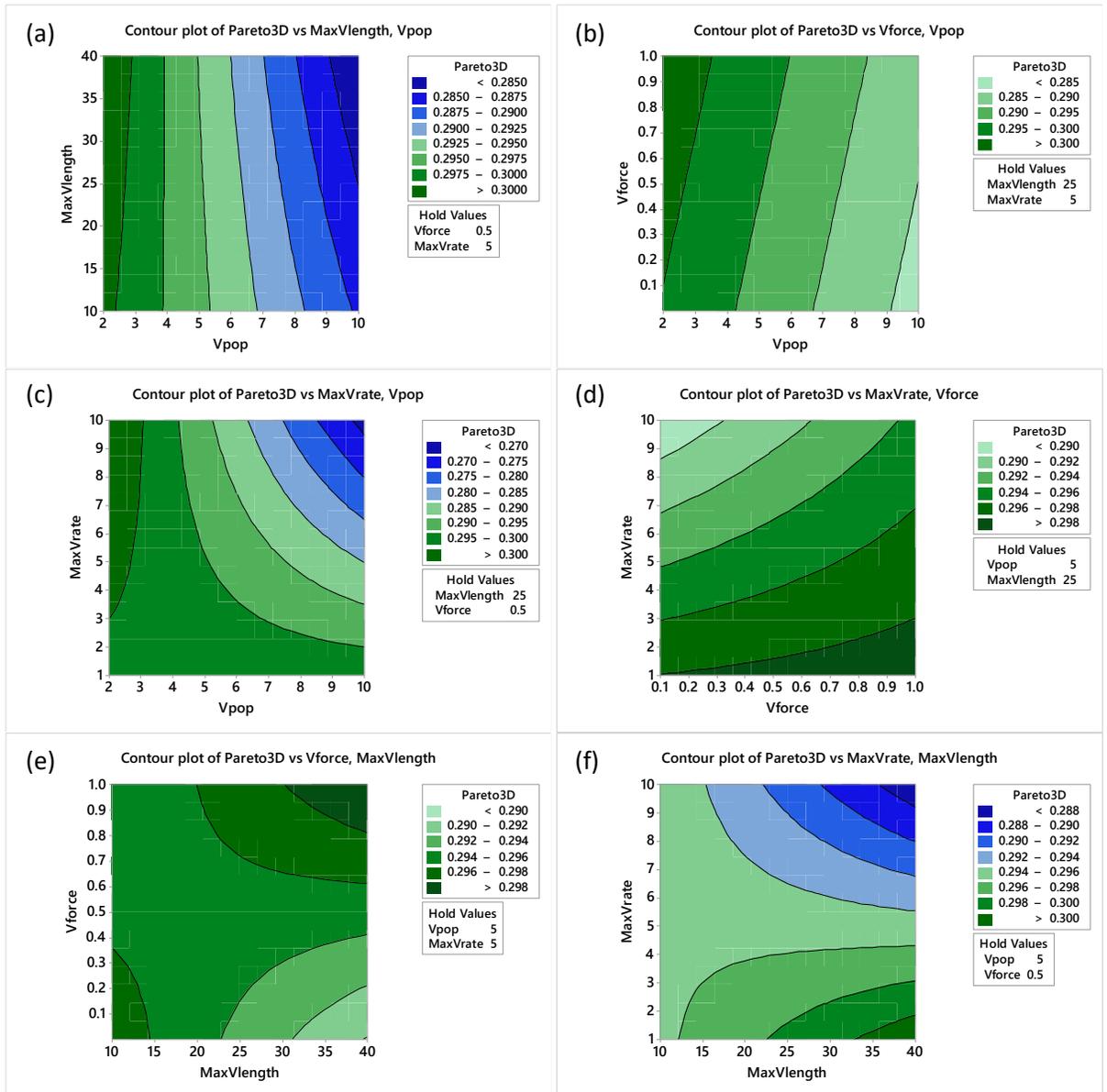
Figure 5.12: Contour plots of RSM-CCD for investigating the curvature of experimental results.

Diversity and spacing of non-dominated solutions of Pareto-optimal fronts have been questioned as essential indices so as to characterize the quality of results. The next subsection examines the properties of Pareto fronts obtained from the algorithmic evaluations according to the RSM-CCD design presented.

5.3.3 Diversity and spacing (coverage) of non-dominated Pareto-optimal solutions

The Pareto front is the 2-dimensional or 3-dimensional solution region where the non-dominated solutions found by solving a multi-objective optimisation problem are presented. The Pareto front is 2D or 3D depending on the objectives under examination whilst each objective is represented by an axis from those formulating the Pareto-front's solution region. If one criterion tends to improve (machining error), then the other tends to deteriorate (number of cutting points – machining time) and vice versa. In general, a Pareto front will consist of solutions that are optimal in the sense that no other solutions can be found capable of improving all criteria simultaneously, under the given (constrained) search space. Two indices are widely examined to characterize the non-dominated solutions using stochastic algorithms, diversity and spacing (Branke et al. 2001). The former index practically refers to the variety of non-dominated solutions with regard to their magnitudes in terms of the optimisation criteria whilst the latter refers to the spread (distribution) of solutions in the entire Pareto region. The general depiction, the practical use as well as the interpretation of results referring to a Pareto front are all dependent both on the nature of the optimisation problem at hand and the philosophy adopted to formulate it. Besides, this the main reason for capturing the researchers' interest to propose and develop new stochastic algorithms or improve current ones for solving a variety of engineering optimisation problems. This argument was initially supported by Wolpert and Macready (1997) and was based on the "No-Free Lunch" theorem that logically advocates that there is no optimisation algorithm capable of solving all optimisation problems.

For the specific problem of globally optimizing sculptured surface CNC machining tool paths, the optimal Pareto fronts where considered to be those that provide a sufficient number of non-dominated solutions that will have the closest distance from the reference point (Pareto front's origin) since all three criteria should be minimised. This introduces the trade-off among optimisation objectives indirectly (i.e. machining error and number of cutting points) as opposed to a variety of other engineering optimisation problems where some criteria ought to be maximised. An example may be derived by a common material removal process where surface roughness is minimised as a quality objective whilst material removal rate is maximised as a productivity objective. In the light of this, the Pareto fronts emerged from the results of the independent optimisation criteria values, were examined and evaluated with regard on the narrowest spacing between non-dominate solutions as well as the lowest diversity. The non-dominated solutions formulating the Pareto fronts refer to the RSM-CCD experimental runs as reported in Table 5.3. The Pareto fronts are depicted in Figure 5.13.

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31
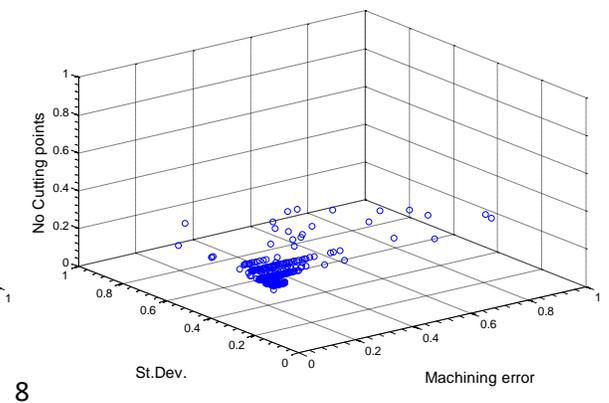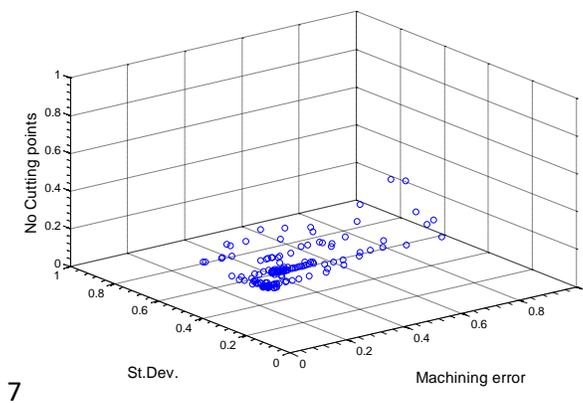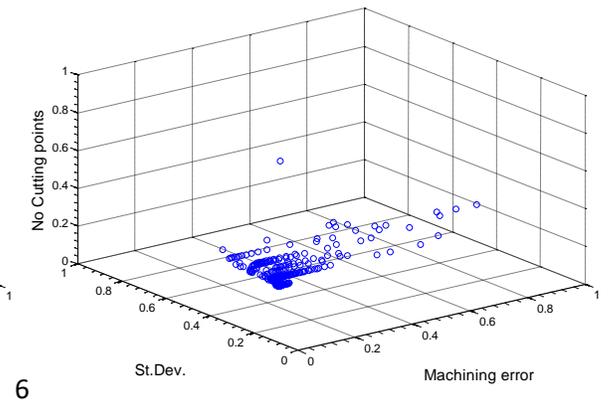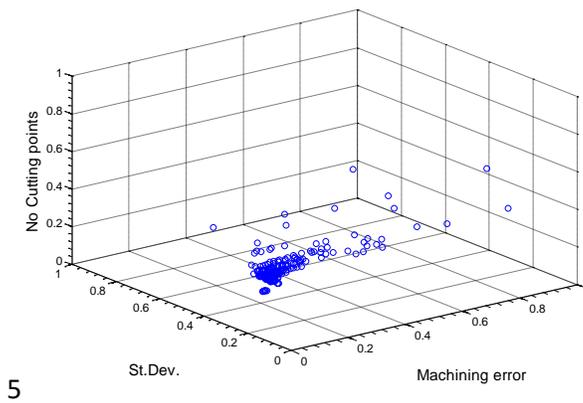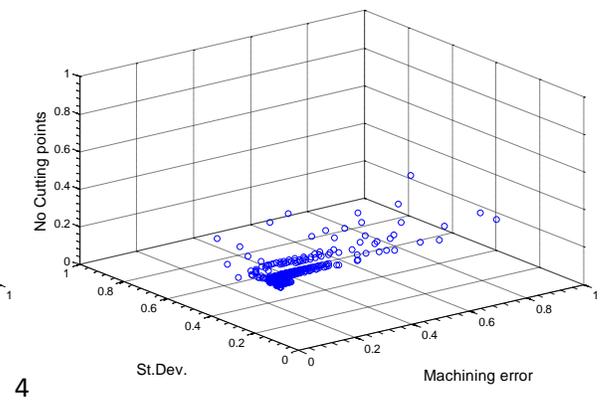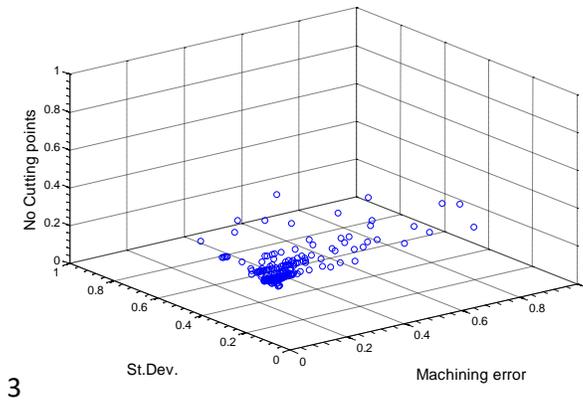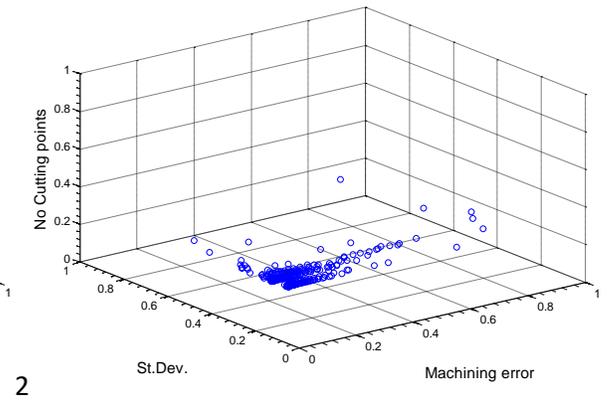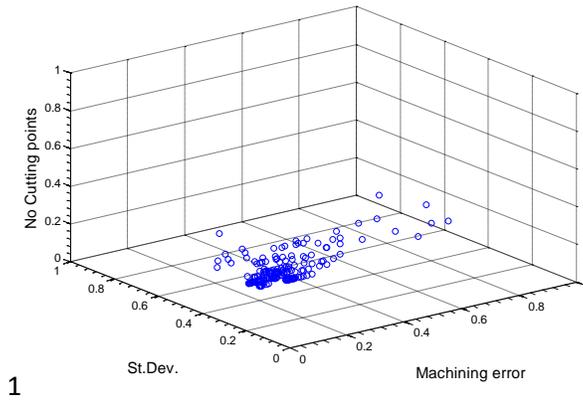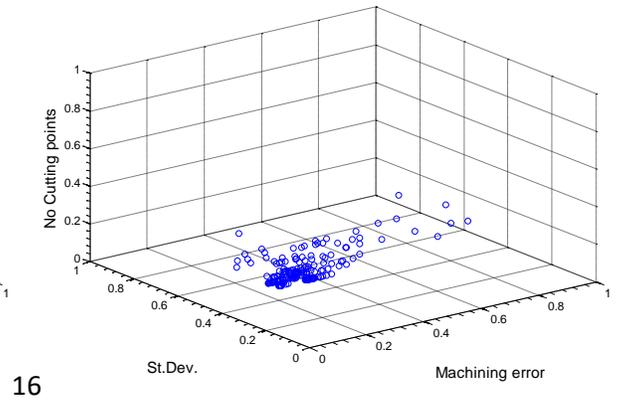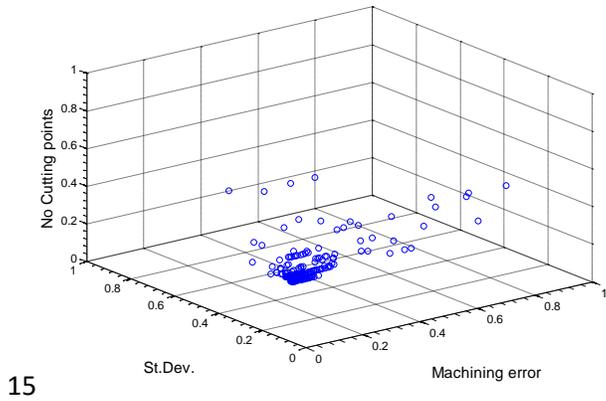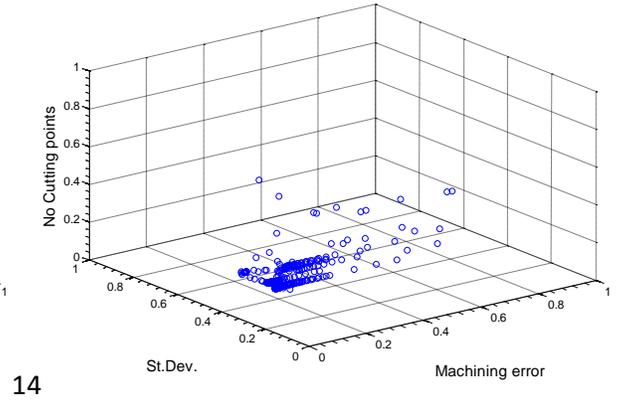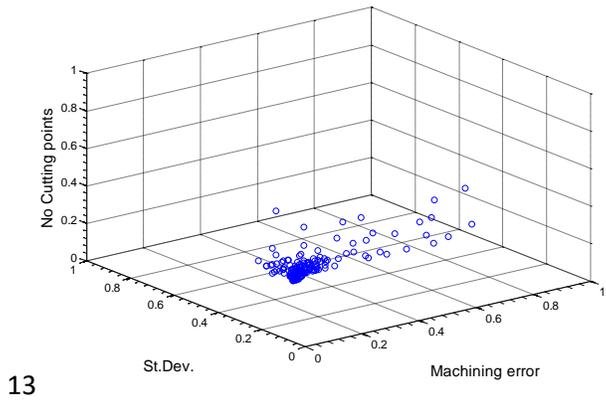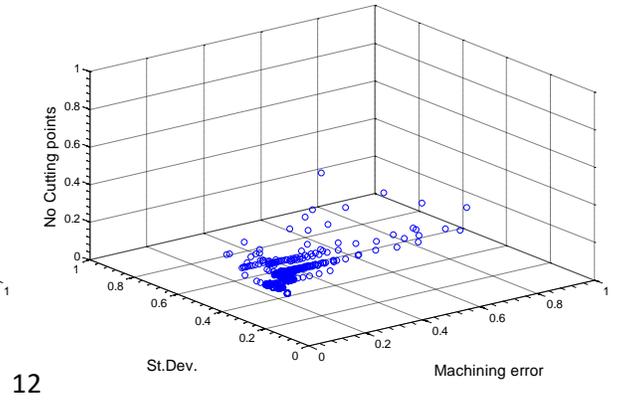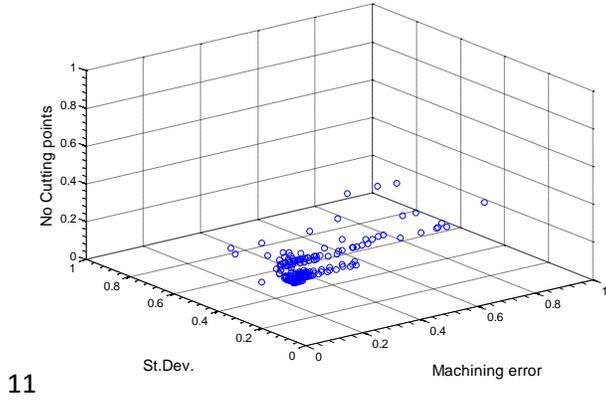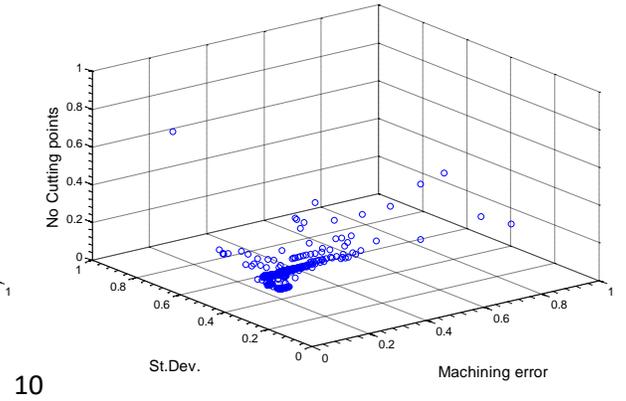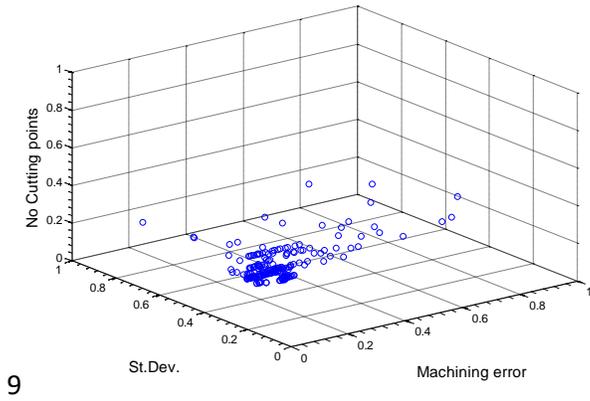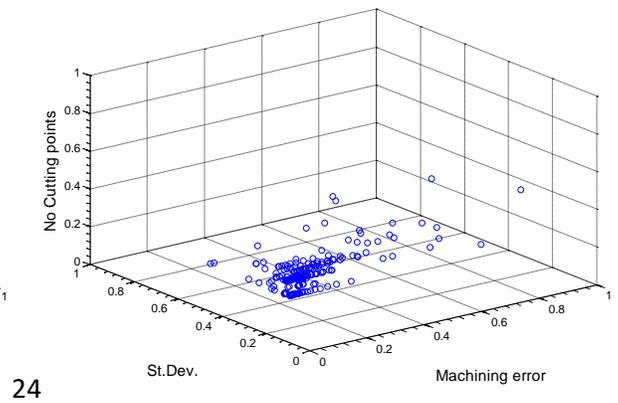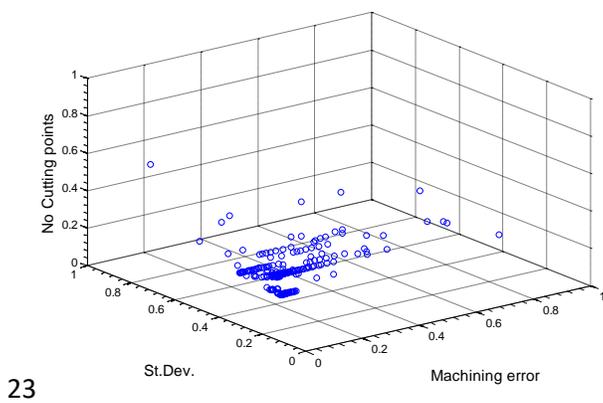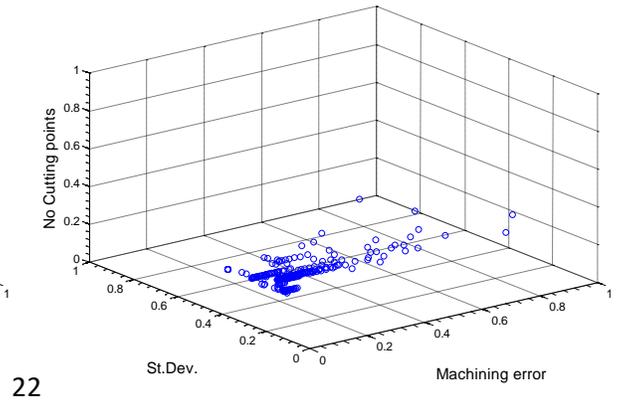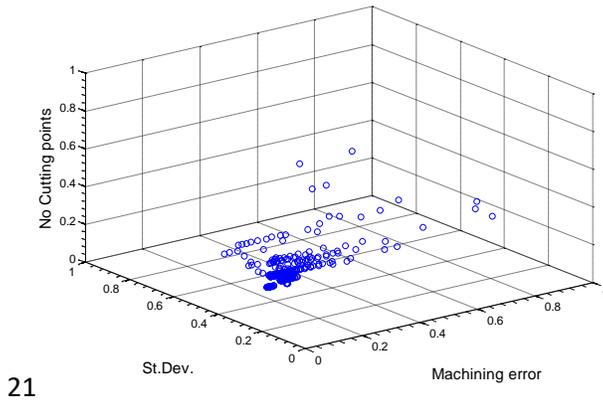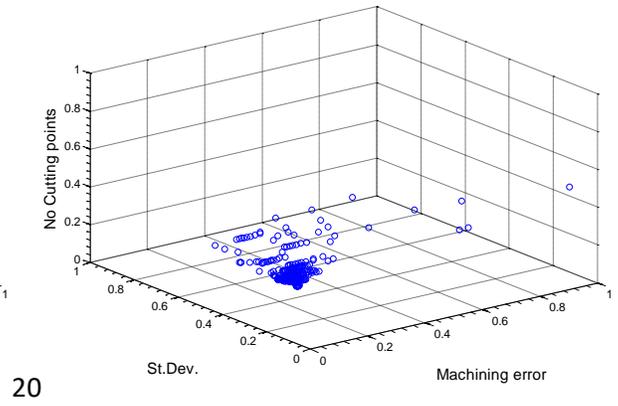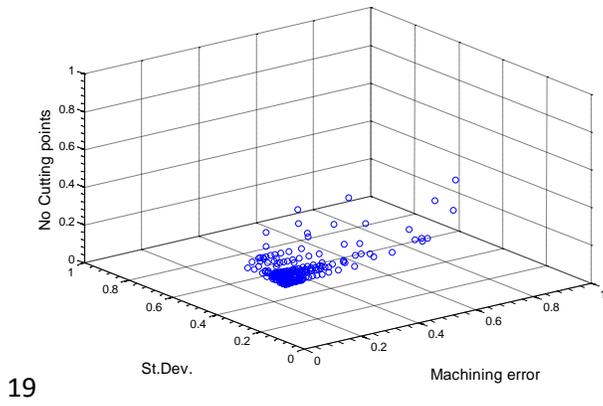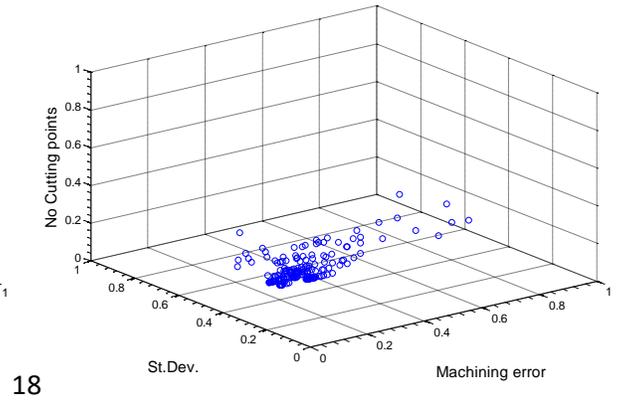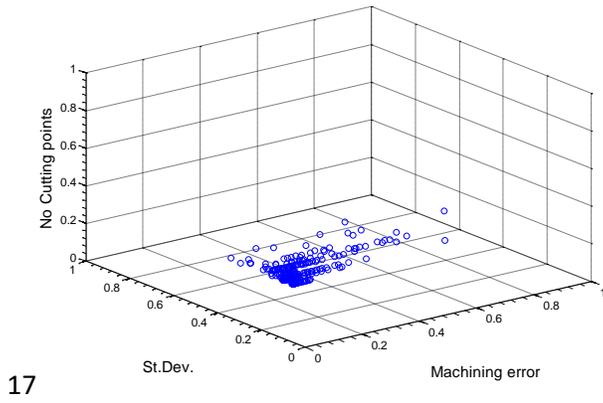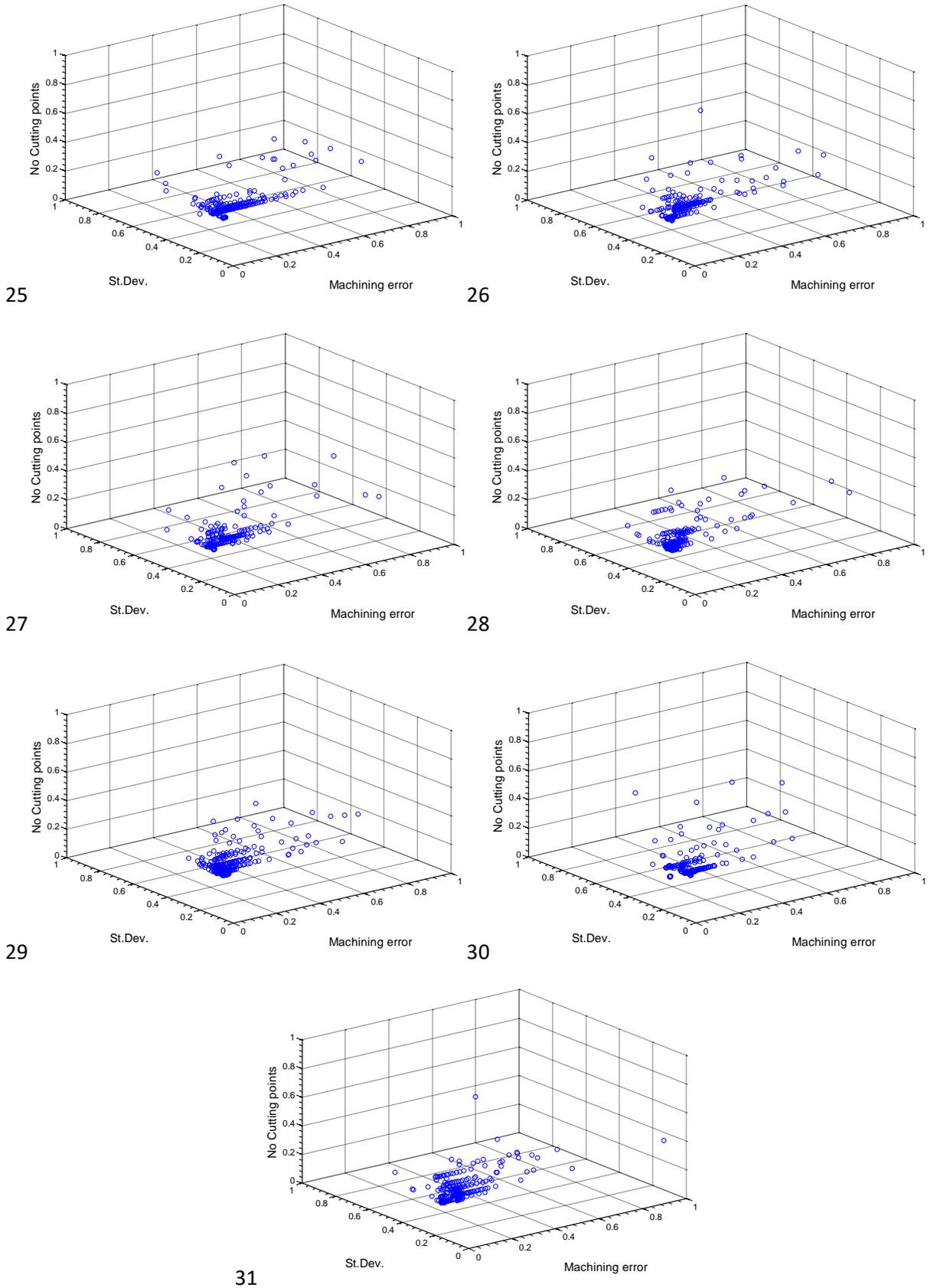
Figure 5.13: Pareto fronts of non-dominated solutions corresponding to the experiments of the RSM-CCD.

The axes of Pareto fronts correspond to the normalised values of the three criteria, machining error, standard deviation and number of cutting points. Two discrete regions can be distinguished to all Pareto fronts obtained. The first region is the region where some of the non-dominated solutions are scattered to satisfy only a single criterion or two out of three criteria and the second region is the main solution domain where all criteria are simultaneously satisfied at the best possible extend regarding the optimisation problem's trade-off.

Regardless of the results examined and interpreted so far, early observations from Fig. 5.13 suggest that Pareto fronts 17, 19, 28 and 29 seem to gather most of their non-dominated solutions close to their origins. For these results the number of function evaluations is 400 for Pareto front 17 (5 viruses) and 450 function evaluations for the rest Pareto fronts (6 viruses). Experiments corresponding to Pareto fronts 17 and 19 are axial points whereas experiments corresponding to Pareto fronts 28 and 29 are center points (replicates). By considering the number of function evaluations for the aforementioned Pareto fronts and the results reported for the effects of MOVEGA's algorithm-specific parameters on the final response (Pareto 3D criterion) it can be asserted that final recommendations for their optimal selections are expected to be approximately in the middle levels.

Diversity and spacing indices were computed using the latest relations found in the literature (Khalilpourazari and Khalilpourazary 2018, Khalilpourazari and Pasandideh 2018, Khalilpourazari and Mohammadi 2016). The relation adopted to compute diversity is given in Equation 5.5.

$$D = \sqrt{\sum_{j=1}^{j_{max}} \left(f_j^{\max} - f_j^{\min} \right)^2}$$

Eq. 5.5

where D the diversity of all j non-dominated solutions, corresponding to the Pareto front, $f_j^{\max}$ is the maximum objective function's value and $f_j^{\min}$ is the minimum objective function's value. Spacing has been computed for subsequent pairs of Pareto non-dominated solutions using Equations 5.6 and 5.7.

$$d_J = \sum_{j}^{j+1} \left| f_j - f_{j+1} \right|$$

Eq. 5.6

$$S = \sqrt{\sum_{j=1}^{n} \frac{\left(d_J - \bar{d} \right)^2}{n}}$$

Eq. 5.7

where, d_J is the spacing between two subsequent non-dominated solutions j, $j+1$, f_j the objective function's value for the j_{th} solution, f_{j+1} the objective function's value for the next solution with reference to the previous one (j), $\overline{d}$ is the average value of all subsequent spacing results, n is the total number of non-dominated solutions and S the general spacing result for the Pareto front's non-dominated solutions. The summation of the normalized diversity and normalized spacing was considered as the final metric to characterize the Pareto fronts. Normalization of the two indices was conducted by considering the largest value and dividing each result to that value. Table 5.5 summarizes the results obtained for diversity and spacing as well as the final metric which was their normalized sum $D+S$.

Table 5.5: Results for diversity and spacing corresponding to Pareto fronts of the algorithmic evaluations according to the RSM-CCD.

Pareto front index	D	d_J	$\overline{d}$	n	S	$D+S$
1	0.3709	0.0953	0.0032	194	0.4635	0.8344
2	0.5880	0.0710	0.0016	441	0.4363	1.0243
3	0.4272	0.0199	0.0034	202	0.4351	0.8623
4	0.7368	0.1243	0.0015	527	0.3890	1.1258
5	0.5983	0.1185	0.0040	226	0.6530	1.2514
6	0.6980	0.2257	0.0020	406	0.6000	1.2980
7	0.4063	0.0540	0.0046	155	0.5588	0.9651
8	0.5759	0.0136	0.0013	479	0.3009	0.8768
9	0.4530	0.0218	0.0035	206	0.4938	0.9468
10	1.0000	0.3132	0.0022	500	0.8270	1.8270
11	0.4394	0.0246	0.0034	209	0.4756	0.9150
12	0.8812	0.1767	0.0015	584	0.4477	1.3289
13	0.4064	0.0589	0.0043	168	0.6800	1.0864
14	0.6023	0.0182	0.0016	446	0.3203	0.9226
15	0.4790	0.0938	0.0040	195	0.5811	1.0600
16	0.6853	0.3003	0.0040	256	1.0008	1.6861
17	**0.4001**	**0.1353**	**0.0019**	**293**	**0.4189**	**0.8190**
18	0.9111	0.3115	0.0024	441	0.8304	1.7416
19	0.5992	0.1447	0.0022	350	0.4944	1.0936
20	0.6977	0.2597	0.0026	341	0.7615	1.4592
21	0.6465	0.1085	0.0024	340	0.4548	1.1013
22	0.4698	0.0374	0.0022	299	0.4451	0.9149
23	0.6968	0.2756	0.0034	280	0.8705	1.5674
24	0.6036	0.0388	0.0030	286	0.5925	1.1961
25	0.5854	0.0440	0.0020	362	0.3687	0.9541
26	0.7549	0.1617	0.0030	293	0.6022	1.3571
27	0.4775	0.0391	0.0032	233	0.5426	1.0201
28	0.5237	0.0272	0.0020	351	0.3546	0.8783
29	0.5159	0.0052	0.0019	344	0.3207	0.8366
30	0.5193	0.0263	0.0035	235	0.6287	1.1479
31	0.6288	0.0827	0.0028	308	0.6604	1.2893

According to Table 5.5 the Pareto front that exhibited the lowest final metric by simultaneously considering diversity and spacing is the one obtained for the non-dominated solutions of the 17th

RSM-CCD experimental run. For this particular Pareto front the score for the final metric which is the normalized sum $D+S$ is equal to 0.8190 with diversity D equal to 0.4001 and spacing S equal to 0.4189. For Pareto fronts 19, 28 and 29 the scores for the final metric is 1.0936, 0.8783 and 0.8366 respectively. These results are in good agreement with the depictions of non-dominated solutions for the aforementioned Pareto fronts. Indeed the 17th Pareto front is the one containing the narrowest region of scattered solutions whilst most of the non-dominated solutions are gathered very close to the Pareto front's origin. Similar indications are observed for the rest of Pareto fronts proving the consistency among the results derived from computations and the outputs illustrated on graphical representations.

5.3.3 Recommended algorithm-specific parameter settings and confirmation experiments

Response optimiser is a utility found to most known and commercially available statistical packages like the one used for formulating the RSM-CCD design and analyzing the related outputs. Optimisation plots or ramp diagrams show the effect of predicted responses under different experimental settings of parameters under investigation according to the model developed for fitting the data. Response optimiser was applied to the study to search for algorithm-specific parameter settings with near-optimal properties. With reference to the regression model created to fit the data given the experimental results several tests were performed to minimise the final response (Pareto 3D). During the initial setup of response optimisation the desirability function was selected for "minimisation" against "maximisation" and "target value" as well as best and worst values (0.251180 and 0.309630) according to the experimental results presented in Table 5.3. All variables where constrained to their corresponding parameter bounds: $2 \leq V_{pop} \leq 10$, $10 \leq V_{strlength} \max \leq 40$, $-0.001 \leq rate \downarrow V_{life} \leq 1.0$ and $10\% \leq V_{\inf Rate} \max \leq 100\%$. Low middle and high levels were tested as starting values for the response optimisation process to reduce the biased search towards local optimal points. Unfortunately, the response optimiser was not considered as the appropriate utility to contribute to the recommendation of near-optimal selections for algorithm-specific parameters since it did not perform a continuous search but a search limited to low-high levels and center point as potential candidates for parameter settings. After examining the diversity and spacing for all Pareto fronts obtained from the 31 experimental runs of the RSM-CCD design, the trend of parameter settings using the means was finally analysed to further examine their variation. Figure 5.14 illustrates the trend of algorithm-specific parameters on the mean of Pareto 3D as the main response when their corresponding levels are investigated for all design points, factorial and center (axial) points.
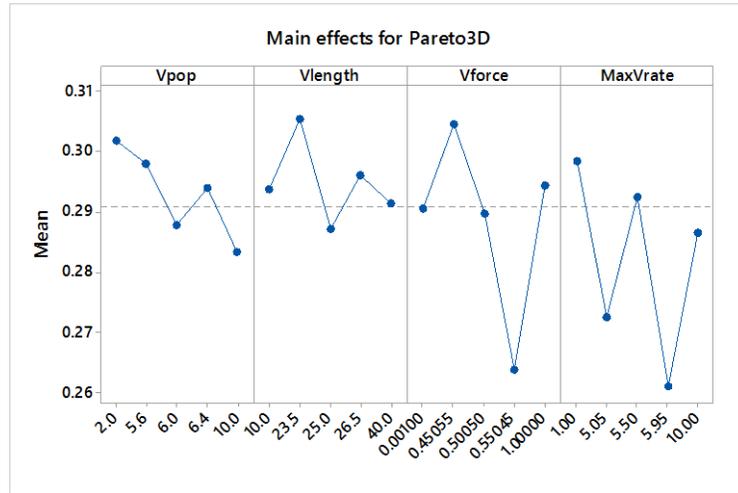
Figure 5.14: Main effects of algorithm-specific parameters by considering the entire RSM design points, factorial and axial.

The selection of the most appropriate algorithm-specific parameters was finally based on all experimental observations discussed above as well as the main effects and interactions of parameters on the response using the entire RSM-CCD design. The final values employed as the recommended ones for capturing the MOVEGA's full potentials were six viruses out of 10 candidates in the main population $\left(V_{pop} = \dfrac{6}{10} C_{pop} \right)$, maximum variable number of bits in the virus chromosome substring equal to 25 as a fraction of the 100-digit chromosome string of the individuals in the main population $\left(V_{strlength} \max = 25 = \dfrac{1}{4} C_{strlength} \max \right)$, maximum virus life reduction rate equal to -0.5 $\left(rate \downarrow V_{life} = -0.5 \right)$ and maximum infection rate equal to 70% of the maximum viral infectivity $\left(V_{\inf Rate} \max = 7 \right)$.

To evaluate the overall functional behaviour of MOVEGA as well as to quantify its contribution to the optimisation process, confirmation experiments were conducted for rigorous comparisons by adopting two different modes. The first mode was replicated 6 times using the recommended algorithm-specific parameters derived from the study presented above whilst the second mode was replicated six times using the same algorithm (MOVEGA), yet, without implementing its viral operators. As it is evident from Figure 5.15 the contribution of viral intelligence to the response optimisation for the multi-objective sculptured surface CNC machining problem is significant with reference to the indications given from the convergence results of MOVEGA versus the GA. According to the convergence trend referring to the first test, it seems that MOVEGA accelerates its

convergence quite early while maintaining a smooth transition towards the minimum recommended value. Both algorithms exhibit a fast convergence up to the 45th evaluation. Thereby, convergence gradually evolves from the 90th evaluation up to 320. A sudden descent (and eventually trapping) occurs for the GA to its recommended "optimum "while MOVEGA maintains its smooth minimisation path up to the lowest score obtained at 383rd evaluation. The second test shows that there is fairly similar convergence behaviour from both algorithms up to the 135th evaluation whilst from that point and further, GA converges and maintains the same value for its recommended "optimal" score up to the last evaluation. On the other hand, MOVEGA continues its smooth convergence from 135th evaluation up to 225th where its lowest score has been reached and is significantly lower than the one that the GA recommends for the same test. The third test suggests that the GA does not only seem to exhibit a faster converge rate than MOVEGA even from early evaluations but is also favored by the fact that it starts its convergence from a lower score than that from which MOVEGA starts. However, there is a decrease in convergence rate observed to the 100th evaluation up to 320th for both algorithms, yet, MOVEGA has clearly converged to lower scores than GA's for these evaluations. Finally, from the 320th evaluation, there is a sudden convergence from both algorithms with MOVEGA pointing to a lower final score. As far as the fourth test is concerned, it is evident that the convergence rate in the first evaluations for both algorithms is almost the same up to the 190th evaluation. Further on, MOVEGA not only converges faster than GA but also achieves a much lower final score for this test. For the fifth test, similar convergence attributes are observed from both algorithms with MOVEGA to minimise yet again the result in contrast to GA. Similarities in terms of convergence rate variations are observed to the same number of successful evaluations for both algorithms. The sixth test reports almost identical behaviour regarding the minimisation path as well as convergence rate for both algorithms up to 315th evaluation. Even though following evaluations do not see any improvement for GA, MOVEGA gives the impression of escaping from that local "minimum" and continues its convergence until the final result.

From the results reported for the aforementioned algorithmic tests, it can be deduced that MOVEGA's success to beneficial convergence characteristics as well as to final recommended values, is obviously due to the robust schemata formulated by viral operators (nvMOGA). It is worth mentioning that the computational cost is the same in the case of MOVEGA compared to GA since the latter was tested for a total of 15 generations (450 evaluations - see Eq. 5.3 and 5.4) for the sake of rigorous comparisons between MOVEGA and GA.
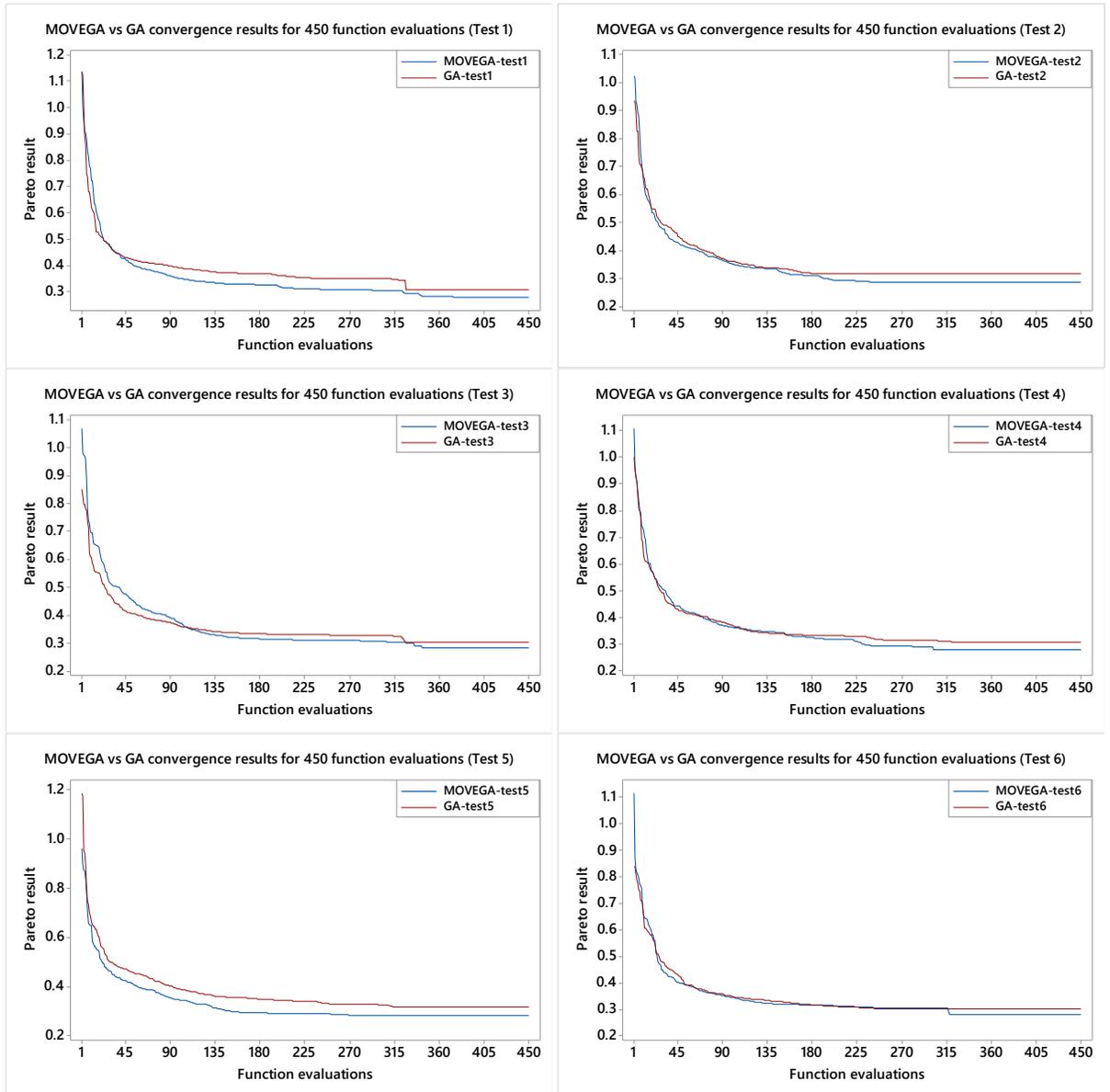
Figure 5.15: Convergence results corresponding to confirmation experiments for the optimal selection of algorithm-specific parameters.

The non-dominated solutions for MOVEGA and GA with regard to the confirmation tests conducted were examined through their corresponded Pareto fronts. The non-dominated solution closest to the origin of a Pareto front is considered as the optimal one and is expected to satisfy all three contradictory objectives. To characterise the Pareto fronts for the confirmation tests, Eq. 5.5, 5.6 and 5.7 were adopted, as presented above. Figures 5.16 and 5.17 illustrate the Pareto fronts with reference to the confirmation tests for MOVEGA and GA respectively and Tables 5.6, 5.7 summarise the results.
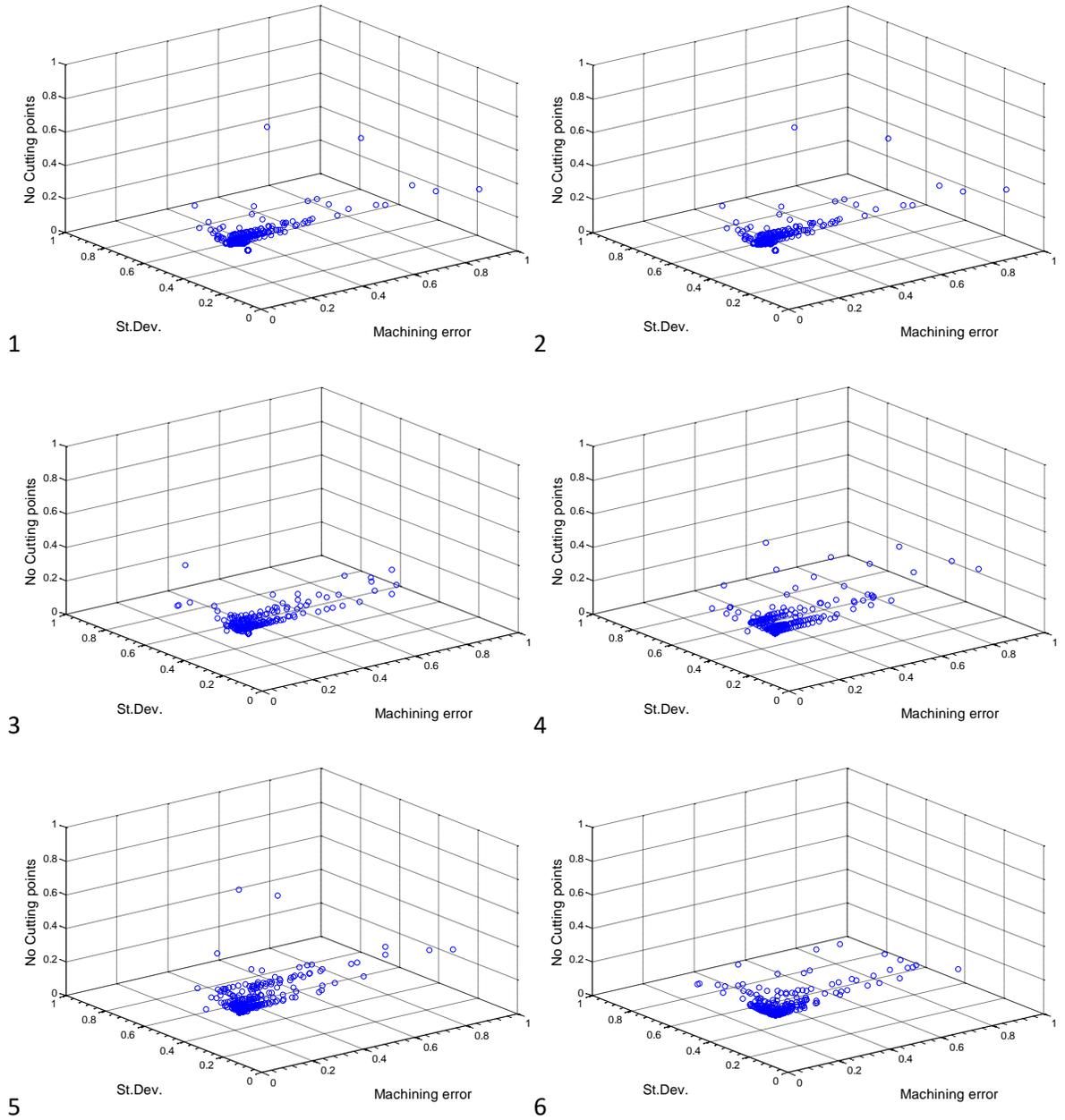
Figure 5.16: Pareto fronts of non-dominated solutions (MOVEGA) corresponding to confirmation experiments for the optimal selection of algorithm-specific parameters.

Table 5.6: Diversity and spacing results for evaluating non-dominated solutions of Pareto fronts obtained by MOVEGA.

| a/a | Performance evaluation indices for Pareto fronts | | | | | |
	D	d_J	$\overline{d}$	n	S	$D+S$
MOVEGA confirmation experiments						
1	9.4457	0.0121	0.0025	328.0000	0.0121	1.8027
2	5.5707	0.0062	0.0031	198.0000	0.0093	1.2172
3	6.0131	0.0226	0.0017	327.0000	0.0066	1.0695
4	7.7262	0.0492	0.0021	332.0000	0.0075	1.3053
5	9.8579	0.0097	0.0027	326.0000	0.0143	2.0017
6	5.2829	0.0033	0.0021	259.0000	0.0058	0.9397
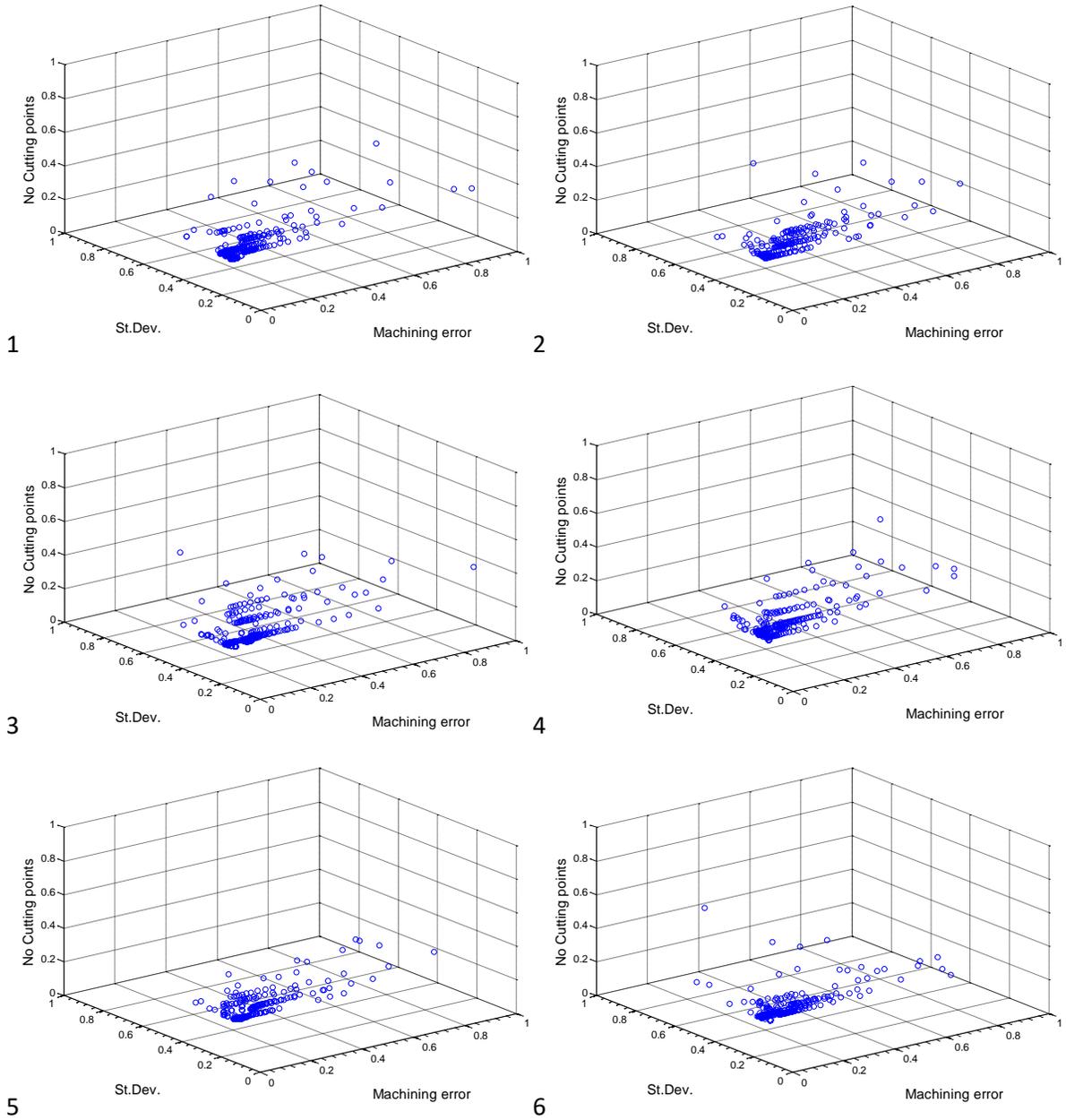
Figure 5.17: Pareto fronts of non-dominated solutions (GA) corresponding to confirmation experiments for the optimal selection of algorithm-specific parameters.

Table 5.7: Diversity and spacing results for evaluating non-dominated solutions of Pareto fronts obtained by GA.

a/a	Performance evaluation indices for Pareto fronts					
	D	d_J	$\overline{d}$	n	S	$D+S$
GA comparative experiments						
1	9.9910	0.1370	0.0022	383.0000	0.0095	1.6324
2	7.3918	0.0100	0.0027	271.0000	0.0089	1.3273
3	8.9459	0.0904	0.0023	349.0000	0.0092	1.5023
4	8.8414	0.1468	0.0028	302.0000	0.0108	1.5991
5	6.6798	0.0688	0.0025	269.0000	0.0091	1.2743
6	8.9160	0.2438	0.0026	318.0000	0.0151	1.8910

The normalised minimum final metric $D+S$ is 0.9397 and 1.2743 for MOVEGA and GA respectively and with reference to the figures depicting the Pareto fronts it is clear that MOVEGA clusters more non-dominated solutions closer to the origins than GA. Through their phenotype, the optimal "non-dominated" solutions referring to all Pareto fronts contain and the optimal parameter values for determining the 5-axis finishing tool path of the benchmark sculptured model used for designing the experiments of the current study so as to investigate the MOVEGA's functional behaviour and its algorithm-specific parameter effects. The parameters for the tool path corresponding to the optimal "non-dominated" solutions are given in Table 5.8.

Table 5.8: Recommended 5-axis tool path parameter values by MOVEGA and GA.

| a/a | Tool path parameters | | | | | Optimisation criteria | | | |
	Tool	Stepover	Lead angle	Tilt angle	MaxDstep	Pareto3D	C_1	C_2	C_3
MOVEGA confirmation experiments									
1	#2	44.147	10.261	2.827	6.062	0.280567	0.0290	0.174218	0.218
2	#2	44.489	10.113	0.773	7.265	0.287282	0.0364	0.184701	0.217
3	#2	44.162	14.971	4.095	6.78	0.283043	0.0303	0.167679	0.226
4	#2	44.066	10.485	0.859	4.433	0.278437	0.0348	0.168384	0.219
5	#2	44.244	15.455	1.462	5.378	0.284258	0.0536	0.170555	0.221
6	#2	44.579	13.673	2.916	5.132	0.281614	0.0476	0.174323	0.216
GA comparative experiments									
1	#2	42.741	15.813	0.668	4.421	0.309707	0.0486	0.129732	0.277
2	#2	38.356	14.176	0.210	3.282	0.317257	0.0500	0.118326	0.290
3	#2	39.464	12.820	4.140	4.973	0.304135	0.0478	0.131569	0.270
4	#2	37.913	15.695	2.568	4.497	0.307393	0.0337	0.11268	0.284
5	#2	40.661	13.415	3.836	5.507	0.318537	0.0325	0.155671	0.276
6	#2	38.259	10.439	4.321	6.839	0.289975	0.0348	0.074775	0.278

As shown in Table 5.8, all the solutions recommended by both algorithms are accompanied by the selections of the filleted end-mill (#2) against the flat end-mill (#1). For MOVEGA, all solutions proposed have a radial cutting tool engagement (stepover) over 44% as a percentage of the cutting tool's diameter, thus, resulting in reduced machining time. Similarly, for GA, only 33% of the solutions proposed (2 of 6) have stepover values over 40% as a percentage of the cutting tool's diameter. In general, the average values of recommended values for the filleted end-mill (#2) are 44.281% (7.085 mm) and 39.566% (6.330 mm) for MOVEGA and GA respectively, exhibiting an increase in productivity with the use MOVEGA as opposed to GA. The recommended outputs for lead angle span a range from 10.113° to 15.455° for MOVEGA, while for GA span from 10.439° to 15.813°. In combination to the values recommended for tilt angle ranging from 0.773° to 4.095° and from 0.210° to 4.321° for MOVEGA and GA respectively, it is derived that compared to GA, MOVEGA enlarges more the machining strip width (MSW) while maintaining low scallop heights. Maximum discretisation step parameter (forward step) spans from 4.433 mm to 7.265 mm and from 3.282 mm to 6.839 mm for MOVEGA and GA respectively. MOVEGA seems to promote productivity as it is

evident from the recommended tool path parameter values for stepover and forward step, while maintaining machining error at low levels without adversely affecting it. These observations are also indicated by the results for the individual criteria C_1, C_2 and C_3.

When it comes to the individual criteria, MOVEGA seems to direct its emphasis on optimising 2 out of 3 goals (machining error and number of cutting points, C_1 and C_3) while GA emphasizes mostly on optimizing machining error distribution C_2. Therefore the MOVEGA addresses more effectively the sculptured surface CNC machining optimisation problem compared to GA, according to the antagonising nature of the three criteria. The behaviour of MOVEGA regarding the emphasis given to machining error and number of cutting points against machining error distribution occurs due to the similar nature of these two objectives. When it comes to finishing, it is essential that machining error itself should be the criterion to emphasize to, since its distribution could be presented as a magnitude proportional to its own magnitude. On the contrary roughing operation may prioritize other criteria, i.e. first the cutting force variation exerted to cutting tool's edges and then the absolute magnitude of cutting force itself, in this respect. These "preference" trends of both algorithms as regards the individual criteria are illustrated in Figures 5.18, 5.19 and 5.20, where the distributions of values obtained for the different states as the optimisation evolves, are shown.

Figure 5.18 illustrates the evolution trend of obtained solutions referring to C_1 criterion by taking into account the effect of the rest criteria as well. A first observation is that compared to GA, the MOVEGA minimises further the result of C_1 criterion. A profound difference between the two algorithms in terms of the final score is observed in test 5 where the GA happens to minimise further the result of C_1 criterion, yet, a set of solutions has already been found in earlier generations for MOVEGA where their indications suggest just as low results as those obtained by GA in last evaluations. A significant observation is that partial distributions of values obtained for C_1 criterion seem to be shifted to evaluations with lower indices that those referring to the same results obtained by the GA. This shift is clearer and more profound in tests 1 and 4. This substantiates the faster convergence speed as well as robustness on the results of C_1 criterion.
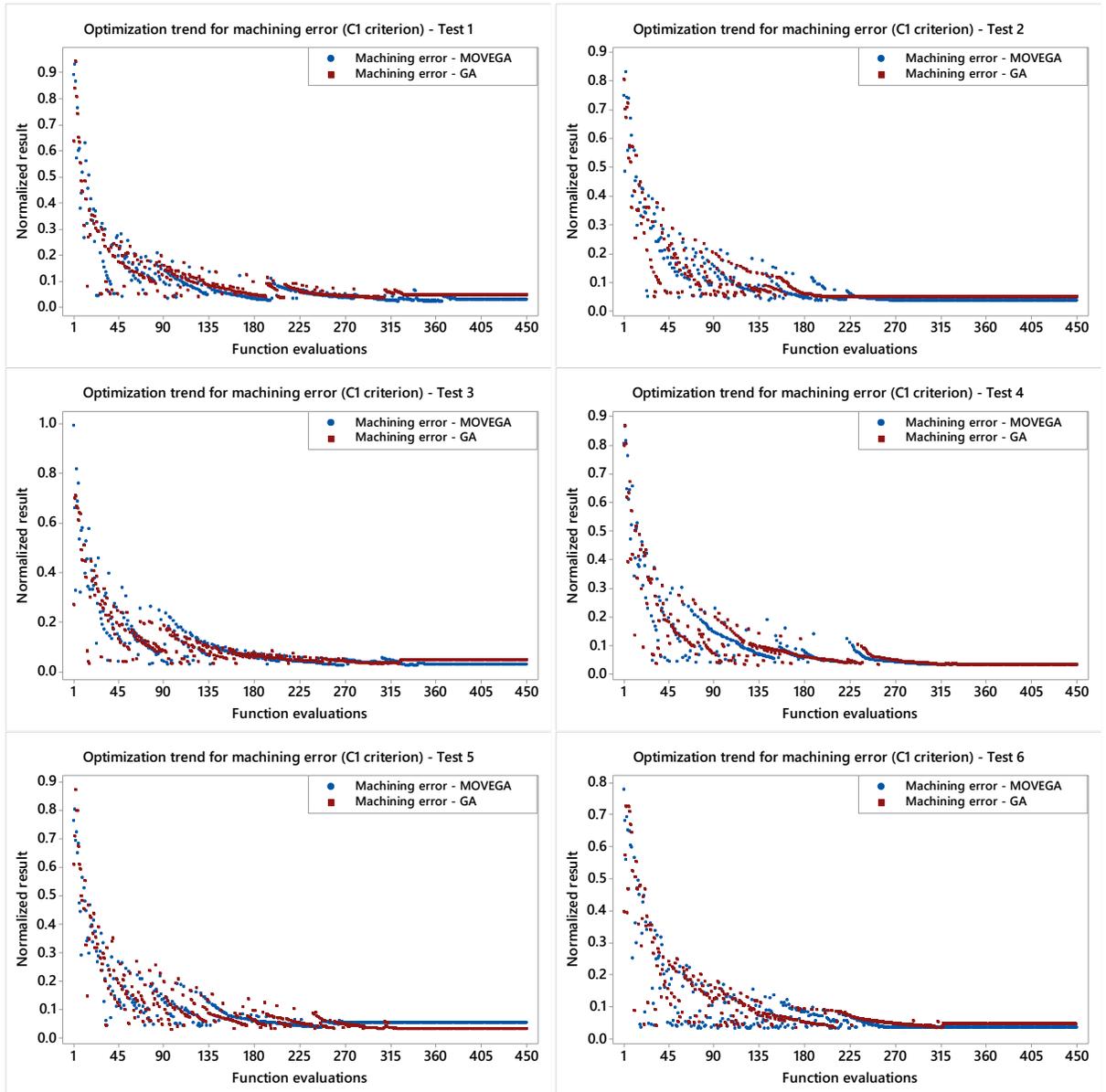
Figure 5.18: Distribution of individual values for "machining error" criterion (C_1) during evolutionary optimisation.

Figure 5.19 illustrates the evolution trend of obtained solutions referring to C_2 criterion by taking into account the effect of criteria C_1 and C_3 as well. Even though the trends imply a superior performance from GA's side against MOVEGA for this particular criterion, MOVEGA has also reached the same optimal solutions. In fact, tests 1 and 5 report some solutions obtained by MOVEGA, very close to those obtained by GA while in test 3 MOVEGA manages to outperform GA despite its inherent emphasis to the rest criteria. By considering all six confirmation tests, only the 2nd and the 4th exhibit a clear dominance of GA over MOVEGA for C_2 criterion.
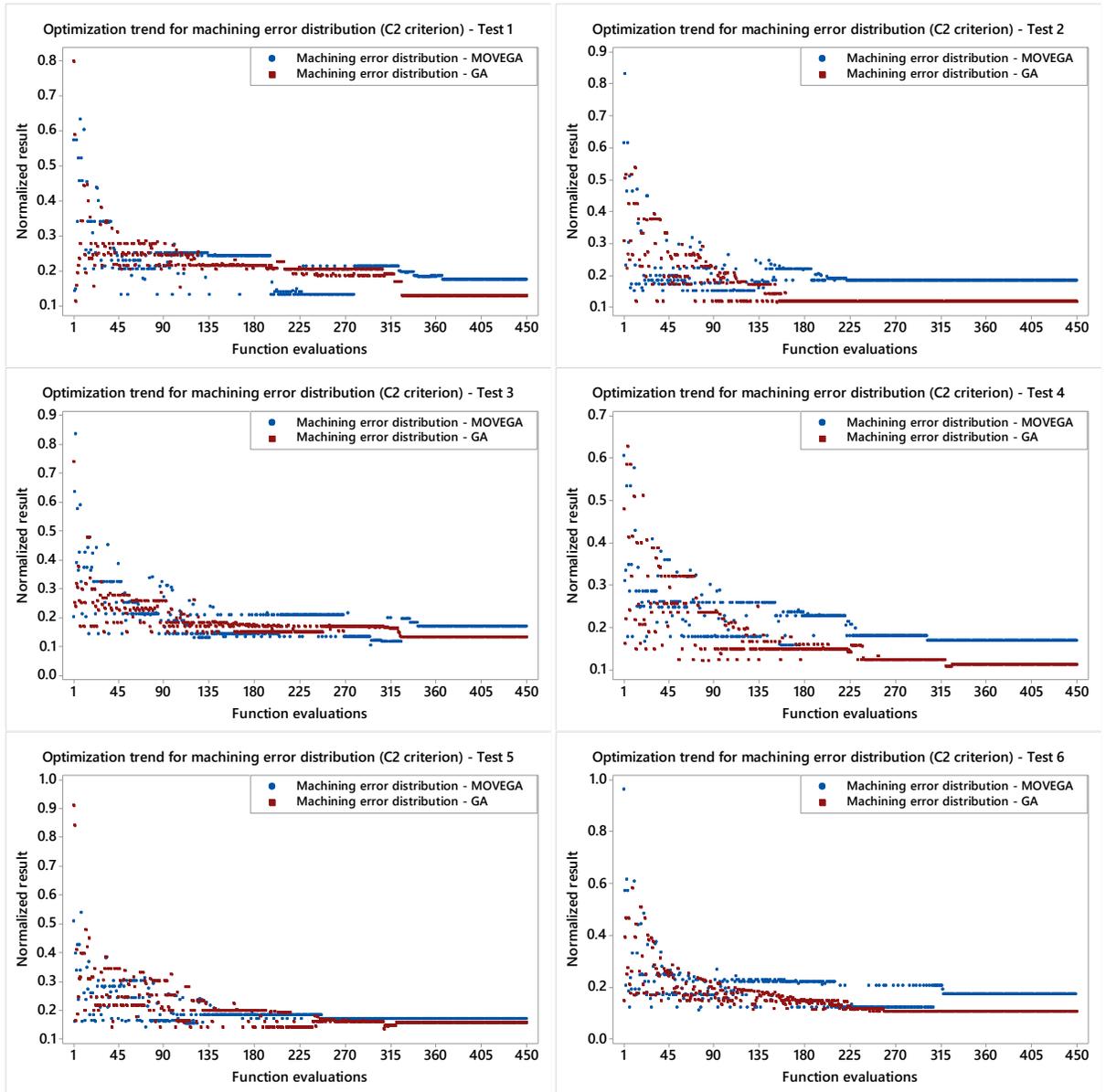
Figure 5.19: Distribution of individual values for "machining error deviation" criterion (C_2) during evolutionary optimisation.

Figure 5.20 illustrates the evolution trend of obtained solutions referring to C_3 criterion by taking into account the effect of criteria C_1 and C_2 as well. As it is evident from the six distribution trends referring to C_3 criterion the MOVEGA completely dominates over the GA with significant differences in the results obtained for the number of cutting points. As regards the GA, advantageous results for C_3 criterion are only reported in test 4. Test 3 reports a single advantageous result obtained by GA near 100th evaluation for number of cutting points and test 5 reports another one near 90th evaluation.
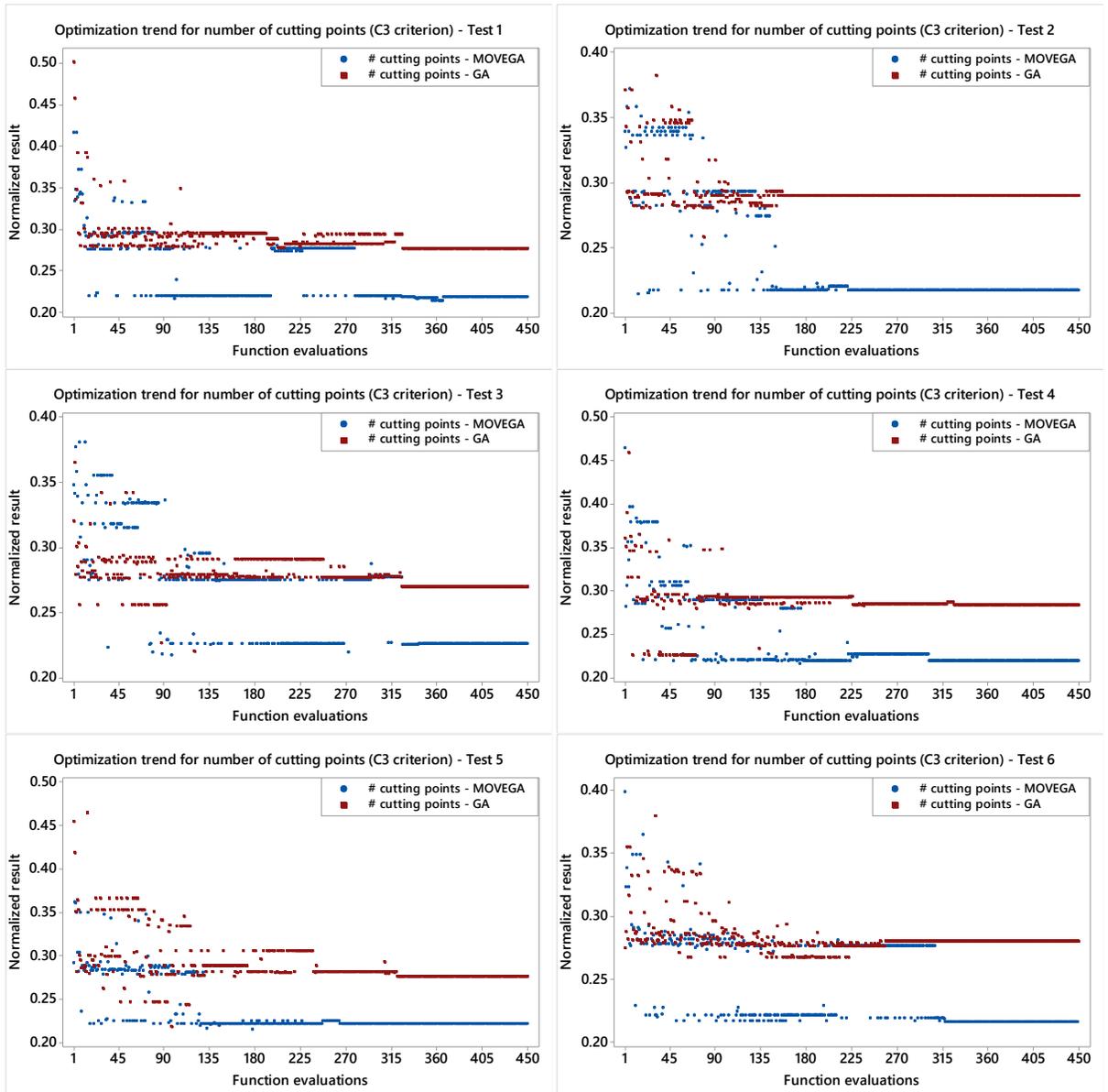
Figure 5.20: Distribution of individual values for "number of cutting points" criterion (C_3) during evolutionary optimisation.

By assuming that none of the best results is truly optimal, it would be reasonable then to consider their average values in order to set the parameters presented by the phenotypic representations of candidate solutions. In order to verify this assumption while bearing in mind the actual industrial conditions as well, the average values of tool path parameters recommended by both algorithms were applied to formulate the respective machining tool paths and conduct material removal simulations. The average values for tool path parameters are tabulated in Table 5.9. According to the average values MOVEGA recommends 44.281% (7.085mm) as a percentage of the cutting tool's diameter for stepover, 12.943° for lead angle, 2.155° for tilt angle and 5.842 mm for maximum

discretisation step whereas GA recommends 39.566% (6.330 mm) as a percentage of the cutting tool's diameter for stepover, 13.726° for lead angle, 2.624° for tilt angle and 4.920 mm for maximum discretisation step. The percentage differences for the averages as the recommended tool path parameter values are 10.65%, 8.98%, 17.87% and 15.78% for stepover, lead angle, tilt angle and maximum discretisation step respectively.

Table 5.9: Average values of the recommended optimal 5-axis tool path parameter values (MOVEGA and GA).

Algorithm	Tool	Stepover	Lead angle	Tilt angle	MaxDstep
MOVEGA	2	44.281	12.493	2.155	5.842
GA	2	39.566	13.726	2.624	4.920

To examine whether there is a statistically significant difference among the solutions obtained from both algorithms three statistical significance tests were conducted, two *2-sample variance* tests and one *paired t*-test with regard to the 450 function evaluations. All non-dominated solutions were treated as independent populations with an adequate number of samples to perform the tests. The first *2-sample variance* test deals with the examination of significantly different variance ratios / standard deviation ratios among the populations of solutions for MOVEGA algorithm. The second *2-sample variance* test deals with the examination of significantly different variance ratios / standard deviation ratios among the populations of solutions for GA algorithm. The third and last test compares the populations of solutions of the two "different" algorithms. The results are reported in Tables 5.10, 5.11 and 5.12 for the samples of MOVEGA, GA and MOVEGA-GA respectively.

Table 5.10: *2-sample variance ratio* test for detecting significant differences among results of MOVEGA.

MOVEGA pairs	N:450	1st-2nd	1st-3rd	1^{st}CP-4^{th}CP	1^{st}CP-5^{th}CP	1^{st}CP-6^{th}CP
StDev		0.106-0.103	0.106-0.113	0.106-0.109	0.106-0.093	0.106-0.097
Variance (V)		0.011-0.011	0.011-0.013	0.011-0.012	0.011-0.009	0.011-0.009
95% CI for Variances		[0.007,0.018] [0.007,0.016]	[0.007,0.018] [0.009,0.019]	[0.007, 0.018] [0.008, 0.018]	[0.007,0.018] [0.006,0.013]	[0.007,0.018] [0.006,0.015]
StDevs ratio		1.035	0.944	0.976	1.141	1.093
Variances ratio		1.071	0.892	0.954	1.301	1.194
CI for StDev ratio	*Bonett*	[0.748,1.422]	[0.689,1.258]	[0.711, 1.309]	[0.827,1.552]	[0.789,1.507]
	Levene	[0.776,1.318]	[0.697,1.162]	[0.656, 1.056]	[0.881,1.506]	[0.854,1.440]
CI for Variance ratio	*Bonett*	[0.560,2.023]	[0.475,1.582]	[0.506, 1.713]	[0.684,2.408]	[0.623,2.270]
	Levene	[0.603,1.736]	[0.486,1.350]	[0.431, 1.115]	[0.776,2.269]	[0.729,2.073]
p -value	*Bonett*	0.827	0.695	0.873	0.399	0.575
	Levene	0.938	0.419	0.134	0.300	0.435

Table 5.11: *2-sample variance ratio* test for detecting significant differences among results of GA.

GA pairs	N:450	1st-2nd	1st-3rd	1st-4th	1st-5th	1st-6th
StDev		0.092-0.091	0.092-0.080	0.092-0.096	0.092-0.103	0.092-0.087
Variance (V)		0.008-0.008	0.008-0.006	0.008-0.009	0.008-0.011	0.008-0.008
95% CI for Variances		[0.005,0.014]	[0.005,0.014]	[0.005, 0.014]	[0.005,0.014]	[0.005,0.014]
		[0.006,0.012]	[0.004,0.010]	[0.006, 0.014]	[0.007,0.017]	[0.005,0.011]
StDevs ratio		1.016	1.153	0.964	0.892	1.055
Variances ratio		1.032	1.329	0.930	0.796	1.114
CI for StDev ratio	*Bonett*	[0.694,1.393]	[0.787,1.586]	[0.656, 1.343]	[0.603,1.281]	[0.723,1.428]
	Levene	[0.885,1.514]	[0.942,1.534]	[0.789, 1.287]	[0.712,1.153]	[0.872,1.450]
CI for Variance ratio	*Bonett*	[0.482,1.939]	[0.619,2.516]	[0.431, 1.803]	[0.364,1.641]	[0.523,2.040]
	Levene	[0.783,2.294]	[0.887,2.352]	[0.623, 1.656]	[0.506,1.330]	[0.761,2.103]
p -value	*Bonett*	0.924	0.421	0.832	0.524	0.747
	Levene	0.301	0.139	0.966	0.410	0.375

Table 5.12: *Paired t*-test for detecting significant differences among non-dominated solutions of MOVEGA and GA.

MOVEGA – GA pairs	N:450	1st-1st	2nd-2nd	3rd-3rd	4th-4th	5th-5th	6th-6th
Means		0.344-0.374	0.337-0.357	0.351-0.353	0.344-0.357	0.329-0.374	0.337-0.343
St.Dev.		0.106-0.092	0.103-0.091	0.113-0.080	0.109-0.095	0.093-0.103	0.097-0.087
SE Mean		0.005-0.004	0.005-0.004	0.005-0.003	0.005-0.004	0.004-0.005	0.004-0.004
95% CI for mean diff.		[-0.032,-0.027]	[-0.022,-0.018]	[-0.005,0.001]	[-0.015,-0.012]	[-0.047,-0.044]	[-0.008,-0.005]
T -value		-25.01	-24.23	-1.14	-15.16	-57.25	-7.49
p -value		0.001	0.001	0.254	0.001	0.001	0.001

By taking into account the results of *2-variance* tests for MOVEGA and GA individually as well as the *paired t*-test for both algorithms, it is concluded that the two algorithms significantly differ in the overall functional behaviour when dealing with the generalized sculptured surface CNC machining optimisation problem as it has been formulated and reported in the thesis.

The average values taken as the optimal recommended for tool path parameters were applied to simulate the benchmark sculptured part used for establishing and conducting the RSM-CCD experiment presented in this chapter. The results examined refer to the number of cutting tool positions regarding the entire surface and its curvature characteristics as well as surface quality as an output virtually assessed using the utilities of CAM software corresponding to this performance metric. By simulating a feed rate V_f = 1500 mm/min, 9 subsequent cutting passes with 393 cutting tool positions were occurred for MOVEGA's recommended tool path whereas 11 subsequent cutting passes with 401 cutting tool positions were occurred for GA's recommended tool path. By considering the variability of sculptured parts in terms of their geometric properties such differences may

correspond to dramatically increased actual machining times. Figure 5.21 illustrates the simulated models where both the tool path and the material removal are depicted as CAM software outputs.
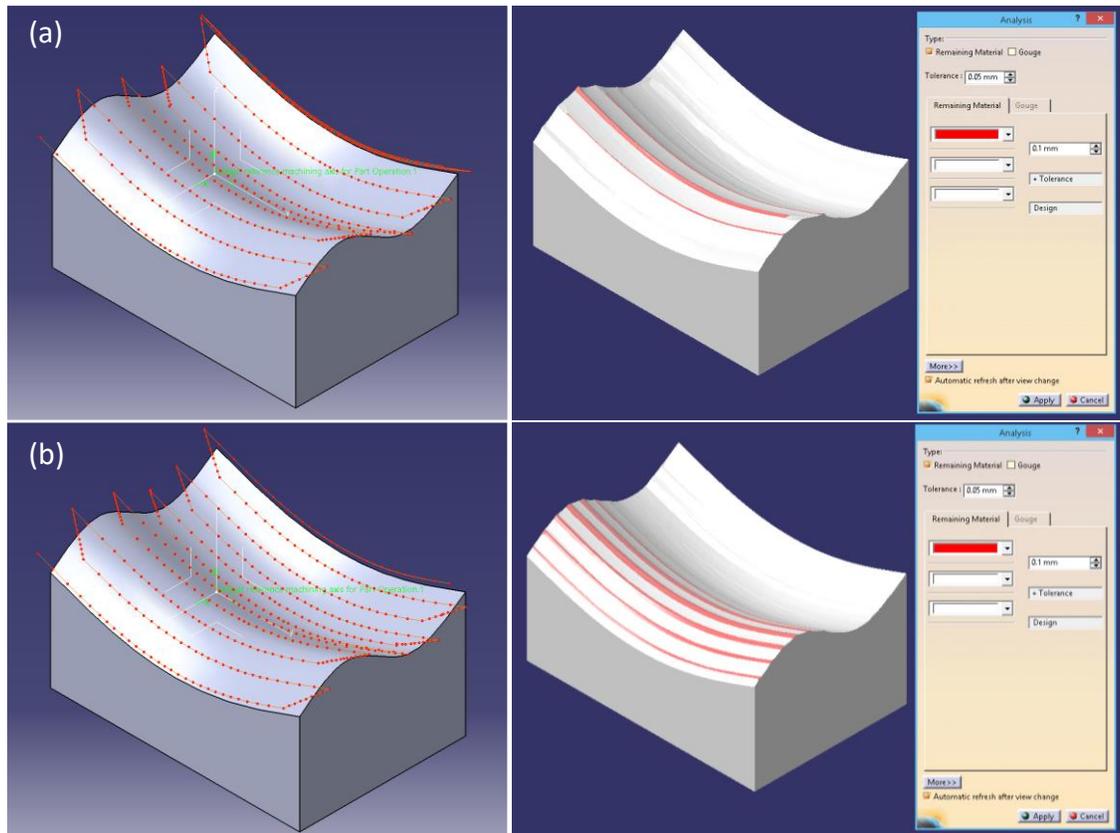
Figure 5.21: CAM software outputs using the average values of the recommended 5-axis tool path parameters: simulated tool paths and machined models for (a) MOVEGA and (b) GA.

5.4   Conclusions

Major scope of the chapter was the investigation of the efficiency of the multi-objective virus-evolutionary genetic algorithm (MOVEGA) on solving the sculptured surface machining problem and the identification of the best possible algorithm-specific parameter settings. With reference to the response surface experiments conducted and analysed it was shown that values corresponding to intermediate levels for the algorithm's parameter settings are generally preferable as the balance between exploration and exploitation is successfully maintained despite the stochastic nature of the algorithm and the high curvature detected to the experimental results. The MOVEGA integrating the proposed optimisation methodology provides robust schemata that represent "optimal" outputs for tool path parameters and they are repeatable under the perspective of statistically insignificant differences in the results despite the stochastic nature and functional complexity. The results indicate that the viral operators accompanying the algorithm have significant (positive) effect on the results

and it is clear that they positively contribute to the solutions of the generalized sculptured surface CNC machining problem. The magnitudes in terms of the main population and the population of viruses as a fraction of the former, was a result of several individual research efforts aiming at proposing a low-cost and efficient optimisation framework for the sculptured surface CNC machining problem. The methodology from the perspective of the MOVEGA may effectively operate using larger magnitudes for populations (candidate solutions and viruses), yet, at the expense of computational cost. Nevertheless it is the job of the decision-maker to determine whether it is worth using the methodology with such large magnitudes for these attributes with reference to the impact case he/she handles. The problem's nature as it has been formulated and discussed in this PhD thesis provides multi-objective Pareto fronts where all criteria are required to be minimised. This does not imply that the non-dominated solutions other than that closer to the origin are useless since cases for optimal tool paths when it comes to semi-finishing operations may also need to be examined. In such cases the final selection as well as its application so as to obtain the "optimal" output depends on the decision-maker's preferences and technical restrictions accompanying the impact case. Hence, it is possible to select a non-dominated solution other than the "optimal" one from the set that Pareto front provides according to the regions distinguished and related to the significance of each individual criterion. The intervals referring to tool path parameters for the benchmark sculptured part used for designing the response surface experiments are experimental and they have been selected to provide results with profound differences to facilitate the experimental investigation towards the optimal performance of MOVEGA. Regardless of the machining case, either experimental or actual according to industrial requirements, the intervals for tool path parameters are essential and should be meaningful, applicable and reliable in order to formulate a true search domain for the problem at hand.

The results presented and discussed in this chapter are the latest research outputs from a broader study aiming towards the direction of fine-tuning the algorithm-specific parameters of the MOVEGA integrating the proposed optimisation methodology. Similar efforts have been made using different inputs, i.e. benchmark mathematical functions, however, they do not truly reflect the problem discussed in this thesis and the results corresponding to such experiments cannot be trusted to determine the functionality of the algorithm. The recommended settings for the algorithm-specific parameters applied to optimise the tool paths for the machining operations of benchmark sculptured parts previously presented in Chapter 3. The results related to these experiments are rigorously compared to those available by other researchers for the same problem, using the same benchmark sculptured surfaces.

Chapter 6

Experimental validation

6.1　Introduction

This chapter reports the results obtained for validating the results of the intelligent methodology presented to optimise the generalized sculptured surface CNC machining problem. The results obtained refer to several tests conducted using the benchmark sculptured surfaces presented in chapter 3. The results are presented with regard to the category of experiments performed, algorithmic and process-related.

The category of algorithmic experiments deals with the selection and the implementation of modern intelligent algorithms other than the one presented (MOVEGA) for integrating the optimization methodology. To establish a common problem-solving environment for rigorously testing and comparing the algorithms, regression equations relating tool path parameters to the optimisation criteria, were generated with reference to the series of experiments reported in chapter 3. The multi-objective optimisation versions of algorithms selected for comparisons with MOVEGA are the multi-verse algorithm (MOMVO), the ant-lion optimizer (MOALO), the grey-wolf optimizer (MOGWO), the dragonfly algorithm (MODA) and another version of a multi-objective genetic algorithm (evMOGA). These algorithms were developed in *Mathworks® MATLAB®* by their inventors whilst they have been deployed using the recommended settings for their algorithm-specific parameters.

The category of process-related experiments deals with the implementation of the intelligent methodology proposed in this PhD thesis to the most complex benchmark sculptured parts in order to compare corresponding outputs to those already available in the literature by other researchers referring to the same problem and ultimately by using the same resources wherever this was possible. For this particular category the methodology presented was applied to the benchmark sculptured parts by considering them as impact cases for generating their tool paths, simulating them, proceed on actual machining and finally examine the experimental results in comparison to the most well-known and often-employed methods for tool path planning / optimisation.

6.2 Methodology validation with algorithmic tests using modern multi-objective evolutionary algorithms (MOEAs)

The aforementioned algorithms adhere to different backgrounds in terms of their operational behaviour even though they share some common utilities. For the sake of rigorous comparisons the functions and parameters of the algorithms have been set as per their inventors' recommendations for "optimal" performance whereas to provide the same problem domain for each impact case (benchmark surface).The generalized sculptured surface CNC machining optimization problem has been expressed here through empirical relations involving the independent variables (tool path parameters) and the responses (optimization criteria).

It should be mentioned that, by no means can such an approach express the generalized sculptured surface CNC machining optimization problem, yet it may establish a quite reliable solution domain to navigate to, for comparison purposes regarding the aforementioned antagonizing algorithms. Nevertheless, the algorithms could be improved by using the CAD/CAM system's compatible programming language so as to handle the problem as MOVEGA does, however such an attempt goes far beyond the research of this PhD thesis. It should be noted that the methodology presented in the thesis has been developed in a way that any intelligent algorithm compatible to the environment could integrate the proposed optimisation methodology instead of MOVEGA.

6.2.1 Fundamental features and properties of selected MOEAs

6.2.1.1 Multi-objective multi-verse optimizer – MOMVO (Mirjalili et al. 2017).

MOMVO algorithm (Mirjalili et al. 2017) adheres to the principles of some cosmological theories suggesting that multiple universes exist and simulates their interaction through white hole, black hole and worm hole. According to Physics objects may be transferred form a universe via a tunnel from a white hole towards a black hole. As regards worm holes, they are capable of moving objects form the "boundaries" of a universe to the "boundaries" of another without the presence of a white or black hole. MOMVO is an evolutionary algorithm and as such it belongs to the population-based heuristics.

Optimization procedure initializes a set of candidate solutions. Each candidate solution is considered to be a "universe" whilst variables are analogous to "objects" in the universe. MOMVO deploys its specific operators to combine solutions and distinguish elite ones. To achieve combination among solutions white and black holes are randomly generated in the "universes" causing the movement of objects. MOMVO evaluates an objective function as it occurs to all heuristics. MOMVO employs also

the inflation rate which is one of MOMVO's algorithm-specific parameters and simulates the growing speed of a "universe" computed proportional to the objective function. In other words, inflation rate is the objective value attained by evaluating the objective function for a given "universe". In MOMVO when inflation rate increases a higher probability occurs for white holes to improve solutions. On the contrary existence of black holes is inversely proportional to inflation rate causing the variables' flow from worse "universes" to better ones.

By incorporating the aforementioned features in MOMVO any solution can contribute to the generation of new solutions as opposed to crossover that mates only two parents for producing a child. In addition white and black holes maintain exploration of solution space owing to changing solutions in a sudden sense. The "elitistic" behaviour of MOMVO keeps the best solutions obtained so far whereas worm holes generate tunnels between the best solution and any other solution to pass information and this finally aims to improve exploitation in MOMVO.

6.2.1.2 Multi-objective ant-lion optimizer – MOALO (Mirjalili et al. 2015).

The multi-objective ant-lion optimization algorithm – MOALO simulates the hunting behaviour of antlions found in nature. There are five steps for hunting a prey such as the random walk (scouting) of ants, trap building, ant trapping, prey catching and trap rebuilding. Technically the MOALO simulates the interaction of antlions and ants as a population-based heuristic whereas optimal solutions are approximated by initializing a group of random solutions. The main goal of ants is to explore the search space. They are supposed to move around the search space by taking a random walk. The antlions maintain the best position obtained by the ants and guide the search of ants towards promising regions of the search space.

The general steps of MOALO for exchanging information among antlions and ants and gradually reaching global optimum according to the natural procedure stated above are the following:

a. Random initialization of a number of ants as main search "agents".
b. Ant fitness evaluation regarding the objective function.
c. Random walk of ants around the antlions in the search space.
d. The population of antlions is never evaluated. In fact, antlions assumed to be on the location of ants in the first iteration and relocate to the new positions of ants in the rest of iterations if the ants become better.
e. There is one antlion assigned to each ant and updates its position if the ant becomes fitter.
f. There is also an elite antlion which impacts the movement of ants regardless of their distance.

g. If any antlion becomes better than the elite, it will be replaced with the elite.

h. Steps *b* to *g* are repeated until stopping criteria are met.

The mathematical model and programming modules proposed for each of these steps are reported in (Mirjalili, Jangir and Saremi, 2017).

6.2.1.3 Multi-objective grey-wolf optimizer – MOGWO (Mirjalili et al. 2014).

Multi-objective grey-wolf optimizer – MOGWO is another population-based algorithm that simulates the behaviour in terms of leadership hierarchy of grey-wolves. In engineering computation four types of grey wolves, alpha, beta, delta and omega are distinguished. Moreover, three types of hunting techniques are followed as major steps by the grey-wolves, prey searching, prey encircling (trapping) and attacking. These steps are also the computational steps for conducting optimization to a problem with this algorithm. Grey-wolves use to live in packs consisting of 5 to 12 grey-wolves in average. They have a very strict social hierarchy starting from alphas, which are a male and a female grey-wolf. Alphas are responsible for decision making when it comes to hunting, sleeping place, wake time, and so on. These decisions should be followed by the rest of grey-wolves in the pack. A more "democratic" behaviour about the living behaviour of grey-wolves has also been observed where alphas may follow the rest of the wolves in the pack. Alpha wolves are those who dominate in their corresponding packs whilst they are the only ones allowed to mate. Surprisingly, alpha wolves are not necessarily the strongest members in a patch but the best in managing and strategic decision making. This implies that discipline and organization is much more essential than strength at least when it comes to grey-wolves.

Second in hierarchy come the "beta" grey-wolves. The "betas" act as advisors to alphas and help them in decision making as well as other pack activities. Betas may be males or females whereas they are probably the best candidates to be alphas, should an alpha wolf passes away or grows too old. Even though a beta wolf should respect an alpha one, a beta may command the rest low-level wolves, as a discipliner. Thus, the beta emphasizes alpha's commands to the whole pack and feedbacks to alpha. Omega grey-wolves are the lowest in hierarchy.

The lowest ranking grey-wolves are "omegas". Omegas undertake the role of scapegoats. Omega wolves always submit to all the rest dominant wolves and they are the last ones allowed to eat. Though it seems that omegas are not just as important individuals in the pack as alphas or betas, it has been observed that the whole pack can face internal fighting and problems in case of losing an omega owing to the venting of violence and frustration of all wolves by the omegas. This contributes to the entire pack's satisfaction and maintains the dominance structure. A grey-wolf other than

alphas, betas and omegas is considered to be a "delta". Delta grey-wolves report to alphas as well as betas, yet, they dominate omegas. Scouting wolves, sentinels, hunters and caretakers are fall to this category. They watch the bounds of their territory and warn the pack for imminent dangers. All engineering computation steps and programming modules corresponding to the steps for executing MOGWO algorithm are reported in (Mirjalili et al. 2014).

6.2.1.4 Multi-objective dragonfly algorithm – MODA (Mirjalili et al. 2016).

Dragonflies are considered as small predators hunting almost all other smaller insects found in nature. Nymphs also predate on other marine insects or small fishes. What is interesting about dragonflies is their unique swarming behaviour. Dragonflies swarm only for two major goals, hunting and migrating. The former is known as the static (feeding) swarm whereas the latter is known as the dynamic (migratory) swarm. When it comes to static swarm dragonflies formulate small groups flying back and forth over a small region to hunt other preys such as butterflies and mosquitoes (Wikelski et al. 2006). Local movements and abrupt changes in the flying path are the major characteristics of a static swarm. When it comes to dynamic swarms a vast number of dragonflies migrate towards long-distanced directions (Russell et al. 1998).

These two swarming behaviours implemented to MODA algorithm simulate the two mandatory attributes of optimization algorithms, exploration and exploitation. Dragonflies formulate sub-swarms to fly over several territories in a static swarm, which is the objective of the exploration phase. In addition, if a static swarm is formulated by a larger number of dragonflies flying along one specific direction facilitates the exploitation phase. According to Reynolds, the behaviour of swarms follows three primitive principles (Reynolds 1987):

   a. Separation - referring to the static collision avoidance of individuals from other individuals in the searching neighbourhood,
   b. Alignment – that indicates velocity matching of individuals to that of other individuals in the searching neighbourhood, and
   c. Cohesion - referring to the tendency of individuals towards the centre of mass of the searching neighbourhood.

As survival is the major objective of any type of swarm or tribe, all individuals (candidate solutions) ought to be attracted towards food sources and avoid outward enemies. Exploration and exploitation phases as well as major steps that MODA algorithm deploys to solve an optimization problem are mathematically modelled and reported in (Mirjalili et al. 2016).

175

6.2.1.5 Evolutionary multi-objective genetic algorithm – evMOGA (Martinez et al. 2009).

Martinez et al. (2009) suggested a multi-objective genetic/evolutionary algorithm in order to obtain robust non-dominated sets of solutions well-distributed in Pareto fronts. The evMOGA algorithm follows the functional principles of the non-dominated sorted genetic algorithm where the elitistic behaviour for preserving some few outstanding candidate solutions for next generations, can be controlled. Non-dominated solutions of early function evaluations are stored internally and to a separate log file. The rest of the non-dominated solutions form the evolving set is stored in an external archive. This ensures minimal disruption of Pareto front solution patterns already obtained by earlier function evaluations.

6.2.2 Algorithmic experimental results

Ten individual try outs were determined with 10 candidate solutions to be evolved for 5 generations which is equal to 450 function evaluations for all MOEAs. Pareto 3D criterion was formulated using the regression models developed (a model per benchmark surface) and after normalizing their corresponding outputs in the {0-1} interval. To create the regression models the 2nd order polynomial relation presented in Equation 5.2 (Chapter 5) was adopted and is also given here for easy reference, as Equation 6.1.

$$Y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_{ii}^2 + \sum_{i=1}^{k} \sum_{i \neq j=1}^{k} \beta_{ij} x_i x_j \qquad \text{Eq. 6.1}$$

where *Y* represents the responses, *k* is the overall number of independent variables, β_0 is a constant term and *i, ii, ij* accompanying β are the coefficients for linear, quadratic and interaction terms respectively. Finally, x_i, x_j represent the coded levels for the independent parameters.

From the ten runs of each MOEA and benchmark surface, the best (minimum), the worst (maximum) the average and the standard deviation of solutions were considered. The study considers the time-consuming effort and computational burden of the approach when operating using CAM software and this justifies the aforementioned settings. During the tests all MOEAs were operated according to the last best population to investigate to what extend the suggested "optimal" solution could be improved. Tables 6.1, 6.2, 6.3 and 6.4 summarize the results of mean values for best (minimum), worst (maximum), average and standard deviation for all MOEAs and test surfaces. Figure 6.1 gives a graphical comparison among the best (minimum) solutions of MOEAs regarding the number of

executions per benchmark sculptured surface. In all cases MOVEGA attained the minimal result, thus, improving the multi-objective optimization criterion (Pareto 3D).

Table 6.1: Optimization results for MOEAs with regard to the benchmark sculptured surface 1 (SS-1).

Indices / Metrics	MOVEGA	nvMOGA	MOMVO	MOALO	MOGWO	MODA	evMOGA
$\overline{\min_{10} ffp}$	0.513	0.520	0.546	0.558	0.547	0.546	0.630
$\overline{\max_{10} ffp}$	1.470	1.343	0.854	0.999	1.003	0.929	0.874
$\overline{avg_{10} ffp}$	0.625	0.610	0.637	0.688	0.657	0.632	0.699
$\overline{stdev_{10} ffp}$	0.070	0.089	0.068	0.098	0.102	0.078	0.059

Table 6.2: Optimization results for MOEAs with regard to the benchmark sculptured surface 2 (SS-2).

Indices / Metrics	MOVEGA	nvMOGA	MOMVO	MOALO	MOGWO	MODA	evMOGA
$\overline{\min_{10} ffp}$	1.085	1.093	1.193	1.209	1.242	1.273	1.390
$\overline{\max_{10} ffp}$	2.052	2.384	2.364	2.461	2.558	2.327	2.217
$\overline{avg_{10} ffp}$	1.154	1.232	1.611	1.668	1.784	1.696	1.739
$\overline{stdev_{10} ffp}$	0.084	0.168	0.275	0.250	0.300	0.232	0.210

Table 6.3: Optimization results for MOEAs with regard to the benchmark sculptured surface 3 (SS-3).

Indices / Metrics	MOVEGA	nvMOGA	MOMVO	MOALO	MOGWO	MODA	evMOGA
$\overline{\min_{10} ffp}$	0.411	0.418	0.466	0.454	0.461	0.458	0.620
$\overline{\max_{10} ffp}$	1.309	1.303	1.076	1.295	1.266	1.256	1.174
$\overline{avg_{10} ffp}$	0.426	0.498	0.740	0.848	0.813	0.791	0.790
$\overline{stdev_{10} ffp}$	0.080	0.142	0.172	0.212	0.217	0.243	0.132

Table 6.4: Optimization results for MOEAs with regard to the benchmark sculptured surface 4 (SS-4).

Indices / Metrics	MOVEGA	nvMOGA	MOMVO	MOALO	MOGWO	MODA	evMOGA
$\overline{\min_{10} ffp}$	0.848	0.879	0.876	0.917	0.857	0.907	1.085
$\overline{\max_{10} ffp}$	1.599	1.757	1.406	2.080	1.896	1.789	1.594
$\overline{avg_{10} ffp}$	0.861	1.079	1.065	1.572	1.199	1.173	1.267
$\overline{stdev_{10} ffp}$	0.060	0.124	0.166	0.314	0.243	0.231	0.143
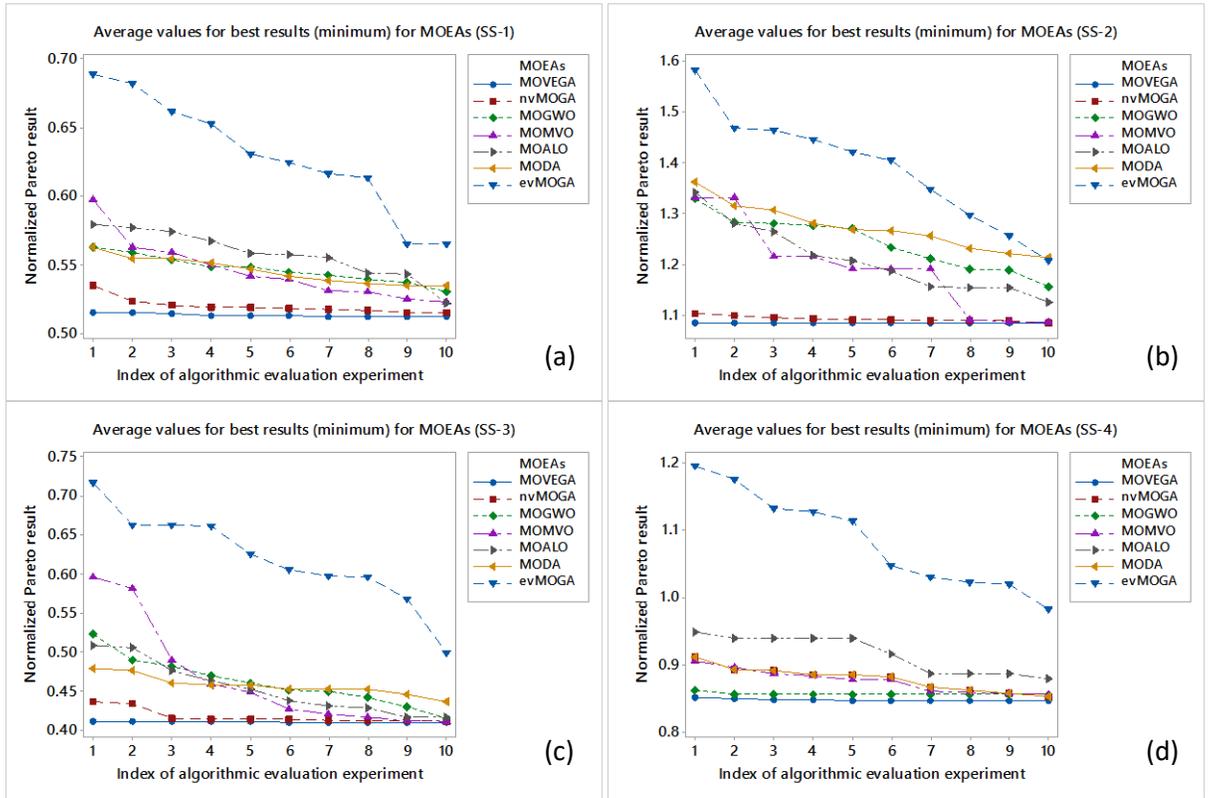
Figure 6.1: Optimal Pareto results for the independent algorithmic evaluations: (a) SS-1, (b) SS-2, (c) SS-3, (d) SS-4.

The algorithmic study also demonstrates the MOVEGA's ability to produce approximately the same result, since for each independent test, final points were obtained with very small differences. Figures 6.2 to 6.5 show the convergence diagrams for the tests exhibited the optimum result for all the algorithms examined, taking as objective functions the regression models based on the experiments carried out for the benchmark sculptured surfaces presented in Chapter 3. Figure 6.2 shows the algorithms' convergence diagrams for the best test out of the 10 totals for each of those concerning the benchmark sculptured surface SS-1. It is clear that the competitive algorithms that actually follow a swarm-based intelligent philosophy attain faster convergence towards the best result in relation with nvMOGA and MOVEGA. Nevertheless, the last two, nvMOGA and MOVEGA, maintain the convergence beyond the evaluation numbers where results for the algorithms-competitors are obtained. This significant development is observed in 315th function evaluation and further where both nvMOGA and MOVEGA continue the downward path towards the minimum, while even more intense convergence is presented by MOVEGA.
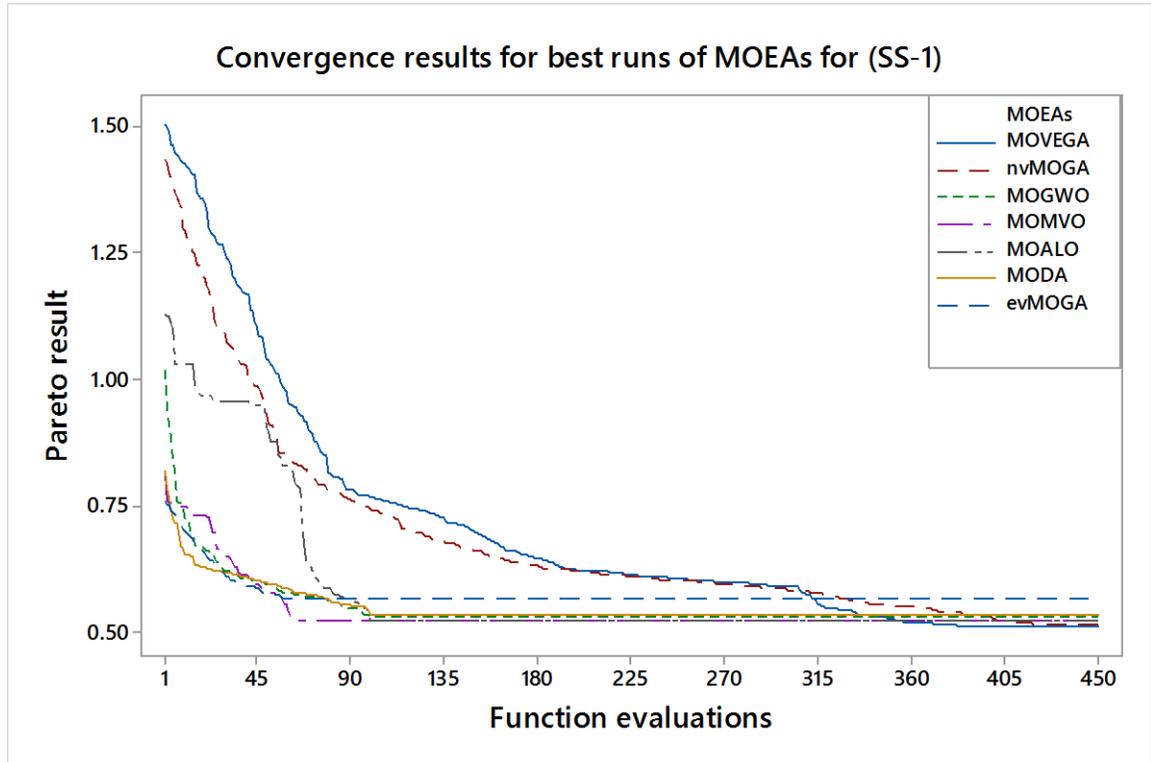
Figure 6.2: Comparison of best runs for MOEAs for SS-1.

Similar behaviour is also observed for the algorithms with regard to their best algorithmic tests out of 10 total, for the benchmark sculptured surface SS-2 (Fig. 6.3), except that nvMOGA and MOVEGA start to converge from much lower values than those of MOGWO, MOALO and MODA algorithms, a behaviour which was not observed in the corresponding convergence diagrams referring to surface SS-1. This is most likely due to effect of each objective function's unique characteristics / factors on initialization operation for generating candidate solutions. In the case of benchmark sculptured surface SS-2 the difference in convergence speed between nvMOGA and MOVEGA is more profound with the latter to achieve the lowest objective value over all other algorithms. The trend of the abrupt convergence exhibited by the algorithms-competitors is obvious again, yet, without achieving a better result than of MOVEGA. Some of the convergence diagrams of algorithms-competitors seem to suggest local trapping to near-optimal values rather than a quick convergence to the best possible result. This phenomenon is strongly observable in the convergence diagrams of MOMVO, MOGWO and evMOGA algorithms. Nevertheless, the performance of MOMVO, MOGWO as well as MODA algorithms seems to be very good considering that their final result is quite close to the final point that MOVEGA suggests, while it is achieved during the very first function evaluations.

179
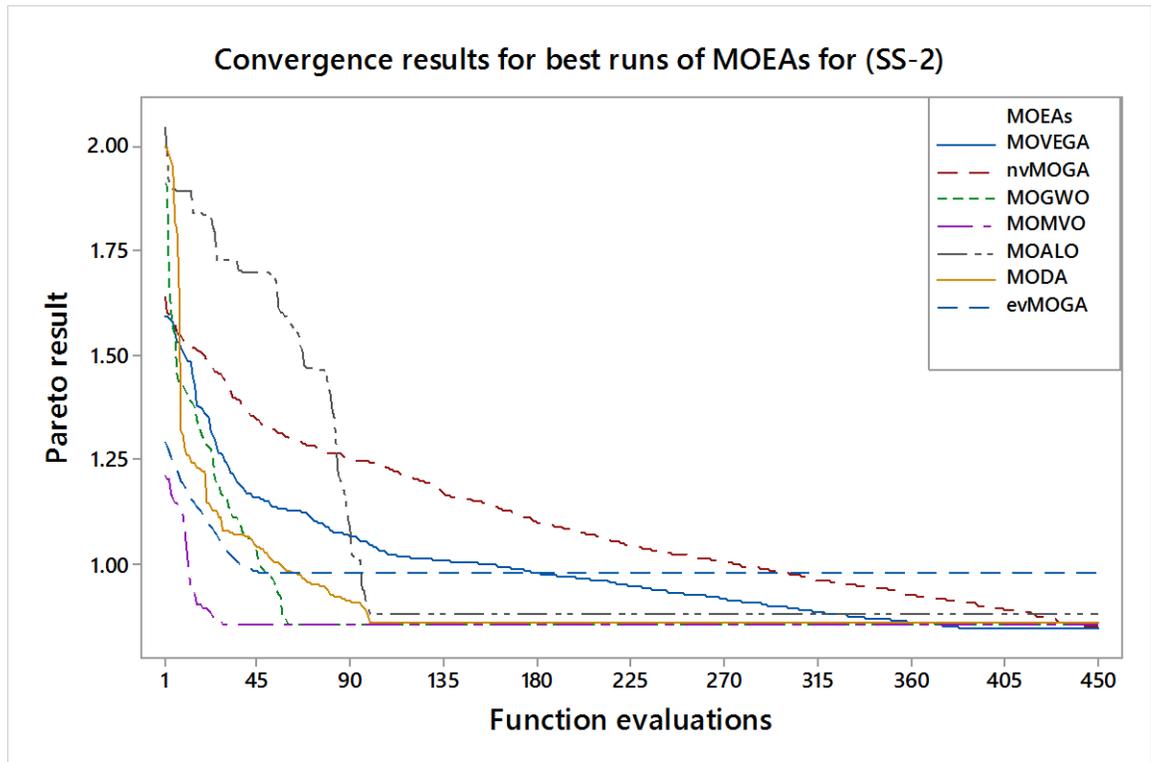
Figure 6.3: Comparison of best runs for MOEAs for SS-2.

Figure 6.4 shows the algorithms' convergence diagrams for the tests exhibited the optimum result relating to benchmark sculptured surface SS-3. In this case, all algorithms except MOGWO begin their convergence towards the optimal objective value, almost by the same starting point. Yet again, the greatest convergence speed in favor of MOVEGA against nvMOGA is clear. The steep convergence of the rest algorithms-competitors as a main feature is also apparent in this case, whereas there is a significant improvement in MOALO's convergence trend, which is not observed in previous cases. Early convergence appears to be presented by the MODA and evMOGA algorithms, while the final results of both MOGWO and MOALO are deemed moderate. Final convergence path towards the optimal result for MOVEGA appears to start from the 360th function evaluation while nvMOGA exhibits a final steep convergence path in the last function evaluations. There is also a fairly similar behaviour in the convergence trend between MOVEGA and nvMOGA algorithms, during 90th and 315th function evaluations, yet, the difference in MOVEGA's efficiency in terms of convergence compared to nvMOGA is evident and is clearly owing to viral operators.
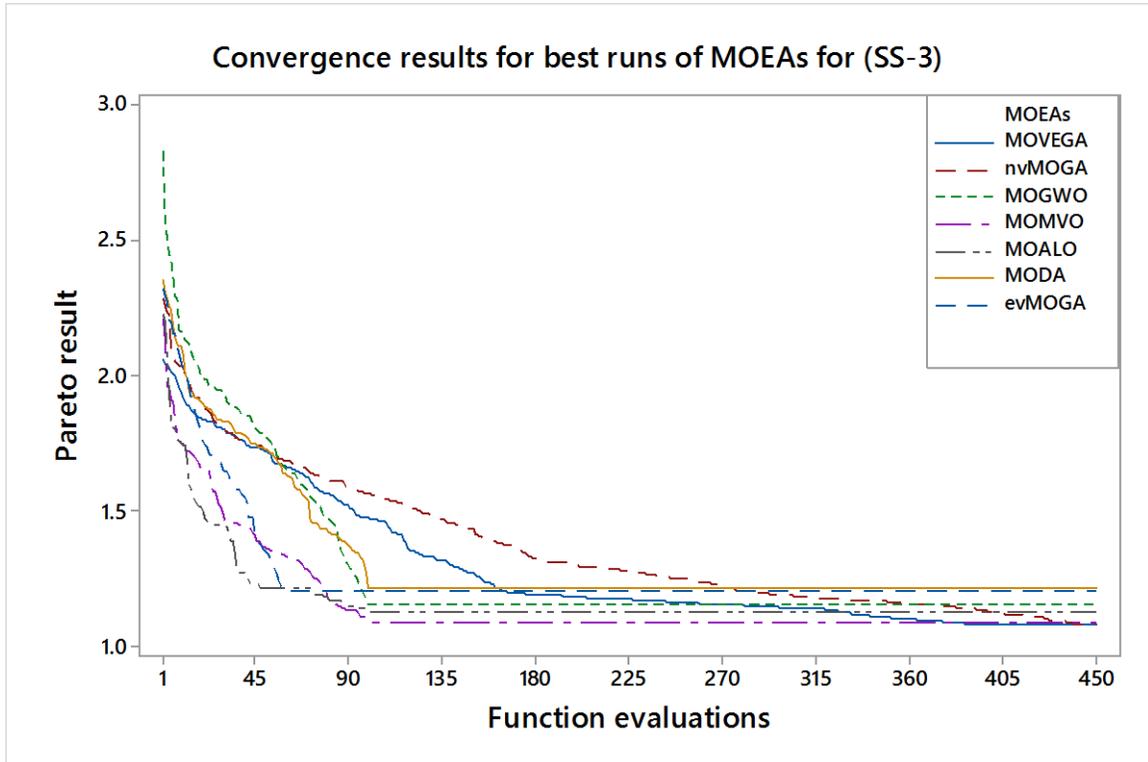
Figure 6.4: Comparison of best runs for MOEAs for SS-3.

Figure 6.5 shows the algorithms' convergence diagrams for the tests exhibited the optimum result relating to benchmark sculptured surface SS-4. In this case, all algorithms seem to start their convergence by the same objective function values. MOVEGA and nvMOGA show almost identical behaviour in terms of their convergence, while the viral operator's contribution is evident from the 280th evaluation up to the convergence to final result. As far as the rest of the algorithms-competitors are concerned, MODA and evMOGA appear to trap to local minima whilst evMOGA's performance leads to a final result far from those obtained by the rest algorithms.

By examining the overall performance of algorithms with emphasis to MOVEGA, it appears that the latter maintains a fairly stable convergence path regardless of the differences in the associated objective functions it evaluates for solving the sculptured surface CNC machining problem, at least when it is represented using regression models that correlate the criteria with regard to the independent variables. The same seems to apply to the rest of algorithms-competitors yet to a lower extent, an issue that does not allow for drawing safe conclusions as to their overall functionality and robustness for addressing the sculptured surface CNC machining problem.

Convergence results for best runs of MOEAs for (SS-4)
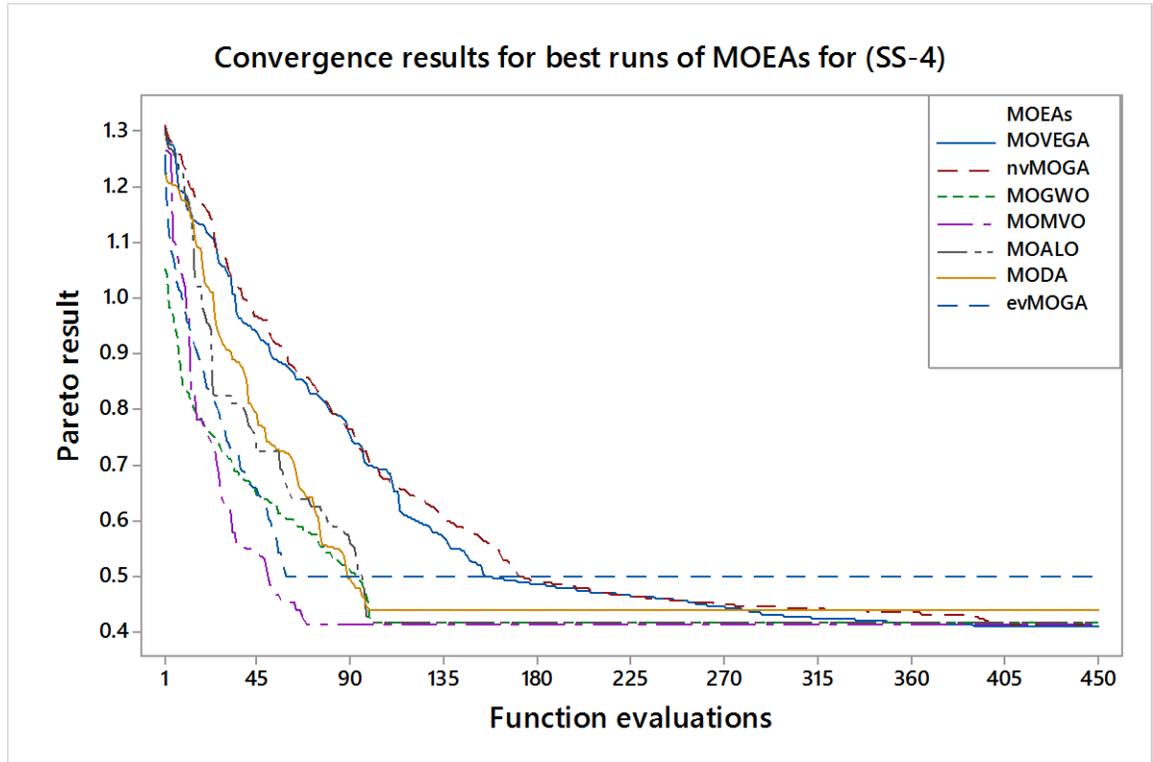
Figure 6.5: Comparison of best runs for MOEAs for SS-4.

Summarizing the observations on the results obtained by the best algorithmic tests and by considering their final results, it broadly seems that all algorithms-competitors present small deviations which could give the false impression that the algorithms show the same performance, hence, they can practically lead to the same outcome. To either prove, or reject this assumption, a series of statistical hypothesis tests were carried out to investigate for any statistically significant differences in the results obtained by the algorithms.

Successive *2-sample t*-tests were performed per pairs of algorithms considering all their non-dominated solutions as independent populations. The null hypothesis H_0 in all tests assumed that there is no statistically significant difference in MOVEGA's results with any other multitude of results of the rest algorithms-competitors. The results of the *2-sample t*-tests for all benchmark sculptured surfaces and best algorithmic tests are summarized in the Tables 6.5 to 6.8. In these results, as in previous statistical analyses of experiments, emphasis is given to p -value. For most of the pairs of algorithms it was found that there is a statistically significant difference between them and therefore it cannot be questioned that the results obtained by the algorithms examined, ultimately differ. It should be noted here that the sets of the non-dominated solutions resulting from each algorithm involve only the original ones and not their replicates, since it is common for intelligent algorithms to

exhibit the same solutions in a given time-span for performing evaluations until they finally converge to their best scores. This was done in order to reduce the inherent bias that would appear in statistical metrics such as the mean and standard deviation of each population, leading thus to an unreliable conclusion about whether or not the results of the algorithms-competitors are different. All pairs of algorithms included the MOVEGA as the major algorithm under question and another algorithm of those examined, whereas pairs between the algorithms-competitors themselves were not examined since this is of less importance for the current research. Table 6.5 summarizes the results for the *2-sample t*-test with reference to the best experiment and its related results per algorithm and in accordance with the regression model representing the sculptured surface CNC machining problem using the benchmark sculptured surface SS-1. It is shown that MOVEGA and nvMOGA have presented a greater number of solutions than the other algorithms-competitors, suggesting that MOVEGA and nvMOGA are far more superior in terms of the accuracy in searching for the global best within the search space.

Table 6.5: *2-sample t*-test results for best runs of MOEAs for testing significant differences with regard to the benchmark sculptured surface 1 (SS-1).

Pairs	N	Means	StDev	95% C.I. for difference	T-value	p - value
MOVEGA-nvMOGA	399-422	0.729-0.691	0.244-0.196	(0.0078, 0.0682)	2.47	0.014
MOVEGA-MOGWO	399-100	0.729-0.621	0.244-0.090	(0.0593, 0.1567)	4.36	0.001
MOVEGA-MOMVO	399-062	0.729-0.659	0.244-0.075	(0.0095, 0.1321)	2.27	0.024
MOVEGA-MOALO	399-100	0.792-0.827	0.244-0.188	(0.0463, 0.1490)	3.74	0.001
MOVEGA-MODA	399-100	0.729-0.606	0.244-0.052	(0.0757, 0.1720)	5.05	0.001
MOVEGA-evMOGA	399-062	0.729-0.631	0.244-0.058	(0.0368, 0.1590)	3.15	0.002

Table 6.6 summarizes the results for the *2-sample t*-test performed for the best outputs from the total of 10 algorithmic experiments per algorithm with reference to the regression model representing the sculptured surface CNC machining problem using the benchmark sculptured surface SS-2. In this case, MOVEGA and nvMOGA have also provided a larger solution sets than those obtained by the rest of algorithms-competitors. In this particular case a minor statistical difference between MOVEGA's and MOMVO's results was found. However, this occurrence may not represent the true statistical conclusion since the number of solutions is much smaller than that of MOVEGA and consequently the overall outcome may be influenced by the comparison between two sets of solutions that are significantly different in their magnitudes.

Table 6.6: *2-sample t*-test results for best runs of MOEAs for testing significant differences with regard to the benchmark sculptured surface 2 (SS-2).

Pairs	N	Means	StDev	95% C.I. for difference	T-value	p - value
MOVEGA-nvMOGA	474-441	1.031-1.098	0.173-0.182	(0.0448, 0.0908)	5.780	0.001
MOVEGA-MOGWO	474-061	1.031-1.201	0.173-0.241	(0.1219, 0.2193)	6.880	0.001
MOVEGA-MOMVO	474-030	1.031-0.995	0.173-0.131	(0.0990, 0.0276)	1.110	0.268

MOVEGA-MOALO	474-100	1.031-1.580	0.173-0.287	(0.5070, 0.5925)	25.26	0.001
MOVEGA-MODA	474-100	1.031-1.101	0.173-0.262	(0.0286, 0.1114)	3.320	0.001
MOVEGA-evMOGA	474-046	1.031-1.102	0.173-0.096	(0.0203, 0.1223)	2.750	0.006

Table 6.7 summarizes the results for the *2-sample t*-test performed for the best outputs from the total of 10 algorithmic experiments per algorithm with reference to the regression model representing the sculptured surface CNC machining problem using the benchmark sculptured surface SS-3. MOVEGA and nvMOGA have both presented an adequate number of solutions as opposed to the rest of algorithms-competitors, whilst evMOGA exhibits only 59 solutions. However the population size exceeds the limit for reliable statistical analysis, and therefore one can conclude that the results of MOVEGA and MOALO do not statistically differ in this particular case. For the rest of the pairs and the one between MOVEGA and nvMOGA in particular, the results have statistically significant differences.

Table 6.7: *2-sample t*-test results for best runs of MOEAs for testing significant differences with regard to the benchmark sculptured surface 3 (SS-3).

Pairs	N	Means	StDev	95% C.I. for difference	T -value	p - value
MOVEGA-nvMOGA	398-442	1.320-1.371	0.255-0.253	(0.0171, 0.0859)	2.940	0.003
MOVEGA-MOGWO	398-100	1.320-1.761	0.255-0.336	(0.3813, 0.5013)	14.45	0.001
MOVEGA-MOMVO	398-100	1.320-1.424	0.255-0.246	(0.0488, 0.1600)	3.690	0.001
MOVEGA-MOALO	398-100	1.320-1.349	0.255-0.243	(0.0848, 0.0262)	1.400	0.300
MOVEGA-MODA	398-100	1.320-1.702	0.255-0.263	(0.3262, 0.4389)	13.33	0.001
MOVEGA-evMOGA	398-059	1.320-1.691	0.255-0.305	(0.2992, 0.4428)	10.16	0.001

Finally, table 6.8 summarizes the results for the *2-sample t*-test performed for the best outputs from the total of 10 algorithmic experiments per algorithm with reference to the regression model representing the sculptured surface CNC machining problem using the benchmark sculptured surface SS-4. In this case, MOVEGA and nvMOGA do not exhibit statistically significant differences between them as it is indicated by the corresponding p -value. By observing the magnitudes of the solution sets for both MOVEGA and nvMOGA algorithms, one would advocate that the statistical conclusion of whether there are significant differences or not, can ultimately be affected by the equality of "sample" magnitudes considered. In other words, if the sample sizes examined are equal (or almost equal), then there is high possibility that no statistically significant differences are to be found. However, this is not valid because, such statistical tests emphasize on the average, the standard deviation, or median, rather than the sample size, and on the other hand, *2-sample t*-test is robust enough to characterize such solution sets despite the difference in sample magnitudes provided that the limited size to allow for a reliable statistical analysis is satisfied. Besides, the *paired t*-test may be applied when it comes to statistical significance testing among pairs of results of the same magnitude.

Table 6.8: *2-sample t*-test results for best runs of MOEAs for testing significant differences with regard to the benchmark sculptured surface 4 (SS-4).

Pairs	N	Means	StDev	95% C.I. for difference	T-value	p-value
MOVEGA-nvMOGA	403-404	0.590-0.606	0.229-0.234	(0.0485, 0.0155)	1.01	0.313
MOVEGA-MOGWO	403-100	0.590-0.662	0.229-0.138	(0.0247, 0.1186)	3.00	0.003
MOVEGA-MOMVO	403-069	0.590-0.679	0.229-0.242	(0.0296, 0.1479)	2.95	0.003
MOVEGA-MOALO	403-100	0.590-0.779	0.229-0.229	(0.1386, 0.2391)	7.38	0.001
MOVEGA-MODA	403-100	0.590-0.794	0.229-0.236	(0.1538-0.2550)	7.94	0.001
MOVEGA-evMOGA	403-060	0.590-0.787	0.229-0.180	(0.1361, 0.2576)	6.37	0.001

The optimal parameter values proposed by each algorithm were transferred to the advanced machining workbench of *Dassault Systemes® CATIA® V5R18* to compute the tool paths and conduct machining simulations aiming at examining manufacturing results directly from CAM outputs. All machining simulations were conducted by applying a tool path offset equal to 0.2mm above the ideal sculptured surface so as to ensure the noticeable differences in scallop height and excess material. In machining simulations productivity is characterized by machining and total times which are straightforwardly provided by CAM software whist surface finish is characterized by the excess material. For the particular evaluation of surface finish using the CAM outputs from the simulations, the volume of the excess material was measured rather than scallop height and / or chord error because the optimal results for the tool paths have been emerged without having taken into account the importance of CAM properties and without having made the necessary calculations reported in Chapter 3 for the clear and concise formulation of the generalized sculptured surface CNC machining problem as well as its optimisation, thus, the indices of scallop height and/or chord error cannot provide clear evidence of the machining accuracy via regression modelling. In addition, excess material and its graphical distribution over the ideal sculptured surface may, at least, provide information about the material geometry. Thus, a large scallop height with small material volume may suggest tall and sharp scallops. On the contrary low scallop heights accompanied to large volumes may indicate small but wide scallops.

Examination of CAM outputs involved the geometric comparisons between the ideal model (benchmark sculptured surface) and the machined model. All geometric comparisons were conducted using 0.1mm precision which has been considered sufficient to detect and represent most of the features of the machined sculptured surface (Warkentin et al. 2000). The results obtained by the machining simulations are graphically illustrated in the following sections whereas the aforementioned results relating to machining and total times as well as excess material volumes are also reported in Tables.

The graphical representations of CAM outputs as manufacturing results present the machined models (as they occur from the application of the multi-axis sweeping strategy with its corresponding parameter values), the tool paths (where the number of passes is more profound, something that can characterize the cutting strategy from a production perspective) and finally the excess material volume in the form of a three-dimensional (3D) geometric "map". In the 3D geometric "maps" of excess material, the regions where the material has been deemed to be "completely" removed (and therefore no excess material is considered) are obvious, as well as the areas with obvious material left on the target surface, which may be represented either in the form of scallops, or in random formation, according to the material removal pattern depending on the tool path parameter values as well as the topological characteristics of such regions. The excess material's magnitude depends on the tool path parameter values and the predetermined cutting tolerance regarding the machining case.

Whether the differences between these results are graphically evident or not, depends mainly on the difference in the results of the values for the parameters proposed by the algorithms employed and on the geometrical characteristics - complexity of the benchmark sculptured surface. Fortunately, in this study, obvious differences were found mainly in the machined models, as well as in the three-dimensional geometric "charts" of excess material, despite the small differences in parameter values. These differences are also observed in the results tables accompanying the graphical evidences (Tables 6.9 to 6.16) where the small but noticeable difference among the magnitudes of the measured values is given. The graphical results of the study are shown in Figures 6.6, 6.7, 6.8 and 6.9 for the SS-1, SS-2, SS-3 and SS-4 benchmark sculptured surfaces respectively. Note that the classification of results for all tables (Tables 6.9 to 6.16) as well as figures (Figs 6.6 to 6.9) follows an ascending ranking according to the "best" (lowest) Pareto3D outputs obtained by the MOEAs.
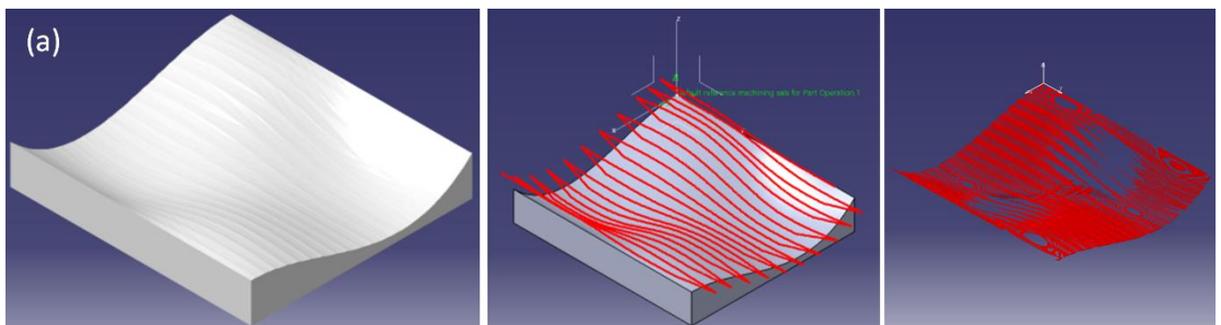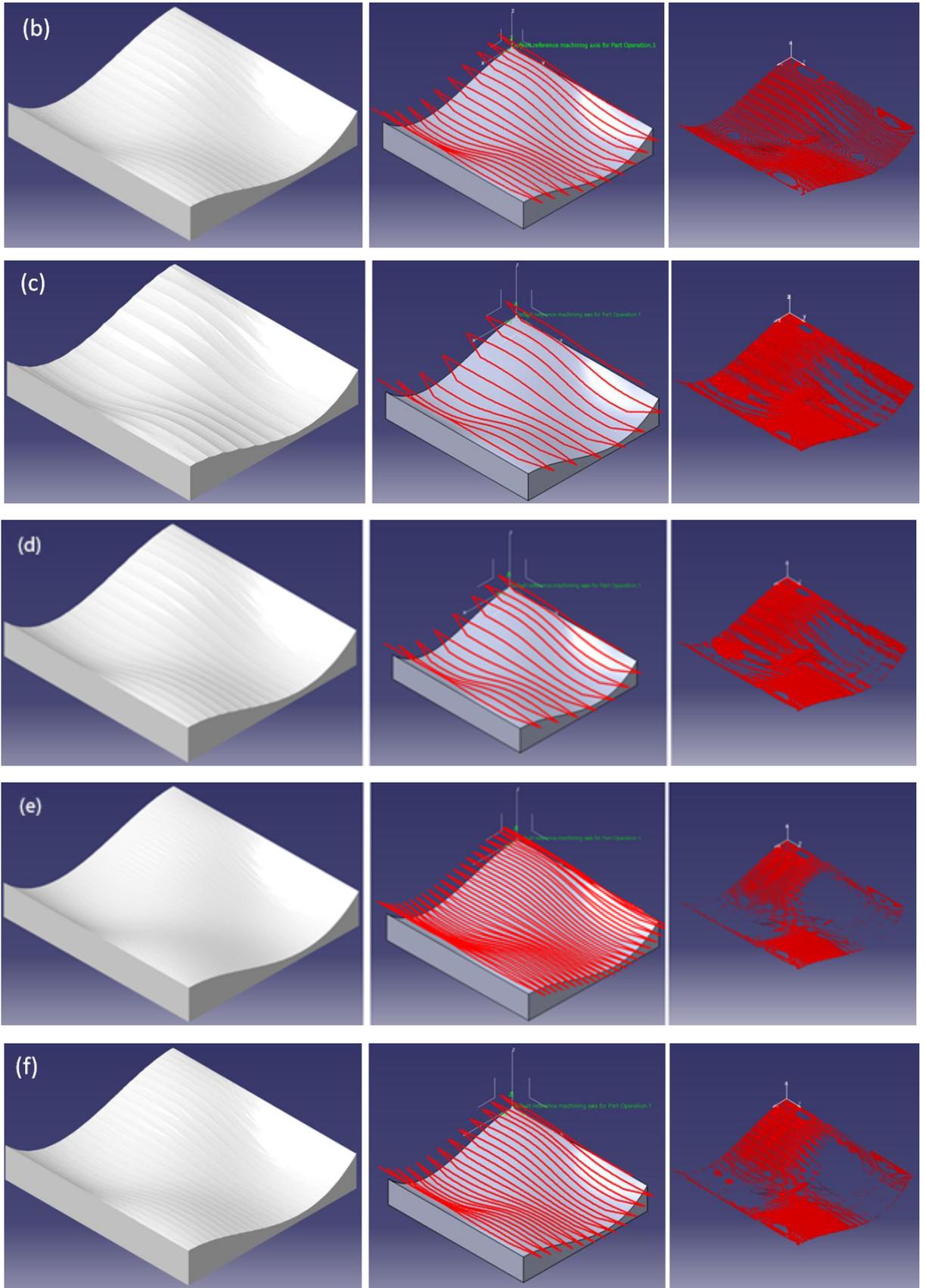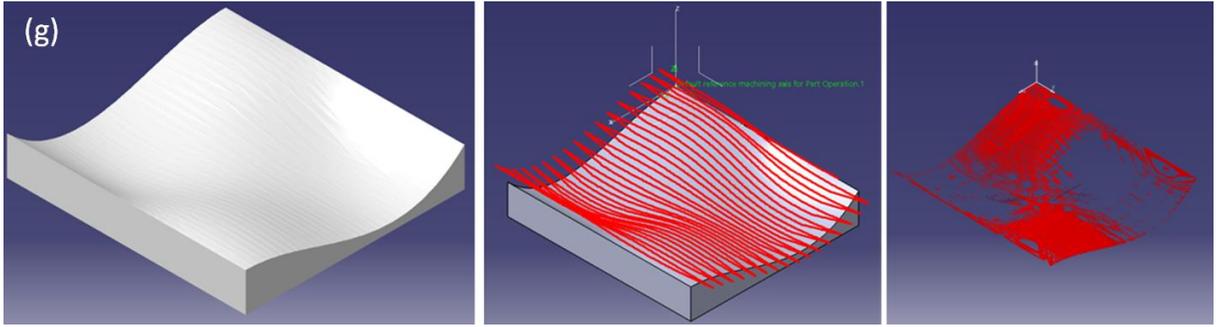
Figure 6.6: CAM outputs, tool paths and 3D maps using optimal parameters for the benchmark sculptured surface SS-1:(a) MOVEGA, (b) nvMOGA, (c) MOALO, (d) MOMVO, (e) MOGWO, (f) MODA, (g) evMOGA.

Table 6.9: Optimal Pareto3D tool path parameter values for best runs of MOEAs in ascending classification with regard to the benchmark sculptured surface 1 (SS-1).

a/a best	MOEA	Pareto3D	Best tool path parameters				
			Tool	Stepover (D%)	Lead (deg)	Tilt (deg)	MaxDstep (mm)
1	MOVEGA	**0.512070**	2	22.378	20.053	0.112	1.386
2	nvMOGA	0.515699	2	21.787	20.113	0.085	1.392
3	MOALO	0.521878	2	24.786	21.082	0.736	1.299
4	MOMVO	0.522748	2	10.000	20.000	0.874	0.700
5	MOGWO	0.530664	2	37.696	21.870	4.954	1.250
6	MODA	0.535120	2	17.825	20.262	1.155	1.397
7	evMOGA	0.565565	2	13.295	20.687	0.760	1.372

Table 6.10: Machining simulation outputs with regard to MOEAs' best runs for benchmark sculptured surface 1 (SS-1).

a/a best	MOEA	CAM simulation outputs					
		t_m (sec)	t_T (sec)	# passes	V_{Rem} (mm³)	$\overline{\delta} + \overline{h}$ (mm)	# points
1	MOVEGA	291.83	345.50	21	337.233	0.141	2400
2	nvMOGA	292.53	345.57	21	345.300	0.147	2432
3	MOALO	266.33	313.51	19	417.170	0.175	2329
4	MOMVO	183.68	215.82	13	480.543	0.223	1648
5	MOGWO	630.41	748.54	45	257.247	0.069	10137
6	MODA	353.07	416.02	25	307.564	0.126	2901
7	evMOGA	475.93	560.47	33	295.074	0.094	3942
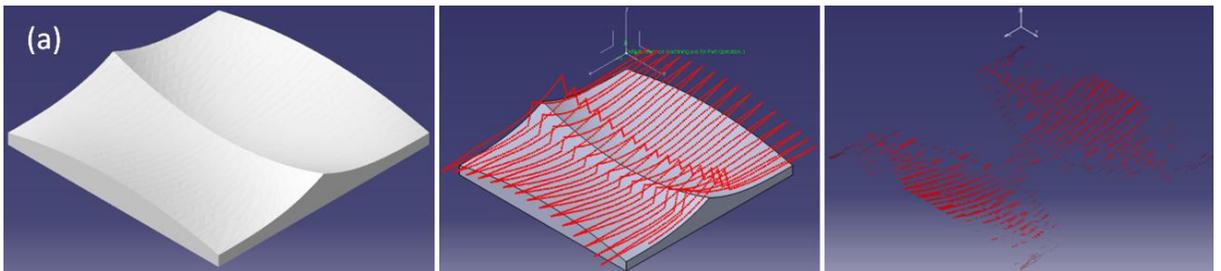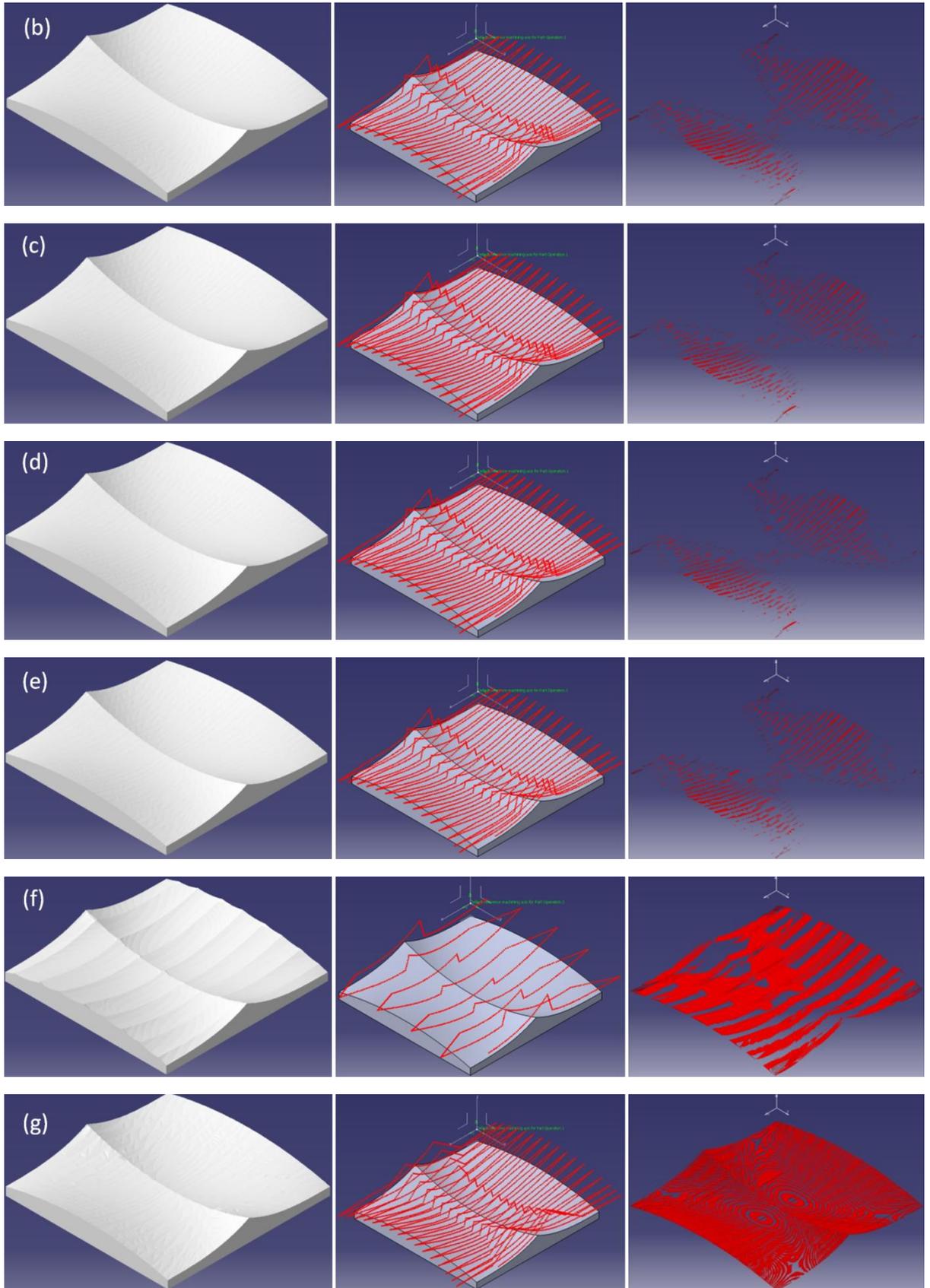
Figure 6.7: CAM outputs, tool paths and 3D maps using optimal parameters for the benchmark sculptured surface SS-2:(a) MOVEGA, (b) nvMOGA, (c) MOMVO, (d) MOGWO, (e) MODA, (f) MOALO, (g) evMOGA.

Table 6.11: Optimal Pareto3D tool path parameter values for best runs of MOEAs in ascending classification with regard to the benchmark sculptured surface 2 (SS-2).

a/a best	MOEA	Pareto3D	Best tool path parameters				
			Tool	Stepover (D%)	Lead (deg)	Tilt (deg)	MaxDstep (mm)
1	MOVEGA	**0.852960**	1	10.668	15.083	0.006	1.999
2	nvMOGA	0.853417	1	10.745	15.001	0.089	1.998
3	MOMVO	0.855975	1	10.000	15.000	0.000	1.781
4	MOGWO	0.856093	1	10.714	15.000	0.177	2.000
5	MODA	0.861344	1	11.375	15.000	0.449	2.000
6	MOALO	0.879842	1	42.127	17.644	0.000	2.000
7	evMOGA	0.982634	2	13.295	20.687	0.760	1.372

Table 6.12: Machining simulation outputs with regard to MOEAs' best runs for benchmark sculptured surface 2 (SS-2).

a/a best	MOEA	CAM simulation outputs					
		t_m (sec)	t_T (sec)	# passes	V_{Rem} (mm³)	$\bar{\delta} + \bar{h}$ (mm)	# points
1	MOVEGA	559.59	694.86	31	047.118	0.0965	2646
2	nvMOGA	559.73	695.03	31	047.890	0.0962	2646
3	MOMVO	597.62	740.53	33	038.783	0.0939	3175
4	MOGWO	560.54	560.54	31	046.852	0.0967	2631
5	MODA	526.76	526.76	29	098.116	0.0998	2459
6	MOALO	165.43	204.9	09	349.560	0.3380	0648
7	evMOGA	630.48	818.83	33	403.860	0.1130	3046
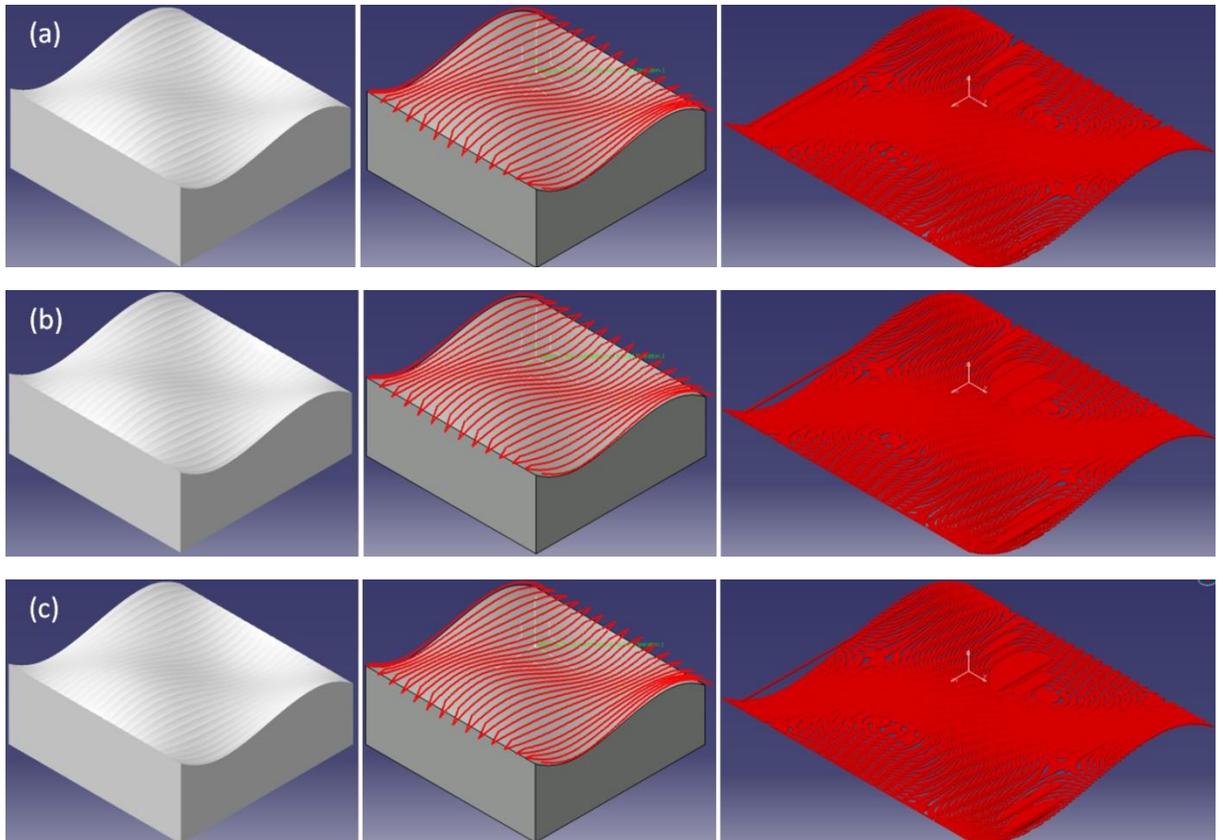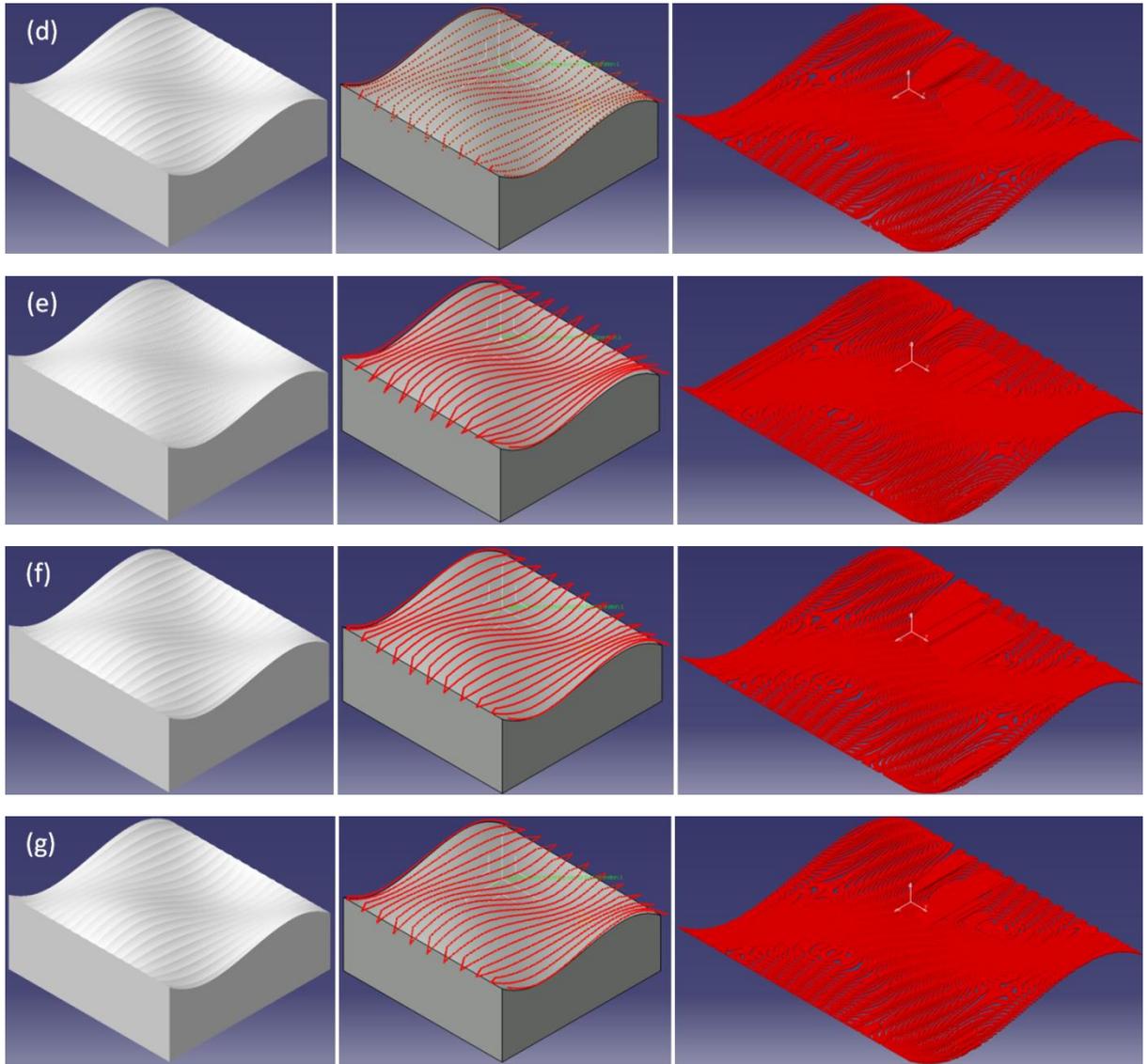
Figure 6.8: CAM outputs, tool paths and 3D maps using optimal parameters for the benchmark sculptured surface SS-3:(a) MOVEGA, (b) nvMOGA, (c) MOMVO, (d) MOALO, (e) MOGWO, (f) evMOGA, (g) MODA.

Table 6.13: Optimal Pareto3D tool path parameter values for best runs of MOEAs in ascending classification with regard to the benchmark sculptured surface 3 (SS-3).

a/a best	MOEA	Pareto3D	Best tool path parameters				
			Tool	Stepover (D%)	Lead (deg)	Tilt (deg)	MaxDstep (mm)
1	MOVEGA	**1.086005**	2	34.649	19.995	0.040	1.011
2	nvMOGA	1.086074	2	33.802	20.000	0.005	1.021
3	MOMVO	1.086620	2	32.111	20.000	0.000	1.000
4	MOALO	1.125946	2	45.000	15.000	1.237	2.287
5	MOGWO	1.155664	1	41.270	17.075	0.000	1.000
6	evMOGA	1.207695	2	44.476	16.867	1.762	1.000
7	MODA	1.212903	2	44.600	19.800	0.935	1.270

Table 6.14: Machining simulation outputs with regard to MOEAs' best runs for benchmark sculptured surface 3 (SS-3).

a/a best	MOEA	CAM simulation outputs					
		t_m (sec)	t_T (sec)	# passes	V_{Rem} (mm³)	$\bar{\delta}+\bar{h}$ (mm)	# points
1	MOVEGA	346.28	375.48	27	249.228	0.2253	3046
2	nvMOGA	346.29	375.49	27	248.798	0.2128	3000
3	MOMVO	373.50	402.72	29	242.692	0.2066	3285
4	MOALO	274.28	297.63	21	241.306	0.2806	0941
5	MOGWO	306.10	340.59	23	254.653	0.2134	3285
6	evMOGA	284.79	309.08	21	264.528	0.2936	2359
7	MODA	285.46	309.81	21	256.563	0.2899	1716
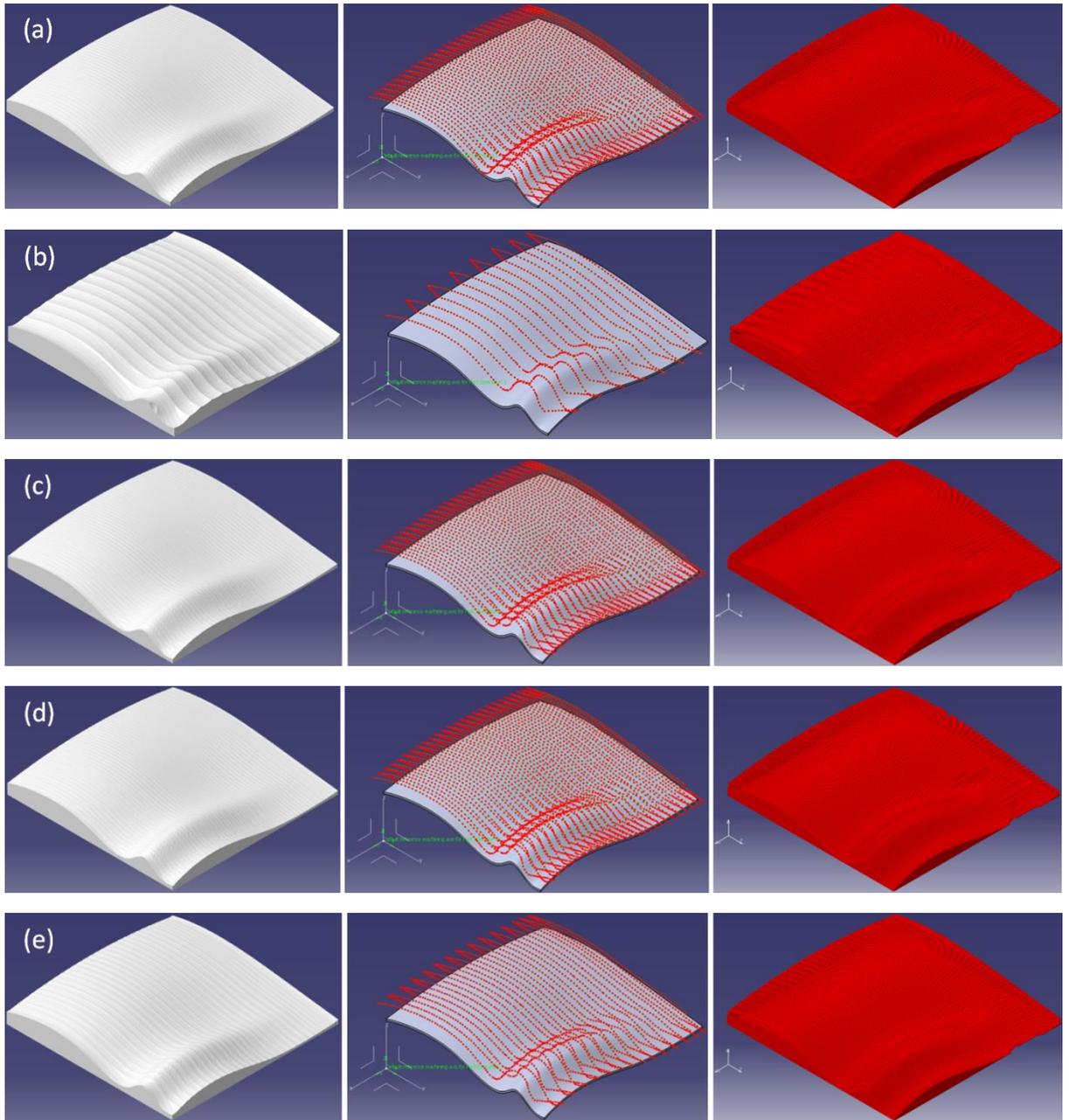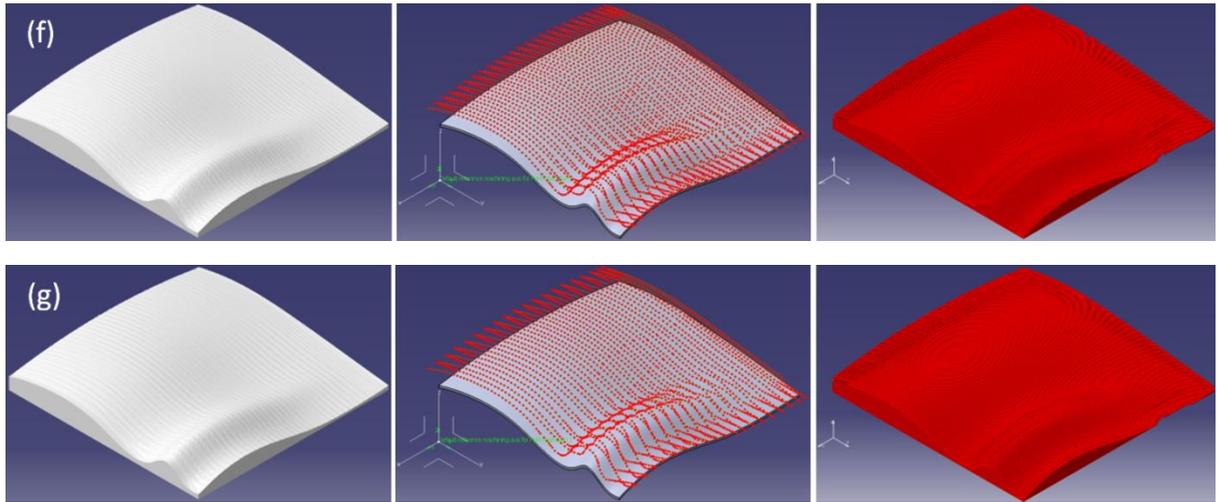
Figure 6.9: CAM outputs, tool paths and 3D maps using optimal parameters for the benchmark sculptured surface SS-4:(a) MOVEGA, (b) MOMVO, (c) nvMOGA, (d) MOGWO, (e) MOALO, (f) MODA, (g) evMOGA.

Table 6.15: Optimal Pareto3D tool path parameter values for best runs of MOEAs in ascending classification with regard to the benchmark sculptured surface 4 (SS-4).

a/a best	MOEA	Pareto3D	Best tool path parameters				
			Tool	Stepover (D%)	Lead (deg)	Tilt (deg)	MaxDstep (mm)
1	MOVEGA	**0.410083**	2	10.084	39.978	1.649	2.497
2	MOMVO	0.412033	2	33.412	32.636	2.435	2.317
3	nvMOGA	0.412234	2	10.111	39.971	1.820	2.499
4	MOGWO	0.415796	2	10.000	40.000	2.626	2.500
5	MOALO	0.416677	2	17.797	38.534	1.688	1.930
6	MODA	0.437219	2	10.453	39.628	1.988	2.416
7	evMOGA	0.499220	2	13.295	40.000	1.077	2.327

Table 6.16: Machining simulation outputs with regard to MOEAs' best runs for benchmark sculptured surface 4 (SS-4).

a/a best	MOEA	CAM simulation outputs					
		t_m (sec)	t_T (sec)	# passes	V_{Rem} (mm³)	$\overline{\delta} + \overline{h}$ (mm)	# points
1	MOVEGA	476.61	540.06	53	229.651	0.0550	2735
2	MOMVO	152.50	173.68	17	235.133	0.1680	0864
3	nvMOGA	477.33	540.88	53	230.800	0.0565	2730
4	MOGWO	478.54	542.25	53	230.794	0.0524	2738
5	MOALO	279.08	317.33	31	231.641	0.0935	1799
6	MODA	467.10	528.90	51	231.590	0.0575	2722
7	evMOGA	391.30	443.77	41	232.608	0.0750	2277

With reference to the overall results presented for exploring the potentials of modern algorithms to optimise the sculptured surface CNC machining problem, MOVEGA achieved the best machining simulation outputs and multi-objective optimisation criterion in comparison to the rest of the algorithms-competitors at least for their given algorithm-specific parameter settings. In order to conduct the study on a common basis, the optimisation problem was formulated here by adopting regression models resulting from the individual designs of experiments reported in Chapter 3. Although this technique provides a common ground to study the abilities of the algorithms tested and

assists on the identification of statistically significant differences in the results they achieve, the problem's representation is not as much sufficient and reliable as needed, compared to the philosophy of modelling it under the philosophy presented in this PhD thesis (see Chapter 3 - problem formulation). In order to rigorously compare these algorithms and draw crystal clear conclusions about their functional behaviour and the results they can obtain, they should be integrated to the methodology's optimisation framework as it has been presented, supported and implemented using the MOVEGA. However, such an attempt goes far beyond the boundaries and research scope of this thesis, nevertheless it is envisioned as a major future perspective towards the enhancement or the integration of the methodology's current status with other intelligent systems to further extend its optimisation capabilities.

6.3 Methodology validation with process-related results from competing sculptured surface CNC machining strategies

This section reports the comparative observations among the results obtained by applying the proposed optimisation methodology for the generalized sculptured surface CNC machining problem and other similar methods dedicated to tool path planning and / or optimisation for multi-axis sculptured surface CNC machining. Some of the methods, even older, have already been applied to integrate computer-aided manufacturing systems that currently see service in industry while others, more recent, share similar functional principles with those developed in the past, for solving the sculptured surface CNC machining problem. The process of comparing the results referring to the different tool path planning and optimisation methodologies is far more challenging here, since results are obtained by taking series of measurements conducted to physical machining parts comprising sculptured surfaces which designed according to literature attributes, studied and manufactured for validating the methodology proposed in the PhD thesis. To achieve unbiased and rigorous comparisons among results, same resources and/or tooling as well as validation approaches were applied wherever it was feasible. Machining simulations for additional comparative analysis between the algorithm integrating the proposed methodology (MOVEGA) and the same algorithm without the application of viral operators (nvMOGA) was not performed since it has already been proved on several occasions that the former is superior to the latter (Figs 6.2 to 6.5 and Tables 6.9 to 6.15). By considering the overall experimental observations for validating the proposed methodology against other methods it is deduced that the proposed methodology not only is competitive but also it implicitly accounts for the behaviour of adhering to noticeable characteristics related to the mechanics of multi-axis CNC machining processes, despite its stochastic nature.

6.3.1 Comparison to tool path generation / optimisation methods based on machining-simulated outputs

A first comparison among the results obtained by the different tool path generation / optimisation methods is referred to the average scallop height characterizing surface machining accuracy. The comparative analysis is referred to "Inclined tool - ITM", "Principal axis – PAM" and "Multi-point machining – MPM" tool path generation / optimisation methods. Results from simulations using the aforementioned methods have been given by Warkentin et al. (1997) where a widely examined benchmark sculptured part was designed to play the role of an impact case to allow for rigorous comparisons among the results of the methods. The benchmark sculptured part has a significant research timespan from 1997 to 2017 (Lu et al. 2017, Chen et al, 2017, Gan et al. 2016, Xu et al. 2010, Warkentin et al. 2000, Rao et al. 1997).

The benchmark sculptured surface representing the impact case for all these tool path planning / optimisation methods is a second-order, open-form parametric sculptured surface fully defined by Equation 6.2 and depicted in Figure 6.10. This surface is designated in the thesis as "SS-5".

$$S(u,v) = \begin{bmatrix} -94.4 + 88.9v + 5.6v^2 \\ -131.3u + 28.1u^2 \\ a_1 + a_2 \end{bmatrix}, \begin{cases} a_1 = 5.9(u^2v^2 + u^2v) - 3.9v^2u + 76.2u^2 \\ a_2 = 6.7v^2 - 27.3uv - 50.8u + 25v + 12.1 \end{cases} \qquad \text{Eq. 6.2}$$
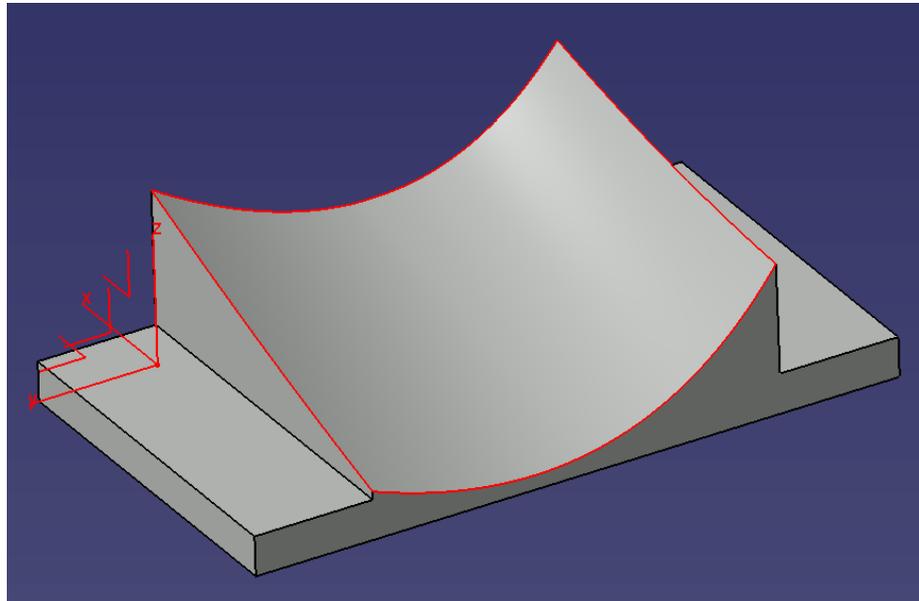
Figure 6.10: The 2nd order, open-form parametric benchmark sculptured surface (SS-5).

Warkentin (1997) compared his tool path planning / optimisation methodology "multi-point machining – MPM" to 3-axis ball end-milling, "Inclined tool - ITM" and "Principal axis – PAM". To validate his results, he conducted machining simulation experiments using the benchmark sculptured surface illustrated in Figure 6.10 while studying the average scallop height for a range of tool path intervals (stepover values), from 1mm to 10mm with reference to a D16 Rc3 toroidal cutting tool. The same work was carried out to investigate the average scallop height on machining-simulated CAM outputs by applying the proposed optimisation methodology. Ten algorithmic evaluations were performed by simulating the same cutting tool and maintaining constant tool path intervals from 1mm to 10mm while trying to find optimal values for the rest of tool path parameters, lead angle, tilt angle, and maximum discretization step. Average scallop height was computed after obtaining a significant number of pick-point measurements on each of the CAM output's surface depending on the noticeable features indicating the error characterizing the surface. Cutting tolerance for the machining simulations was equal to ±0.05 for both excess and gouged material while the ranges for the rest of tool path parameters were from 1◦ to 5◦ for lead angle, 0◦ to 1◦ for tilt angle and 0.06mm to 0.10mm for MaxDstep. For the simulations referring to the tool path intervals from 1mm to 4mm an offset equal to 0.2mm was applied to the tool path in order to ensure that machining error owing to scallop height would be observable when examining the corresponding CAM outputs. Thereby, scallop height was computed by considering the results measured on profound scallops minus the offset value of 0.2mm. In general, the average scallop height obtained by applying the proposed optimisation methodology referring to all tool path intervals was found 93.21%, 65.82% and 12.48% lower than those reported for ITM, PAM and MPM respectively. Table 6.17 summarizes the results for average scallop heights per each methodology and tool path interval tested, while Figure 6.11 gives the graphical depictions of these results. Figure 6.12 illustrates the resulting CAM outputs per tool path interval tested using the proposed optimisation methodology and optimal parameter values.

Table 6.17: Tabulated results of average scallop heights (µm) for ITM, PAM, MPM and proposed methodology (benchmark sculptured surface SS-5, cutting tool D16Rc3).

Tool pass interval (mm)	Inclined Tool - ITM	Principal Axis - PAM	Multi-point machining -MPM	Proposed methodology
1	02.10	0.40	0.40	0.40
2	07.50	0.60	0.60	0.50
3	15.30	0.70	0.90	0.80
4	27.50	1.30	1.20	0.90
5	49.40	2.70	1.60	1.30
6	67.50	6.00	3.00	2.80
7	100.5	12.9	5.30	3.60
8	132.7	21.9	9.50	8.40
9	162.2	37.5	16.2	13.1
10	282.1	84.3	27.0	25.7
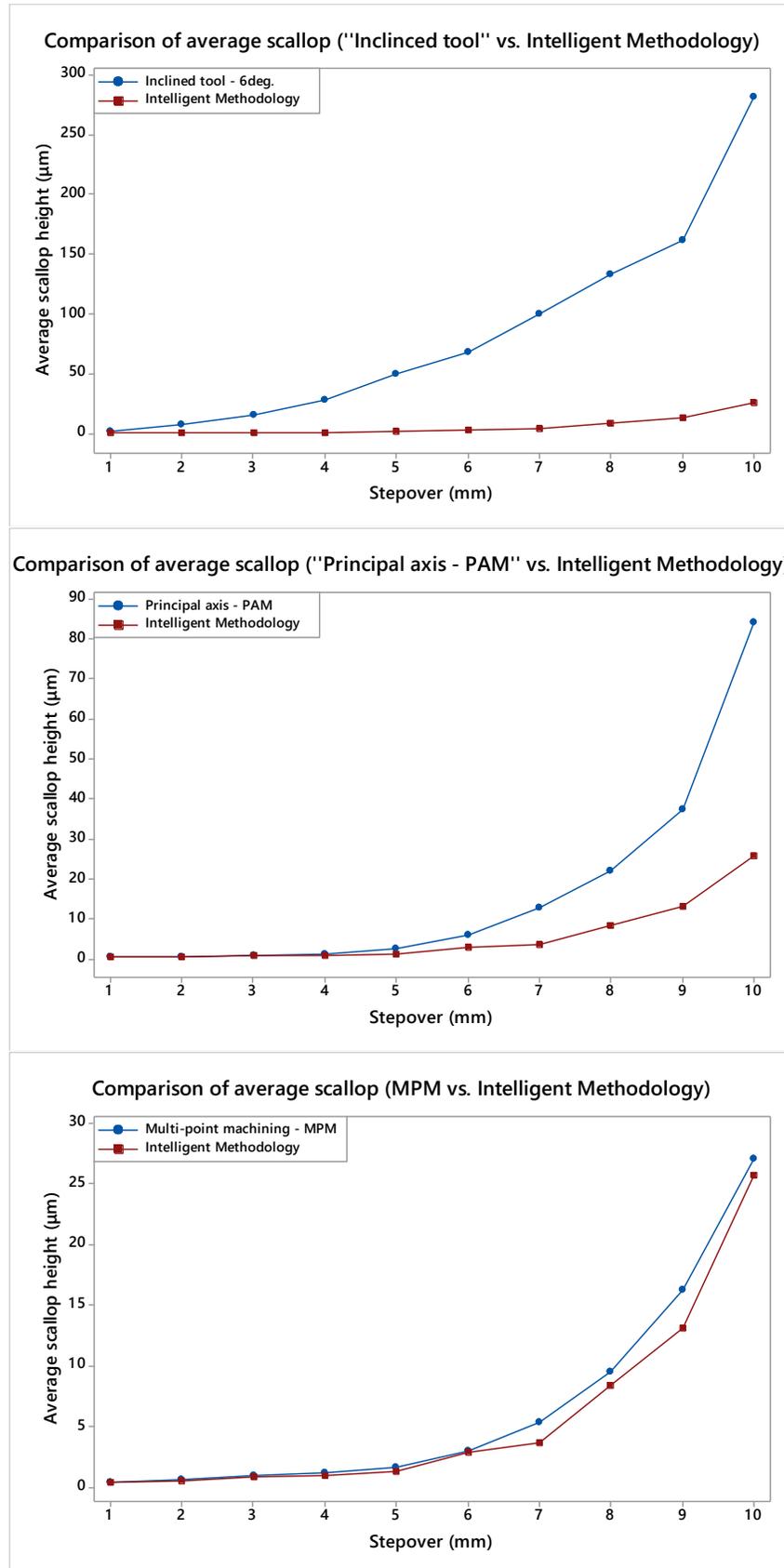
Figure 6.11: Comparative simulation results of average scallop height among the intelligent methodology and "Inclined Tool − ITM", "Principal axis − PAM" and "Multi-point machining − MPM" methods under constant tool path intervals (benchmark sculptured surface SS-5, cutting tool D16Rc3).

Stepover = 1mm (06.25% Ø16mm)

Stepover = 2mm (12.50% Ø16mm)

Stepover = 3mm (18.75% Ø16mm)

Stepover = 4mm (25.00% Ø16mm)

Stepover = 5mm (31.25% Ø16mm)

Stepover = 6mm (37.50% Ø16mm)
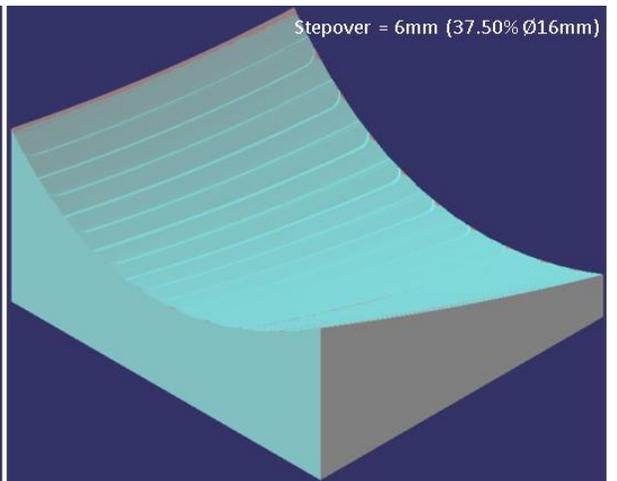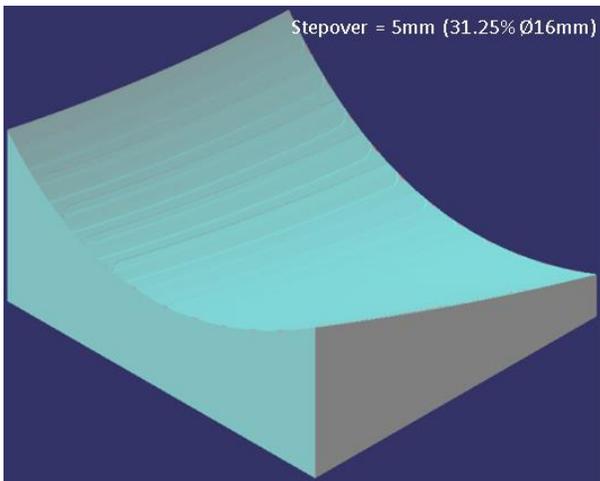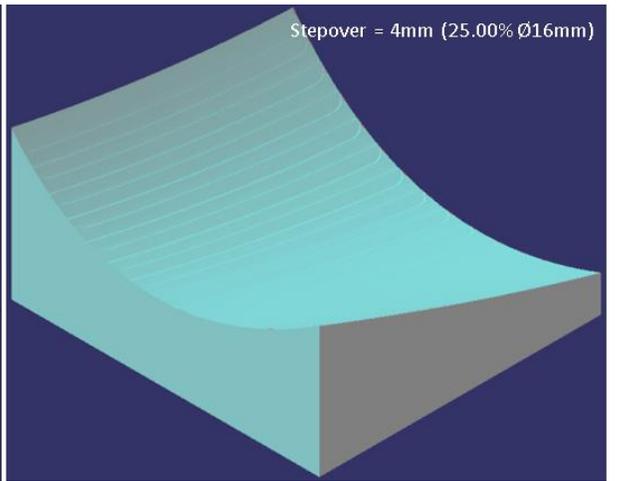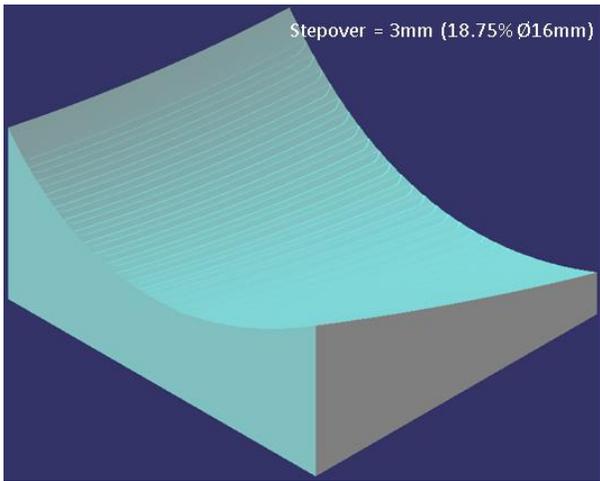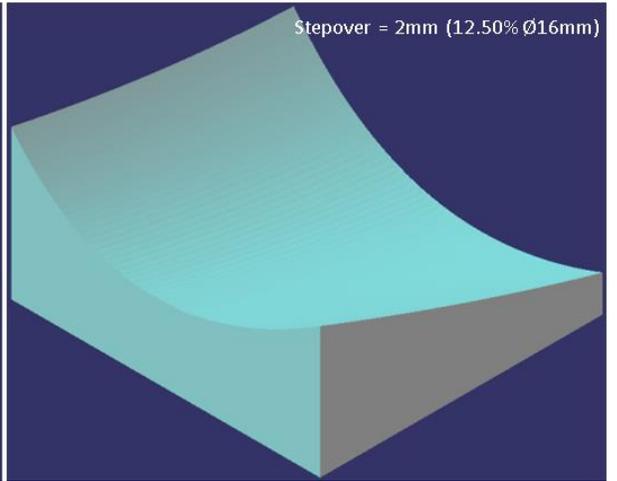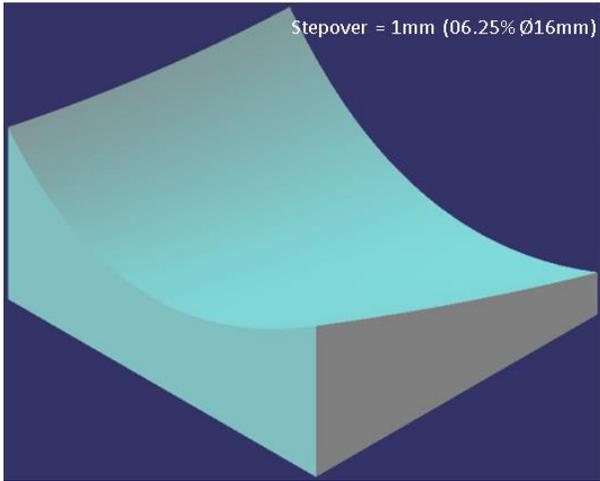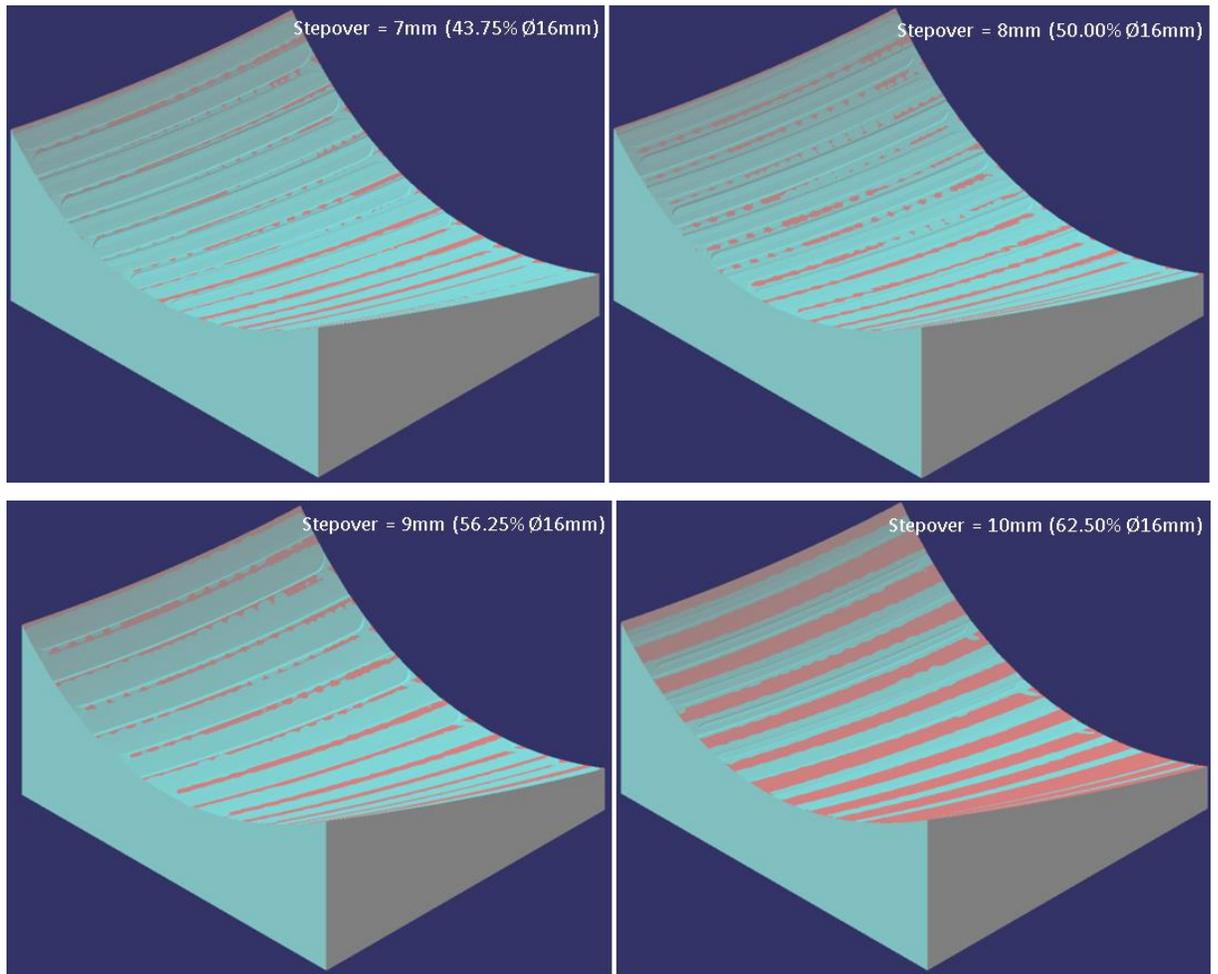
Figure 6.12: Simulated CAM outputs for examining scallop height using the intelligent methodology (benchmark sculptured surface SS-5, cutting tool D16Rc3).

By examining the graphical results depicted in Figures 6.11 and 6.12, with reference to Table 6.17, it is clear that the proposed methodology produces tool paths capable of maintaining surface quality for any tool path interval, and especially for those that side step spans from 5mm to 10mm. Specifically for these tool paths, surface quality is dramatically degraded since the distances between the successive tool trajectories increase, thus, increasing scallop height as well.

The performance of the proposed method compared to ITM starts to be observed by the tool path interval of 2mm and beyond, which is quite reasonable since the ITM method is applied with fixed lead angles whose values should be given prior to the computations necessary for generating the tool path and without considering local surface curvatures as the tool removes material in feed direction resulting in poor surface quality even for low tool path intervals. This observation is evident from the relative graph presenting the irregular increase in average scallop height by ITM method.

The performance of the proposed method compared to the PAM method is observable for tool path intervals greater than, or equal to that of 5mm. While considering local curvatures, PAM method ignores feed direction which provides important information for computing tool paths when it comes to such methods since lead angle's adjustment is based on feed vector. Instead, the proposed optimisation methodology does not need this kind of information since it is applied to already existing tool path planning scenarios whilst local curvatures corresponding to the various tool positions are calculated under an already defined feed direction.

The performance of the proposed method compared to the MPM method is observable for the tool path intervals corresponding to the range of values from 7mm to 10mm, with respect to the toroidal cutting tool geometry used, while the shape of scallops is quite uniform. The MPM method considers both the local curvatures of a sculptured surface (yet implicitly) and feed direction. However, results from computations referring to the various tool orientations per surface regions, as well as the multi-point contact requirement, may not always maintain efficiency in the case of different tool path intervals. In addition, the second cutting tool contact-point that needs to be determined for the successful implementation of the MPM method (Warkentin et al., 2000), might either yield a significant surface deviation regarding the ideal surface, or may not even exist.

The optimisation methodology proposed in the thesis does not require finding optimal tool orientations with respect to local curvatures, but inherently imposes them as an optimisation side-effect, based on the tool path strategy under the stochastic requirement to minimize the machining error for the problem's generalized solution. Whether the advantageous cutting tool positions on the surface result in multi-point contact, depends on the optimal values for tool path parameters according to the selected machining strategy.

Nevertheless, it is reasonable to assume that in order for the generalized result to satisfy productivity requirement in addition to quality, it is expected that optimal cutting tool positions generated by the optimisation methodology for sculptured surface CNC machining, should mostly achieve multi-point contacts to account for wider machining strip widths - MSW.

The following section comments on the requirement of achieving multi-point cutting tool contacts with a given sculptured surface and presents the characteristics of the multi-point tool contact when it comes to convex and concave surface regions.

6.3.2 Effect of stochastically optimised tool path planning parameters on the multipoint tool-surface contact

By accepting that the contact between a cutting tool and a given sculpted surface is rather of a "multi-point" nature in the majority of cutting tool positions, a stochastic optimization methodology at its minimum ought to eventually result (even implicitly) to multi-point tool paths so as to increase machining strip width - MSW wherever is possible, with reference to the geometrical properties of the surface and cutting tool. In other words, any methodology developed for creating and/or optimising sculptured surface machining tool paths should adhere to the multi-point cutting tool contact effect as a key element to maximize efficiency of its tool path strategy in the complex case of the simultaneous multi-axis sculptured surface CNC machining (Sharma et al. 2018, Chen et al. 2017, Gan et al. 2016, He et al. 2015, Fan et al. 2013, Warkentin et al. 2000).

In addition to the regular scallop material left between subsequent cutting passes, the multi-point contact also leaves excess material underneath the cutting tool. This excess material is acceptable as long as it satisfies the predetermined cutting tolerance. Figure 6.13 depicts the six cases for the multi-point contact between a cutting tool and a sculptured surface, three referring to a concave surface (Fig.6.13a-6.13c) and three referring to a convex surface (Fig.6.13d-6.13f). According to Figure 6.13a the machining error will form owing to scallops among subsequent tool passes and material left between the two cutting points underneath the tool. Figure 6.13b suggests that machining error will form only between the two cutting points underneath the tool and Figure 6.13c implies that machining error is the result of the combination of the two aforementioned cases. The same also goes for the cases of multi-point tool contact with a convex surface (Fig.6.13d-6.13f). Note that these cases are also valid for flat end-mills (Lu et al. 2017).

In the case of multi-point tool contact, machining error formation is an outcome of the error distribution curve which is the projection of the cutting tool's characteristic curve, based on its inclined orientation on the surface whilst is in the form of a "W", at least as regards convex surfaces. The error distribution curve characterizes the degree of the geometrical matching between the cutting tool's and the ideal surface's different geometries, it is used by several researchers to estimate machining strip width - MSW (Figure 6.14).
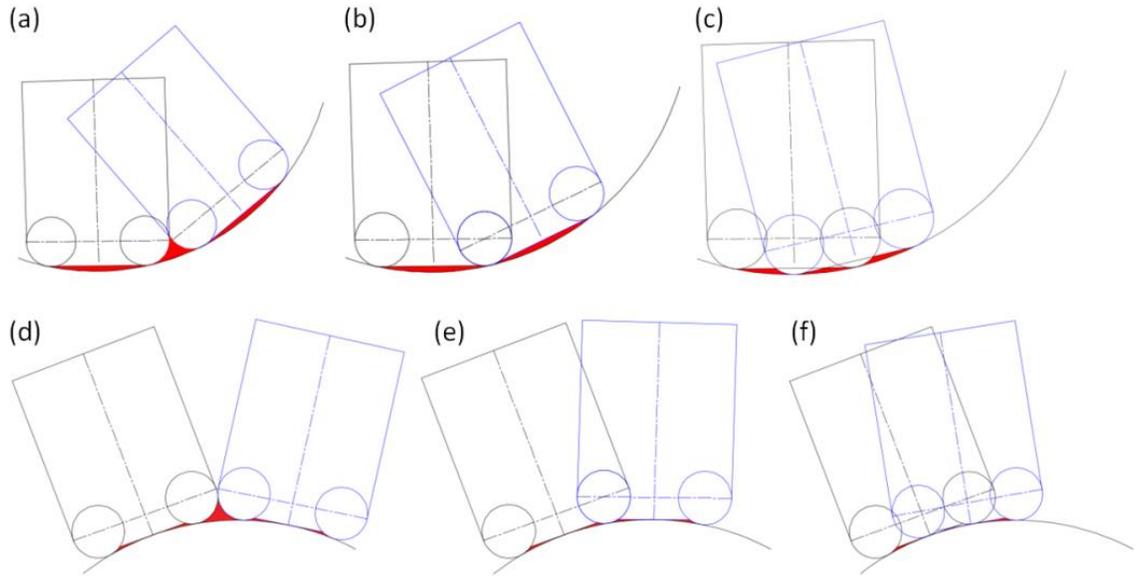
Figure 6.13: Resulting machining error owing to multi-point tool contact for concave and convex sculptured surfaces.
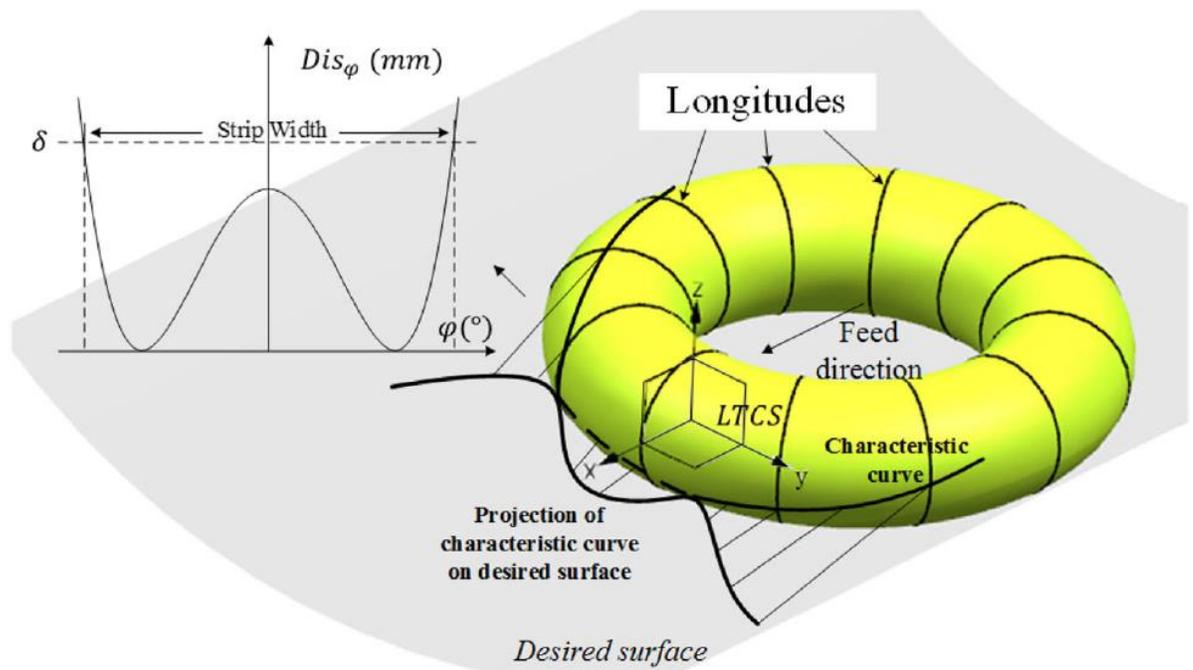
Figure 6.14: Machining error distribution curve and resulting machining strip width by applying a toroidal end-mill to machine a convex sculptured surface (Chen et al. 2017).

In order to examine the characteristics of the optimised tool paths under the perspective of the multi-point contact between the tool and the surface, measurements were taken on four discrete cross-sections of the SS-5 surface in the case of the 50% tool path interval given the cutting tool diameter

of 16mm.   Based on Warkentin's research (Warkentin 1997, Warkentin et al. 2000) four cross-sections were examined at X = -5mm, X = -30mm, X = -60mm and X = -90mm. The results of the measurements obtained using 1 mm as the measuring step are shown in Figure 6.15 where both multi-point error and scallop height are visible. Multi-point contact errors are distinguished in the graphs of Figure 6.15 as low and wide whereas subsequent scallops owing to tool path interval are observed from the peaks which are taller than multi-point contact's error.

The magnitudes of these errors, multi-point contact and scallop height are affected by a number of factors such as the tool path's cutter location topologies, the local curvatures of these locations, the tool path interval, lead and tilt angles and maximum discretization step. Other important aspects affecting these measurements are the selected value for the measuring step and the topologies where the measurements are taken for further evaluation. In order to ensure that an adequate number of measurements will be taken for the scope of multi-point error examination while maintaining low processing time, the value of 1mm for the measuring step was decided according to the benchmark sculptured surface SS-5 length of 103.2mm.

The error is more noticeable in the areas where abrupt changes in curvature occur as the cutting tool removes material from the part. In open-form surfaces with low curvature variation the effect of multi-point machining error may not be noticeable enough. The results from the measurements graphically illustrated in Figure 6.15 show obvious indications of the multi-point tool contact with the machining-simulated benchmark sculptured surface SS-5. Machining error distribution follows the "W" trend to almost the entire measuring space referring to all four cross-sections, X = -5mm, X = -30mm, X = -60mm and X = -90mm. A reasonable emphasis is given to cross-sections X = -30mm, X = -60mm and X = -90mm where the tool has already left behind the approaching region where early tool positioning is produced (i.e., from X=0mm to X=-10mm) and moves towards the main surface region until its departure after X = -90mm. Obviously the surface region between X=-5mm and X=-90mm contains almost all successful tool positions produced by the tool path generation and therefore a profound multi-point machining error reasonably characterizes this surface portion. Note that the scale for presenting the measurements also affects the graphical illustrations referring to error owing to multi-point tool contact with the surface. If the error was examined using a narrower scale, i.e. ±0.025mm the overall effect would be more noticeable. However the scale ±0.050mm has been deemed reasonable to graphically depict the resulting error given its magnitude, despite that the analogous illustrations in the work of Warkentin et al. (2000) have been reported using a larger scale, equal to ±0.100mm. Such a large scale was not considered in order to avoid unsuccessful depictions of the multi-point tool contact effect.

According to the comments made above and the illustrations presented in Figure 6.15 it can be argued that the optimal tool paths the proposed optimisation methodology formulates, adhere to the standard multi-axis surface machining behaviour despite the stochastic nature while they share much of the properties of multi-point machining for which they implicitly account for. As a result, optimally formulated tool paths by the implementation of the proposed methodology are expected to present wide enough machining strips, even though this objective has not been established as an optimisation criterion in advance. This is achieved because the algorithm prompts CAM software functions to affect the cutting tool trajectory of a standard tool path to increase efficiency.

(a) Virtual measurements for cross-section X=-5mm (Stepover: 50% tool diameter)

(b) Virtual measurements for cross-section X=-30mm (Stepover: 50% tool diameter)
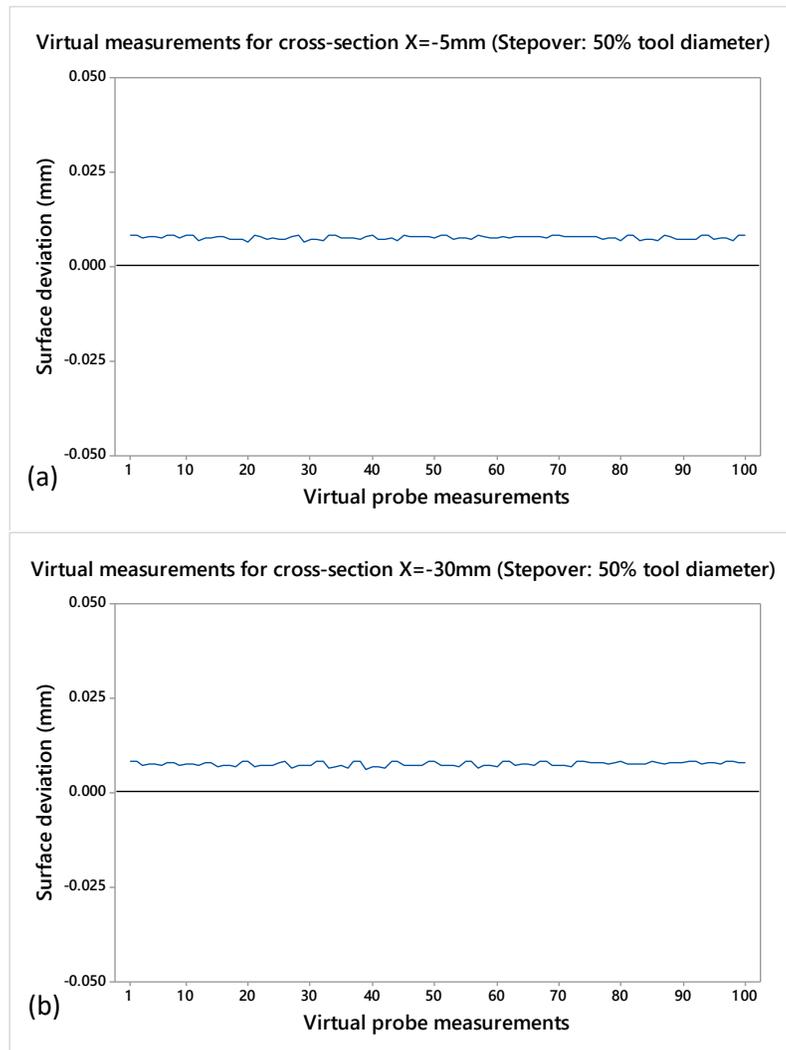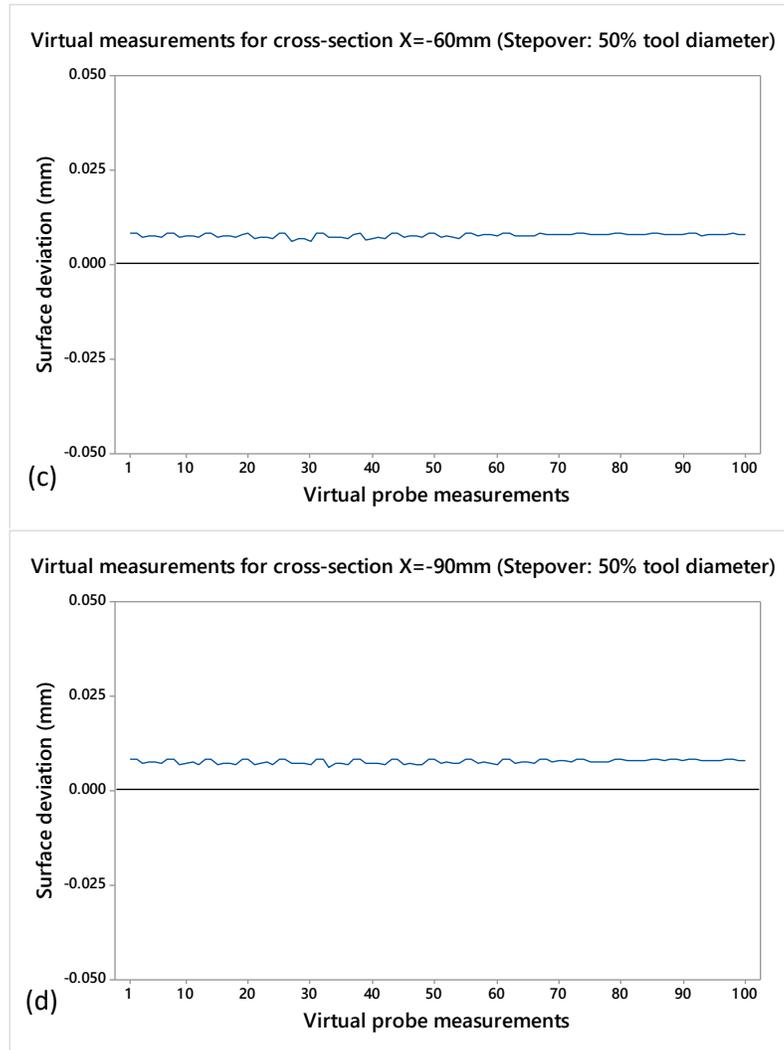
Figure 6.15: Machining error distribution curve and machining strip width for a toroidal end-mill and a convex sculptured surface.

6.3.3 Comparison to tool path generation / optimisation methods based on actual CNC machining results

The results presented in this section are thoroughly related to actual CNC machining operations conducted by implementing the proposed methodology for optimising the generalized sculptured surface CNC machining problem and others corresponding to the same problem yet, under a different problem formulation philosophy. All methods have already been reviewed in Chapter 2 whilst all their outputs correspond to benchmark sculptured surfaces SS-1, SS-2 and SS-5.

Machining operations of benchmark parts as well as corresponding quality inspections per impact case were carried out at Hellenic Aerospace Industry – H.A.I. (http://www.haicorp.com). The FOOKE

Endura® 5-axis gantry-type CNC machining center equipped with the SIEMENS Sinumerik® 840D CNC controller was used for the machining of sculptured surfaces. Al-5083 was selected as raw material for machining SS-1 and SS-2 whilst Al 7050 T-7451 was selected as raw material for machining SS-5. No particular reason was led to the different material selection for machining the parts, besides no specific working material was mentioned by the various comparative tool path planning / optimisation methods. This experimental effort was conducted for validating the successful application of the proposed research against others and is accompanied by a document substantiating the *research validation* with a reference number printed as a common practice for exposing public documents. An electronic copy of this document is available to Appendix A.

The various tool path planning / optimisation methods are mentioned in the following text as the results obtained by the implementation of the proposed methodology are compared to those cited by mentioning their corresponding references. The results will be examined per benchmark sculptured surface since the various tool path planning / optimisation methods do not provide exactly the same outputs for full comparative analysis. To cover all experimental cases, the results presented for the proposed methodology refer to all these alternative performance metrics stated by the comparative methods.

6.3.3.1 Benchmark sculptured surface SS-5

The results obtained from simulations and the actual cutting experiment performed on SS-5 were compared to the results obtained by Lu et al. 2017, Chen et al. 2017, Gan et al. 2016, Xu et al. 2010, Warkentin, Ismail and Bedi 200a, 2000b, Rao, Ismail and Bedi 1997 for the same benchmark sculptured surface. In the work of Warkentin et al. (2000) rigorous comparisons were made among simulations and machining results obtained by implementing the multi-point machining (MPM) method (Warkentin et al. 2000a, 2000b), against those obtained by Vickers and Quan (1989) and Rao et al. (1997) by employing the "inclined tool - ITM" and the "principal axis method – PAM" respectively.

Comparisons were made regarding simulation and actual cutting trends referring to the surface deviation examined on four 2D cross sections (X=-5mm, X=-30mm, X=-60mm and X=-90mm) for which a number of measuring points were taken using a DEA CMM machine tool model. With reference to these results, inclined tool method reaches the lowest surface deviation error as the tool approaches the surface contour (X=-5mm) and is estimated as being close to 0.040mm. However, the largest surface deviation value exceeds 0.1mm. In all four cross-sections examined, "inclined tool" method presents a highly non-uniform error. PAM significantly improves the machining operation by

maintaining low surface deviation in all four-cross sections. The value for this deviation was estimated equal to 0.010mm at X=-5mm whereas the largest surface deviation value was observed at X=-60mm, equal to ±0.040mm. The whole deviation fluctuates strongly throughout the trend of both simulation and experimental results with emphasis to X=-5mm and X=-90mm cross-sections which is reasonable since lead and tilt angles yield higher vibration magnitudes in these surface regions (tool approach and departure). MPM method's results shown further improvement mainly in terms of the scallop height magnitude. Indeed, according to Warkentin et al. (2000a, 2000b) sharpness of peaks representing the scallop magnitudes are hardly observed. Nevertheless minimum and maximum values for surface deviation stay at the same levels as those attained with PAM, yet, with the significant difference of presenting noticeable irregularities in terms of the error distribution especially at cross section X=-30mm. Warkentin's results for the measurements (Warkentin et al. 2000) in the four discrete cross-sections X=-5mm, X=-30mm, X=-60mm and X=-90mm are given in Appendix B along with those obtained by ITM and PAM for easy reference.

The work of Xu et al. (2010) contributes to the results reported above by simultaneously controlling tool path smoothness criterion and machining strip width maximization. The same benchmark surface (SS-5) was machined using a toroidal cutter with a torus radius 5 mm and insert radius 3 mm while cutting tolerance was set to 0.01 mm. The spindle speed used was 16000 rpm and feed was 5000 mm/min. The total time was around 1 min. The cross-sections selected for validating their methodology were at X=-5mm, X=-30mm and X=-60mm. By reviewing their results, it is deduced that the entire surface deviation is found under a zone of ±0.045mm with no profound peaks in scallops whereas the number of tool passes was found to be equal to 17 implying narrower machining strips. Gan et al. (2016) machined the benchmark surface SS-5 using the "mechanical equilibrium" method - MEM. According to their work, error distribution curves are examined to optimise matching for two contact points of a toroidal cutting tool on the surface. In their work a cutter with major radius R=6.5 mm and minor radius r=1.5 mm (D16Rc1.5) was used for machining the same benchmark surface. Gan et al. (2016) claimed that their strategy produced 14 subsequent machining strips with an average width equal to 8.21 mm whilst scallop height was found under the preset allowance of 0.01mm. Obviously machining strip width is maximized and the number of machining passes is reduced owing to the smaller rounded inserts (Rc 1.5mm) of the tool selected. Chen et al. (2017) implemented their "efficient convergent optimization - FCO" method to evaluate the matching degree between the tool and the theoretical surface. A cutter with major radius R=6.5 mm and minor radius r=1.5 mm (D16Rc1.5) was used for machining the same benchmark surface SS-5. Their strategy produced 12 subsequent machining strips whereas their average was equal to 9.5 mm. Although a noticeable smoothness was achieved throughout the junctures of strips created by their method, a

207

significant degradation of the surface's free-form profile is observed. Lu et al. (2017) tried to implement a "global optimisation" method using flat end-mills and two algorithms (differential evolution and sequence linear programming) to balance tool path smoothness and machining strip width by avoiding "step-by-step" computational methods for tool positioning. The results reported concern the machining strip width which was found equal to 8.74 mm using a D16 flat end-mill. 16 tool passes were obtained. By reviewing their results from a practical perspective, it is argued that flat end-mills do not facilitate their accurate tool orientation owing to their discontinuous contact between the machining surface and theoretically sharp corner. This uncertainty is suggested owing to the inability of applying the tangency criterion for flat end-mills, implying that a free-gouge and tangential contact cannot be ensured. As opposed to cutters with squared flutes (teeth) as flat end-mills, tools with rounded corners tend to maintain the uniform wear spread in longer cutting edge, leaving thus a much smoother surface finish result on a machined sculptured part.

The benchmark sculptured surface SS-5 was machined using the proposed tool path optimisation methodology. The optimisation ranges for tool path parameters were the same as those determined for the simulations to examine the average scallop height for the different stepover parameter settings. To present a rigorous optimisation process three tools were tested, D16Rc8, D16Rc0 and D16Rc3 corresponding to a D16 ball end-mill, D16 flat end-mill and D16 toroidal (filleted) end-mill respectively. The parameters recommended as optimal were implemented for the machining simulation and the actual cutting experiment using the benchmark surface SS-5. By applying a feed velocity V_f = 3000 mm/min under 12.000 rpm spindle speed n, machining time resulted to 1min-24sec. Table 6.18 tabulates the optimisation range (low-high) per tool path parameter as well as the optimal settings recommended by the proposed optimisation methodology.

Table 6.18: Tool path parameter bounds and optimal recommended values for the case of benchmark surface SS-5.

Benchmark surface	Levels	Tool	Stepover (%D)	Lead angle (deg)	Tilt angle (deg)	MaxDstep (x10^{-3} mm)
	Low	D16-Rc3	20	1	0	6
SS-5	High	D16-Rc0	45	5	1	10
	Optimal	D16-Rc3	41.729% (6.677 mm)	2.957	0.027	6.338 (x10^{-3})

Figure 6.16a illustrates the spindle setup during machining, Figure 6.16b the machining operation and Figure 6.16c the finished result. The simulated machining time was found in agreement with actual machining time given by the CNC unit. 15 smooth and uniformly distributed cutting strips were left on the actual cutting surface whilst their theoretical widths were computed during the machining

simulation by examining sequential pairs of two scallop lines. This way allowed for finding the real cutting strip widths without their overlaps. The average machining strip width was equal to 8.583 mm and their average overlap was 2.79 mm. The average machining strip width measured on the actual cut surface was estimated around 6.62 mm. The actual cut surface was examined at the four cross-sections with respect to the previous works reported above, X=-5mm, X=-30mm, X=-60mm and X=-90mm (Figure 6.17). In the simulation the test points were arranged in the same way as the measurement points taken by the CMM for the experimental results. According to the results the maximum deviation error does not exceed 0.026 mm and the minimum deviation equals 0.012 mm.
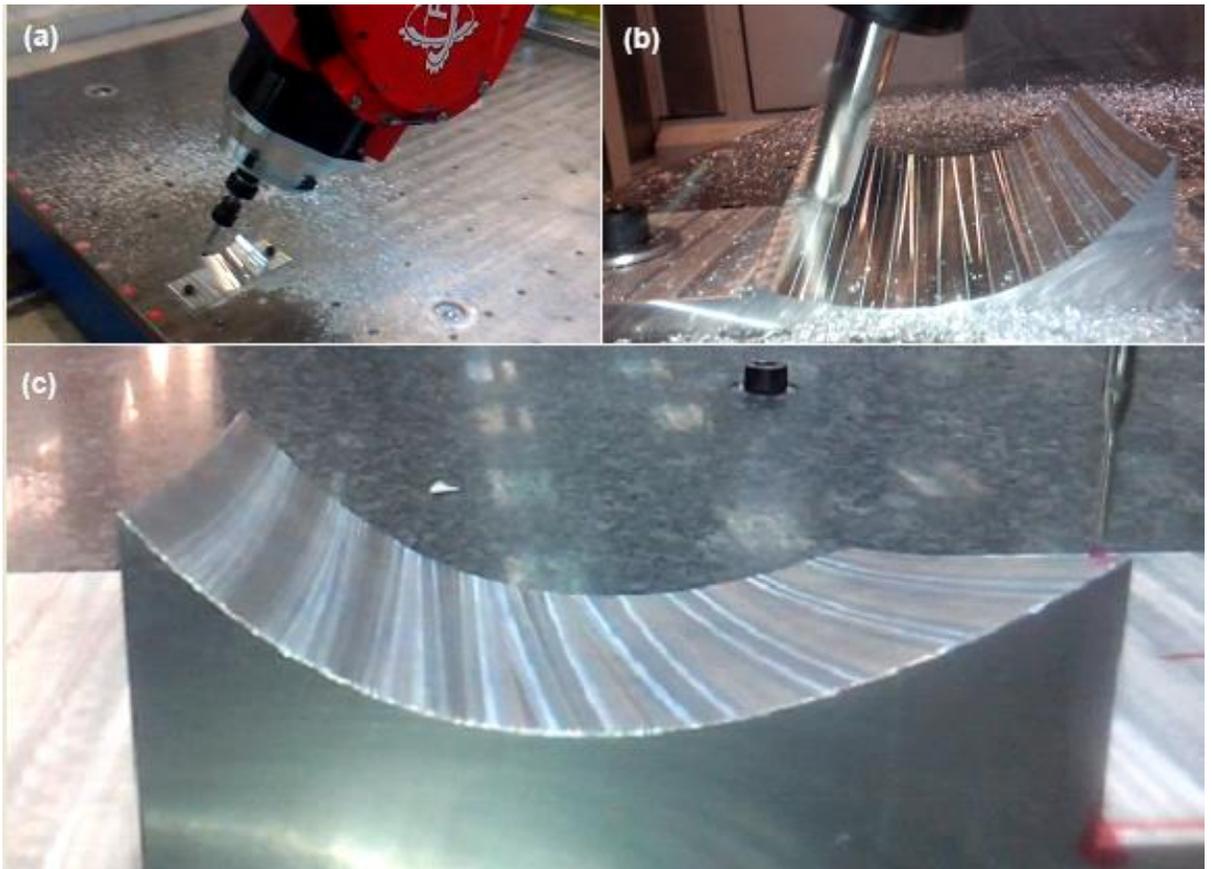
Figure 6.16: Machining results for SS-5: (a) machine spindle setup, (b) machining process, (c) final part.

By comparing these results with those reported in the above stated methods, one can notice that not only the deviation is much lower but it is well distributed to both positive and negative error directions as well. Two cases are distinguished in X=-5 mm and X=-90 mm where the error significantly fluctuates yet, still under tolerance. The fluctuations occur in these regions owing to tool's vibrations in approach and departure.

CMM simulation results are in very good agreement with CMM experimental ones, yet, slight differences exist owing to various inconsistencies. Referring to the experimental results, these inconsistencies deal with the CMM's reference axes misalignment during the job setup, missing of measurements in potential scallop regions where sharp peaks might exist and sliding of touch probe sensor in curved surface regions. Another type of error in experimental results might be given owing to the simultaneous rotation of the two additional axes of the 5-axis machine tool, A and C. This error propagates during finish-machining and may affect CMM measurements.

Even though CMM simulations were performed using the CAM output in *.stl format in CAM environment an additional effort was carried out to provide more accurate results by simulating the same CMM machine tool (DEA) used for collecting experimental CMM measurements with the same measuring step. 200 measurements were taken every 0.5mm as a measuring step, by implementing the novel cyber-physical manufacturing metrology model – CPM[3] of Majstorovic et al. (2017). CMM-simulated results obtained may also involve errors mainly due to the quality of *.stl CAM output representation and inconsistencies of wrapping technique for producing *.stl models.

By examining the results of CMM measurements depicted in Figure 6.17 referring to all four cross-sections investigated, it can be estimated that 25% to 30% of the experimental CMM measurements tend to fall close to zero reference line without significant peaks suggesting wide scallops with negligible height. Machining error is uniformly distributed across the entire sculptured surface and it was neither observable nor could be felt by touch. If the aforementioned inconsistencies of both actual and virtual CMM methods for obtaining the necessary measurements for assessing surface finish weren't experienced, simulation and experimental results could be very close to an excellent agreement.

By reviewing the results obtained for the impact case of benchmark sculptured surface SS-5 with emphasis to machining error (i.e. surface deviation, average scallop height, machining strip width – MSW) with reference to those reported in the research works related to the same optimisation problem it can be concluded that the proposed methodology not only is competitive but outperforms other methods especially in the objective of machining quality.
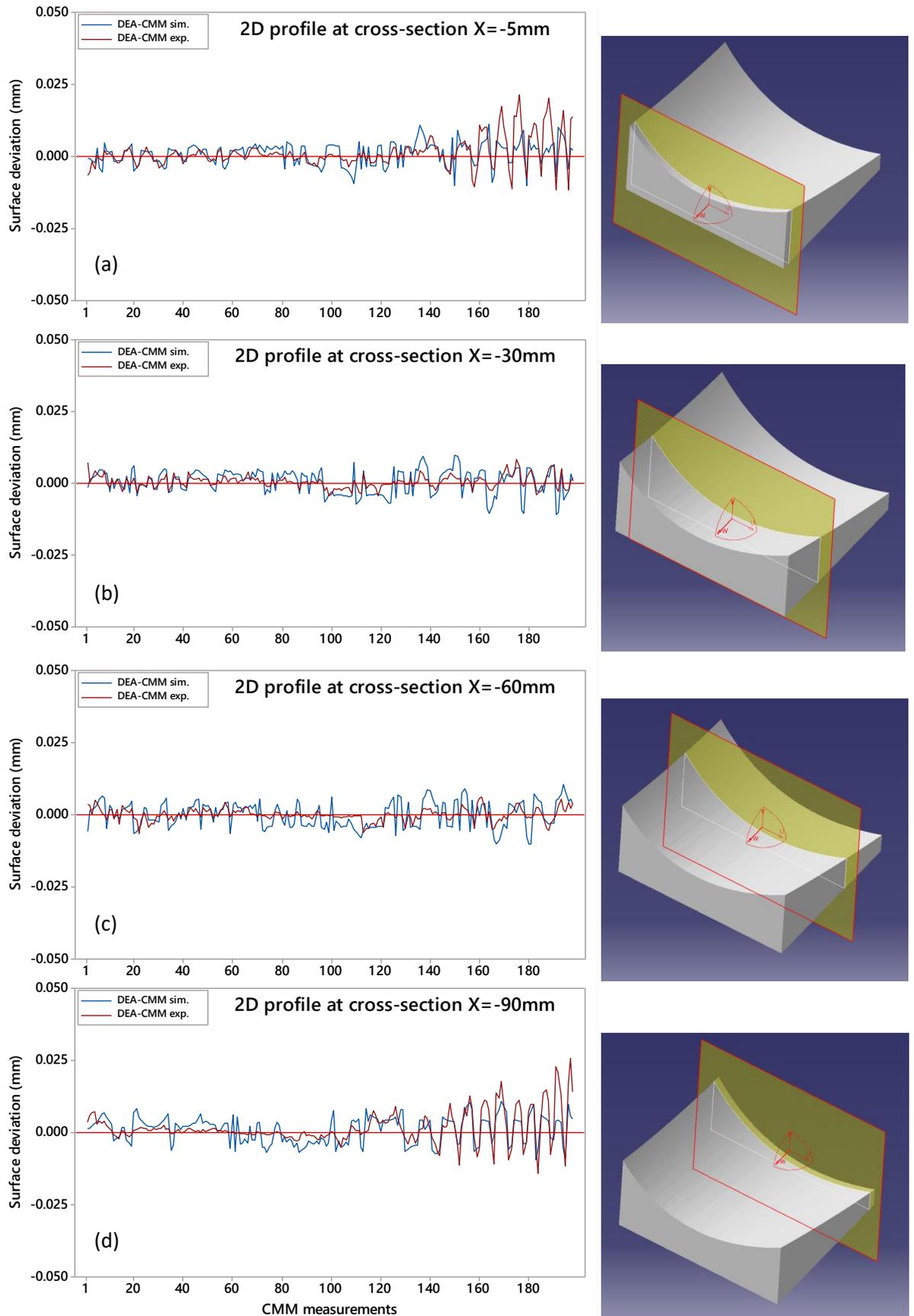
Figure 6.17: Comparison of experimental CMM and simulated CMM results for the 2D cross-sections of SS-5: (a) X=-5 mm, (b) X=-30 mm, (c) X=-60 mm, (d) X=-90 mm.

An additional quality inspection was conducted on the benchmark sculptured surface SS-5 to examine the result of maximum discretization step which determines the location of cutting points in relation to feed rate and variation of the two rotational axes, A and C. For this type of inspection, the *Taylor-Hobson® Surtronic 3+* roughness tester was used for examining the continuity among sequentially connected postures of cutting points referring to X-axis feed-forward direction. Except from the reasonable expectation of obtaining physical surface quality indicators as well, the roughness tester was used mainly under the assumption that, with a continuous measuring step to be performed by the instrument's travelling stylus, the uniformity of the interpolation error might also be observed. Figure 6.18 shows the process of testing three of the machining strips as representative to the machining error owing tool interpolation. Two machining strips selected close to the part's curved edges referring to Y-direction and a third one was selected in the middle. Proper positioning was ensured to reduce the process-related errors to the best possible extent.
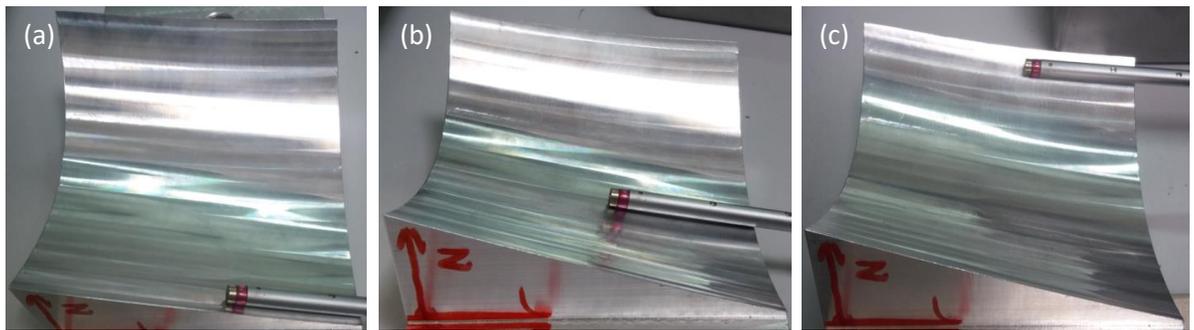
Figure 6.18: Roughness testing for the finished sculptured surface towards feed direction: (a) measurement taken to the left machining strip, (b) measurement taken to the central machining strip, (c) measurement taken to the right machining strip.

The corresponding measurement processing software *Talysurf®* was used for measuring and analyzing the machining strips. A measuring length equal to 0.8 mm was applied for the measurements. By measuring all machining strips to several regions, the means of the unfiltered roughness parameters were computed and are summarized in Table 6.19. These values reveal important information concerning the characterization of machining.

The results corresponding to this type of inspection presented a remarkably similar pattern of roughness profiles indicating that the physical machining process is not only successful from a manufacturing perspective but also maintains the smoothness of tool path postures without noticeable error fluctuations. Figure 6.19 illustrates three of the roughness profiles as representative indications for the forward step error and physical surface finish.
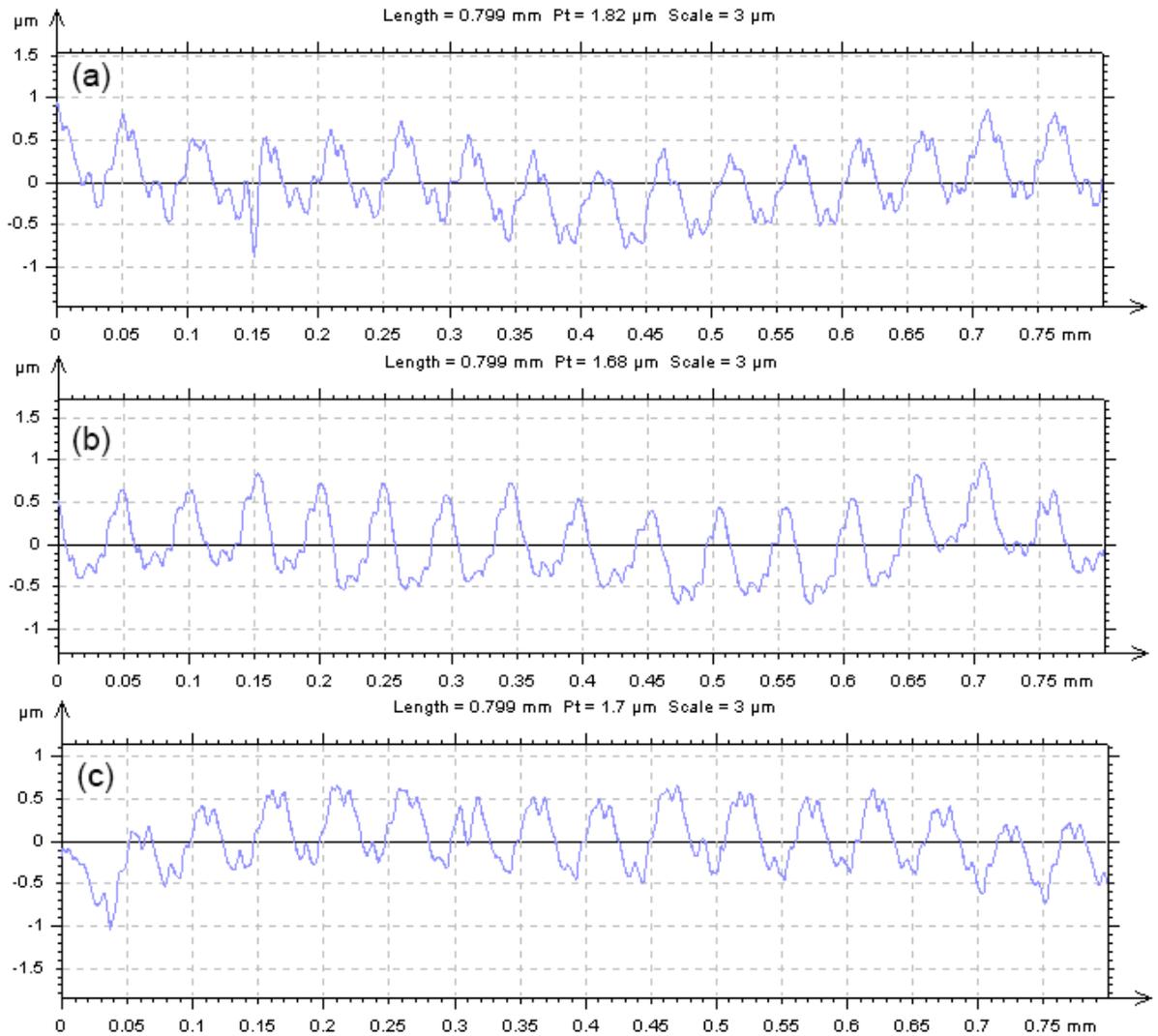
Figure 6.19: Surface quality inspection: (a) roughness profile obtained for the left machining strip, (b) roughness profile obtained for the central machining strip, (c) roughness profile obtained for the right machining strip.

Table 6.19: Mean values for unfiltered roughness parameters.

Unfiltered roughness parameter	Mean value from machining strip measurements
Pa (µm)	0.383765
Pq (µm)	0.465059
Pp (µm)	1.085294
Pv (µm)	1.087059
Pt (µm)	2.172353
Psk	0.034563
Pku	2.355882
Pz (µm)	2.172353
PTp (%) [1µm under the highest peak]	48.08353
PHTp (µm) [20%-80%]	0.858529
PSm (mm)	0.057759
PDq (o)	3.802353
PLq (mm)	0.044012

Plo (%)	0.286176
PPc (pks/mm) [+/- 0.5µm]	2.720588
PzJIS (µm)	1.604706
P3z (µm)	1.908824
Pc (µm)	1.041765
Pfd	1.322353
PHSC (Num Of Peaks) [1µm under the highest peak]	15.76471
PH (µm)	1.511176
PD (1/mm)	18.92176
PS (mm)	0.036800
Pvo (mm3/mm2)	0.000497
Pmr (%) [1µmunder the highest peak]	48.08353
Pdc (µm) [20%-80%]	0.858529

6.3.3.2 Benchmark sculptured surface SS-1

Gray et al. (2003) presented a methodology to implement for the machining of complex sculptured surfaces known as the "rolling ball" method. This method takes advantage of a computational approach capable of positioning a toroidal tool inside a hypothetical rolling sphere. The rolling ball radius is chosen as a curvature pseudo-radius which is used for positioning the cutter at a given surface contact point. Under this scheme several pseudo-radii are created according the surface properties and the tool subsequently utilizes them for being properly positioned under a preset tolerance to avoid gouging with the surface. The part was machined using a 5-axis CNC machining center with a toroidal cutter with major radius R=12.7 mm and minor radius r=6 mm (D37.4Rc6). Their algorithm implemented a range for discretization step from 0.007mm to 1.397mm. Scallop profiles were examined via CMM measurements and their average scallop height was found equal to 0.025 mm. 23 sequential machining strips were observed on the cut surface yet, the average width was not measured.

The methodology proposed in this PhD thesis was implemented to optimise the 5-axis machining tool path for the same benchmark surface (SS-1) using the parameters recommended as optimal. Table 6.20 summarizes the upper and lower inputs as well as the optimal values found.

Table 6.20: Tool path parameter bounds and optimal recommended values for the case of benchmark surface SS-1.

Benchmark surface	Levels	Tool	Stepover (%D)	Lead angle (deg)	Tilt angle (deg)	MaxDstep (mm)
	Low	D37.4-Rc0	10	20	0	0.007
SS-1	High	D37.4-Rc6	45	35	7	1.397
	Optimal	D37.4-Rc6	18.9% (7.069 mm)	20.231	0.114	1.090

The recommended parameters were implemented for the machining simulation and the actual cutting experiment. As the optimal tool D16Rc3 was used against D16Rc0. By simulating a feed equal to 1000 mm/min the simulation result was found equal to 1min-51sec for machining time and 2min-04sec. for total time. The simulated machining time was found in agreement with actual machining time given by the CNC unit. 22 smooth and uniformly distributed cutting strips were left on the actual cutting surface. The average machining strip width was equal to 20.082 mm and their average overlap was 13.733 mm. The optimal simulated and actual cut surfaces were examined at three cross-sections with respect to the work of Gray et al. (2003). The cross sections were taken on Y=39 mm, Y=76.5 mm and X= 151.5 mm. In the simulation the test-points were arranged in the same way as the measurement points taken by the CMM for the experimental results with 1.683 mm measuring step. Figure 6.20 depicts the machining result, Figure 6.21 the normalized deviation of the machined surface examined in the three aforementioned cross-sections and Figure 6.22 the results for the same surface obtained by Gray et al. (2003) for easy reference. By examining each of the three cross-sections it was observed that not only the Z-height difference between actual and nominal surface was lower than that reported for the "rolling ball" method but it was also uniformly distributed across the measuring path. Maximum deviation error does not exceed 0.07 mm whereas minimum deviation approximates -0.02 mm. Scallop curves were almost unnoticed in the actual cut surface and their average height did not exceed 0.02 mm.
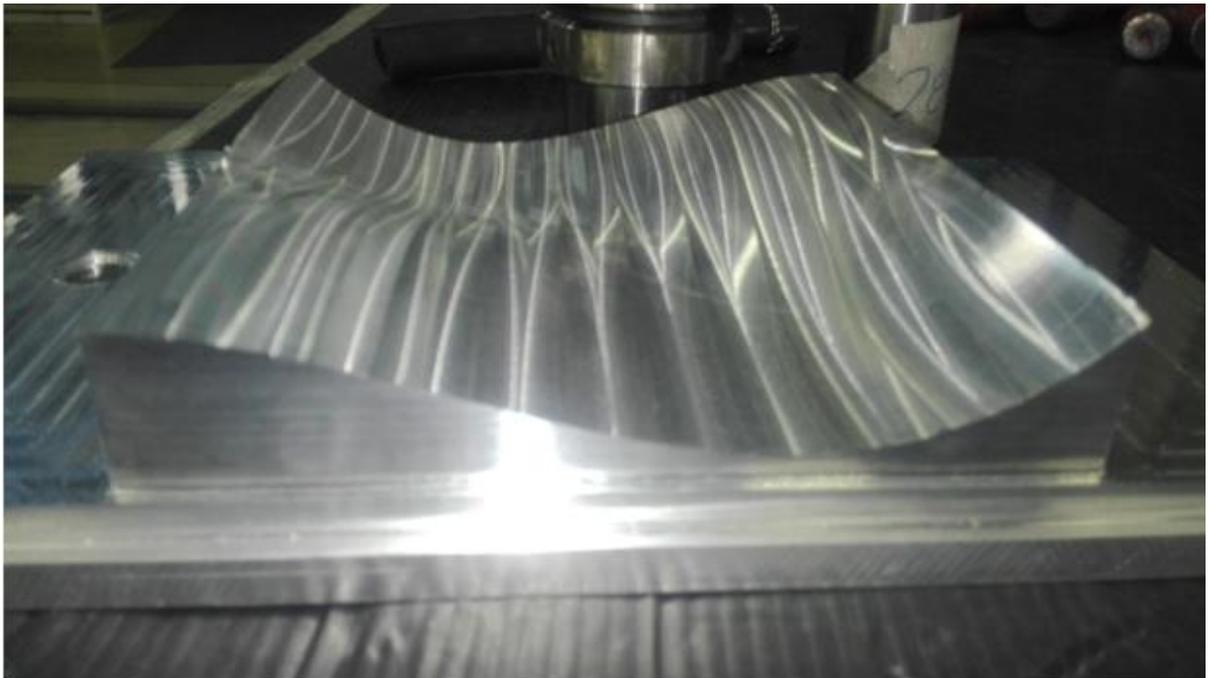
Figure 6.20: Machining result for the benchmark sculptured surface SS-1.
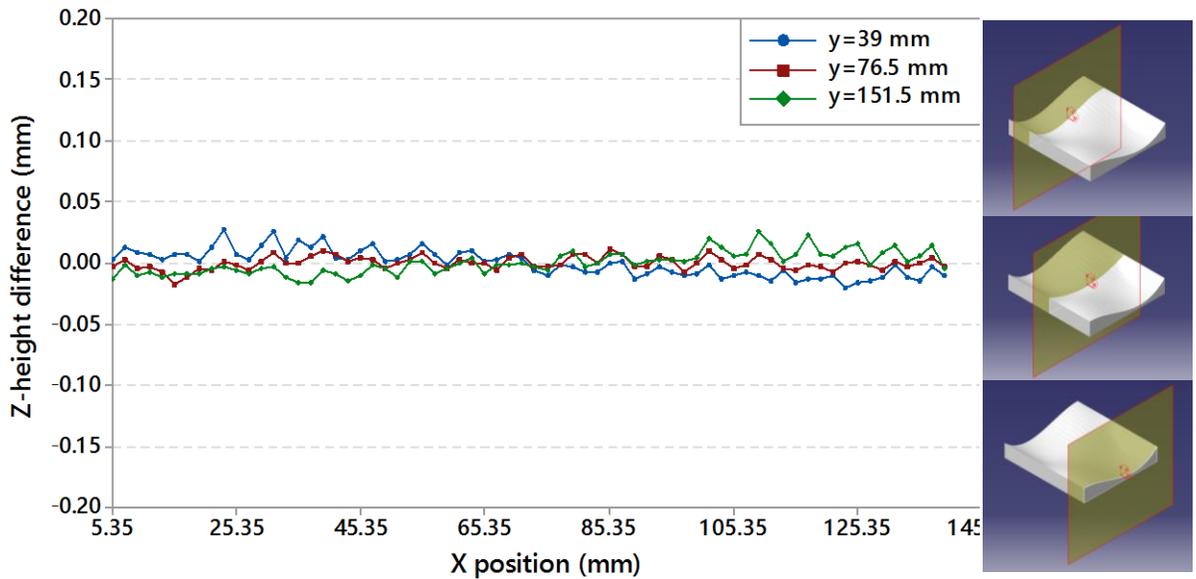
Figure 6.21: Plot of the Z-height difference between actual and nominal measurements for the various cross-sections of benchmark sculptured surface SS-1.
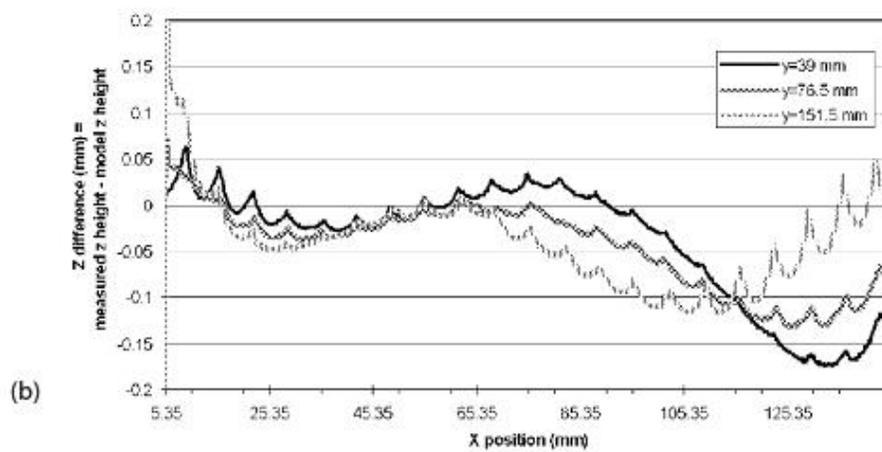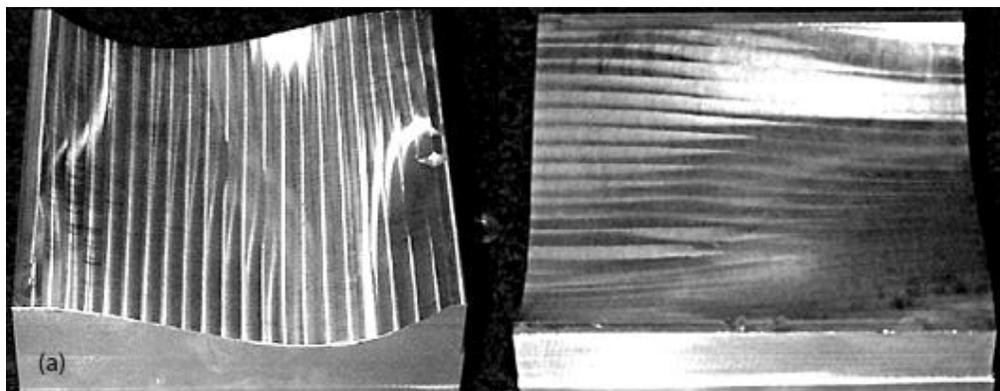
Figure 6.22: Research results from Gray et al. (2003): (a) actual surface machined using the "Rolling ball" method, (b) plot of the Z-height difference between actual and nominal measurements for the various cross-sections of benchmark sculptured surface SS-1.

6.3.3.3 Benchmark sculptured surface SS-2

The method proposed by Gray et al. (2003) was integrated by graphics-assisted utilities to contribute further to the tool path planning problem for sculptured surface CNC machining. In the work of Gray et al. (2004) tool paths for sculptured surfaces are generated using triangulated data rather than employing parametric surface equations. In addition, the method can create tool paths for sculptured surfaces where only positional continuity exists. To verify their approach, they implemented it on the benchmark surface SS-2 which is a surface with two bi-cubic contours connected with a C_0 continuous curve. This was suggested as an extreme case in the machining of multiple patches having only C_0 position continuity (Gray et al. 2004). Results reported in the work of Gray et al. (2004) were limited to the forward step value computed in the vicinity of the C_0 curve and the rest of the surface which was found equal to 0.762 mm and 2.00 mm respectively. 22 machining strips were left on the surface whilst the maximum scallop height was found equal to 0.1 mm. The maximum undercut was 0.07 mm. Note that feed direction was intentionally determined to be vertical to C_0 curve during surface machining to introduce special challenge in terms of quality and productivity. The methodology proposed in this PhD thesis was implemented to optimise the 5-axis machining tool path for the same benchmark surface using the parameters recommended as optimal. Table 6.21 summarizes the upper and lower inputs as well as the optimal values found.

Table 6.21: Tool path parameter bounds and optimal recommended values for the case of benchmark surface SS-2.

Benchmark surface	Levels	Tool	Stepover (%D)	Lead angle (deg)	Tilt angle (deg)	MaxDstep (mm)
	Low	D50.8- Rc6.35	10	15	0	0.762
SS-2	High	D50.8-Rc0	45	20	15	2.000
	Optimal	D50.8-Rc6.35	14.232% (7.230 mm)	15.7	5.373	1.653

The recommended parameters were implemented for the machining simulation and the actual cutting experiment. As the optimal tool Ø50.8 Rc6.35 was used against Ø50.8 Rc0. By simulating a feed equal to 1000 mm/min the simulation result was found equal to 3'43'' for machining time and 4'33'' for total time. The simulated machining time was found in agreement with actual machining time given by the CNC unit. The rotational speed was set to the relatively low value of 4000 rpm to avoid vibrations during cutting owing to the length of the tool assembly. 22 smooth and uniformly distributed cutting strips were left on the actual cutting surface. The average machining strip width was equal to 27.088 mm and their average overlap was 21.121 mm. Figure 6.23a depicts the 5-axis machining center's spindle setup during machining, Figure 6.23b depicts the machining operation

close to the C_0 continuous curve and Figure 6.23c shows the final part. Figure 6.24 shows the resulting part by implementing the "graphics-assisted rolling ball method" of Gray et al. (2004) for comparison purposes. The finished part was inspected by taking several CMM measurements with 2.5 mm measuring step in five 2D cross sections determined on X=25.4 mm, X=50.8 mm, X=76.2 mm, X=101.6 mm and X=127 mm (Figure 6.25) vertical to feed direction with reference to the machining axis system (*G54*). The average deviation was found equal to 0.0148 mm, 0.0116 mm, 0.0220 mm, 0.0131 mm and 0.0185 mm for the cross sections respectively, giving a total average deviation equal to 0.0160 mm. The maximum scallop height was equal to 0.071 mm whereas the maximum undercut measured was 0.058 mm.
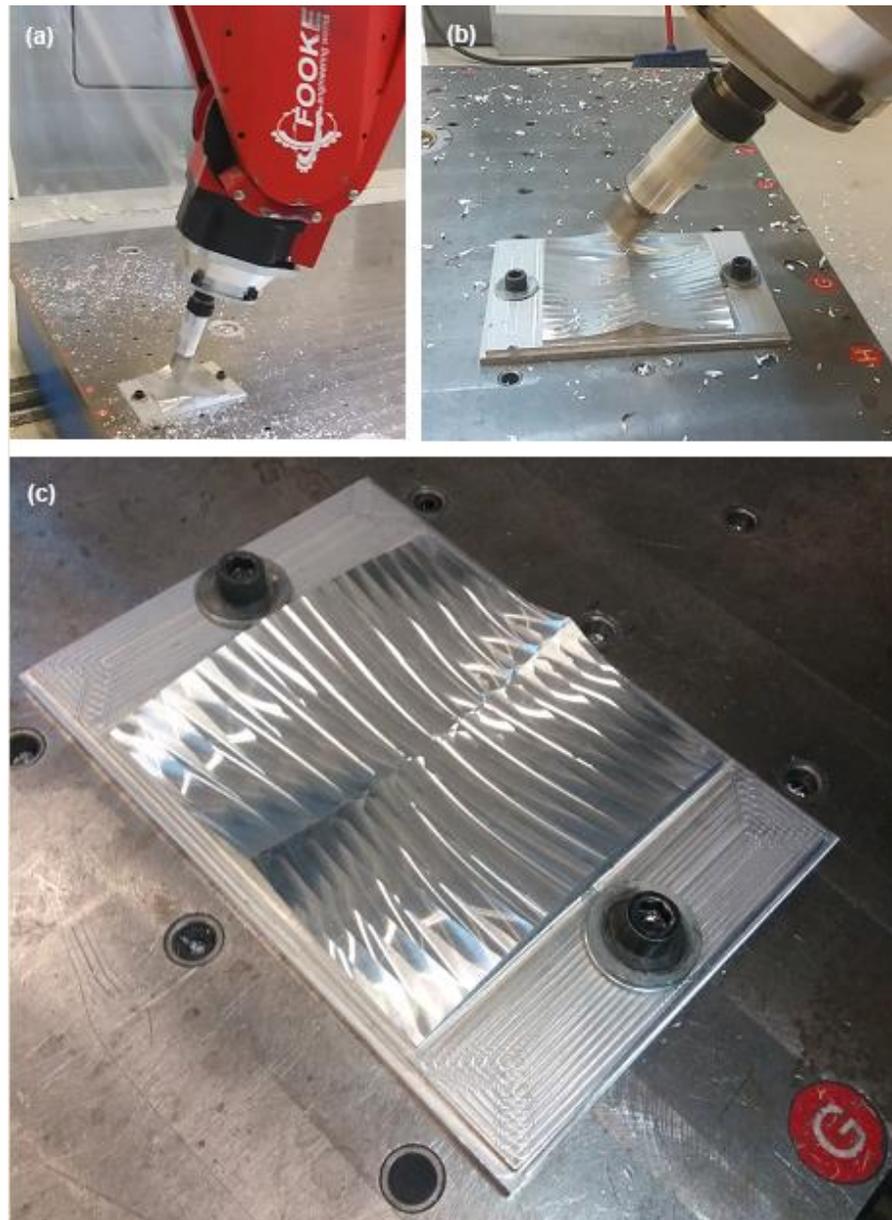
Figure 6.23: Machining results for SS-2: (a) machine spindle setup, (b) machining process, (c) final part.
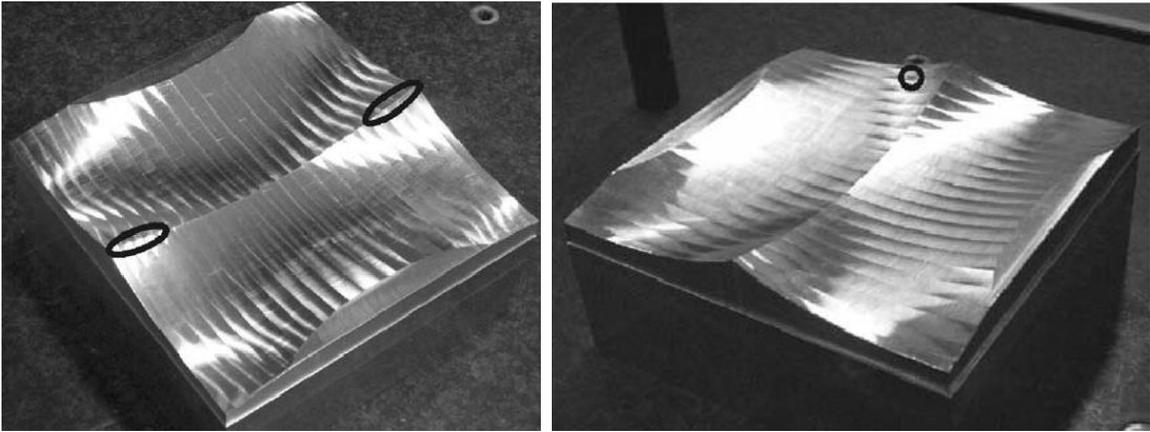
Figure 6.24: Machining result of the benchmark sculptured surface SS-2 (Gray et al. 2004).
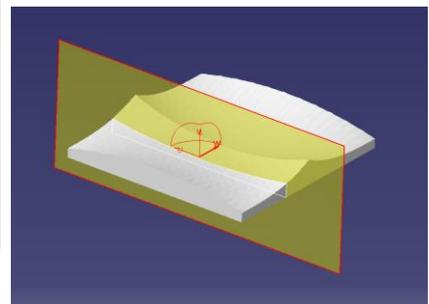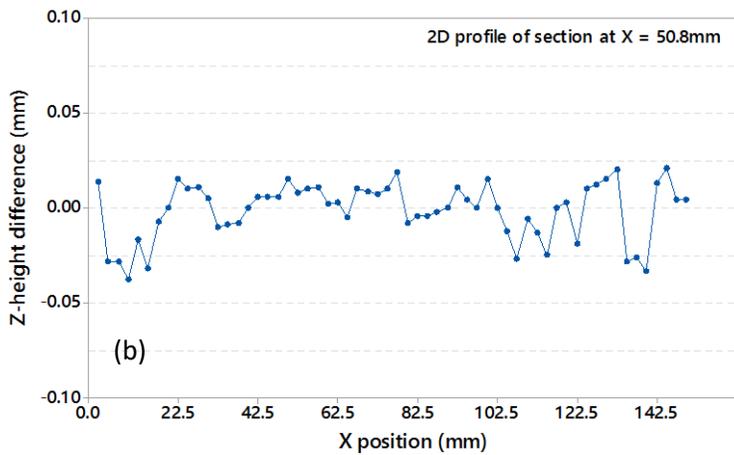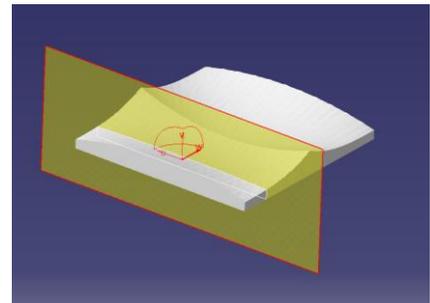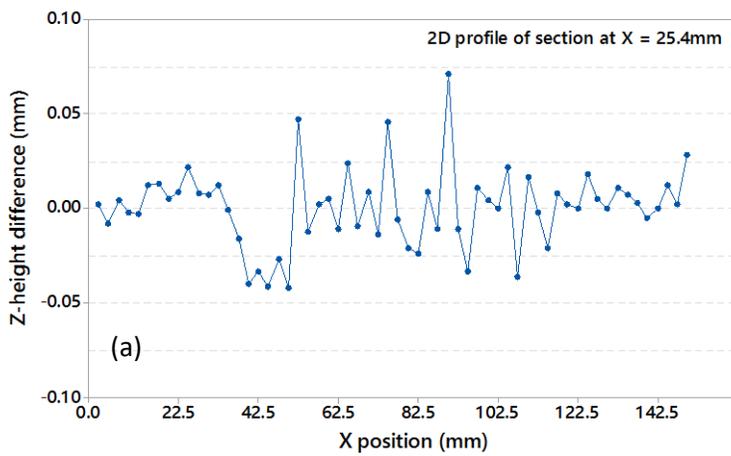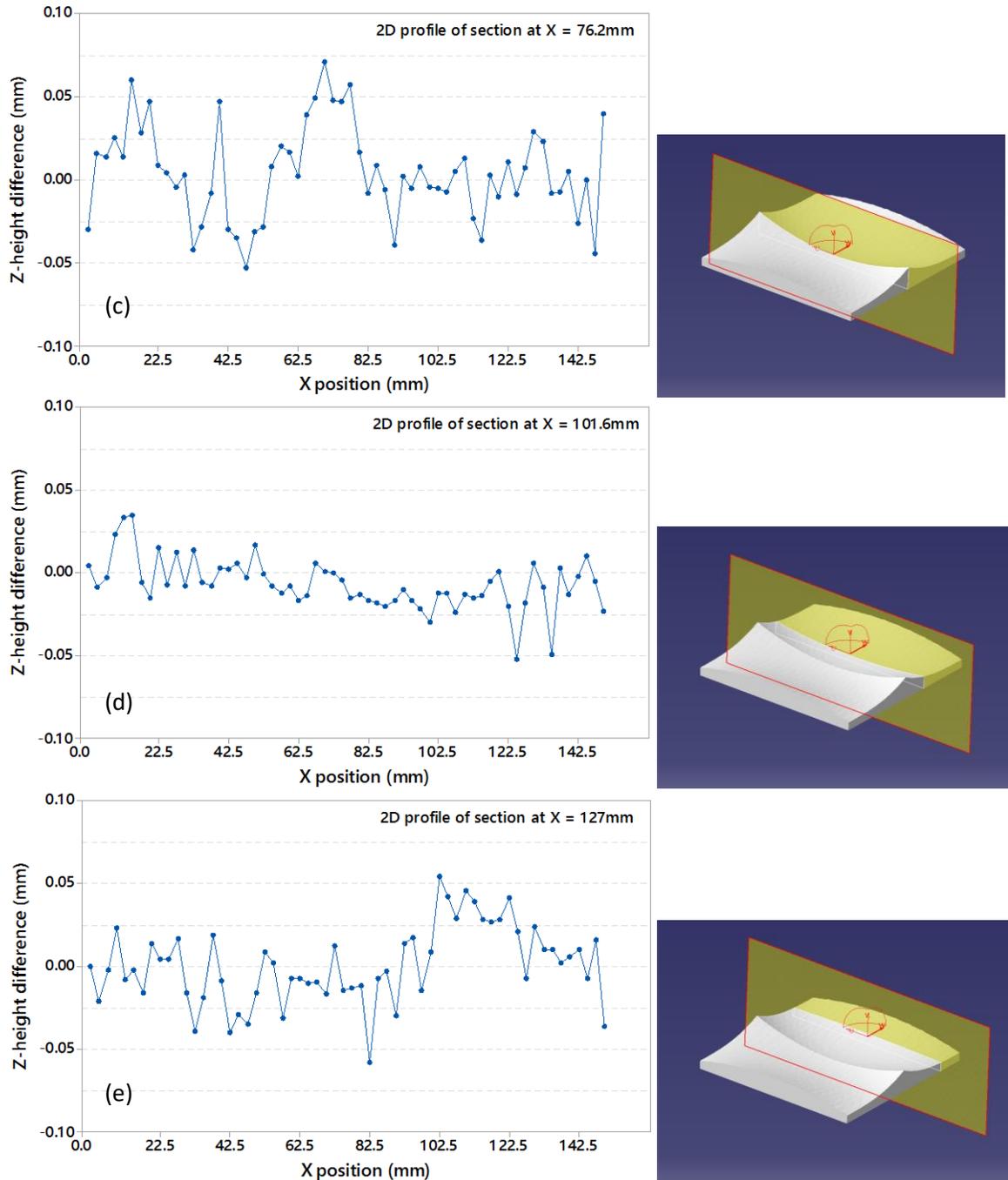
Figure 6.25: Experimental results (CMM measurements) of 2D cross section profiles for SS-2: (a) X=25.4 mm, (b) X=50.8 mm, (c) X=76.2 mm, (d) X=101.6 mm, (e) X=127 mm.

Further validation tests were examined on the same benchmark sculptured surface SS-2 to examine the fluctuation (uniformity) of the deviation error on the two scallop curves where the largest error was observed (Figures 6.26a and 6.26b). These two scallop curves were on the contours of the surface where the cutting tool approached to and departed from. The 2D profiles determined on the cross sections at Y= 4mm and Y= 149.5mm were examined through simulation measurements taken

with 1 mm measuring step using the CAM output since no probe accuracy could be achieved on the scallops by CMM. For these two profiles the height of measuring points in Z-axis was found in good agreement when compared to the exact points taken on the same cross sections of the ideal CAD model. It was observed that the error fluctuates smoothly and uniformly at the bi-concave regions of the surface whilst approaching the vicinity of C_0 continuous curve this error is reduced. A remarkable agreement of the simulated error was also observed on the C_0 continuous scallop curve where another 2D profile taken on its corresponding cross section was examined (Figure 6.26c). This result implies that C_0 continuous scallop curve was not significantly affected (in terms of its geometry) by the changes of tool axis orientation which means smooth transition among tool position vectors. By comparing the results obtained using the proposed optimization methodology to the related ones available by Gray et al. (2004) it is deduced that further improvement has been achieved for the tool path to machine SS-2. Both the machining error deviation and its distribution leads to the conclusions of achieving more beneficial tool positions regarding the discretization step as well as lead and tilt angle values for the same cutting tool suggested.
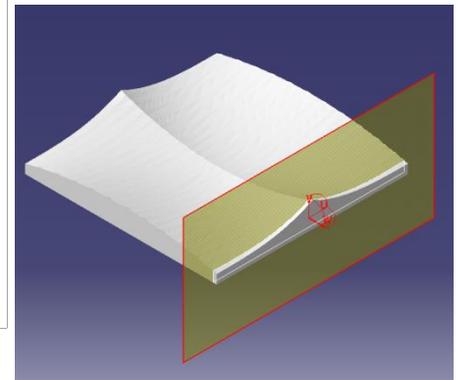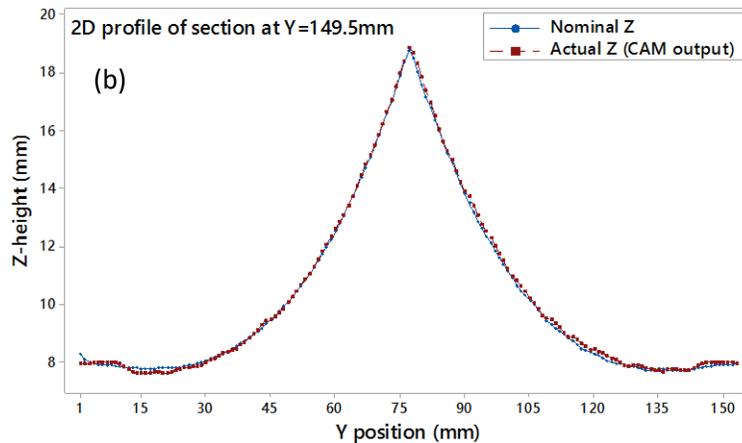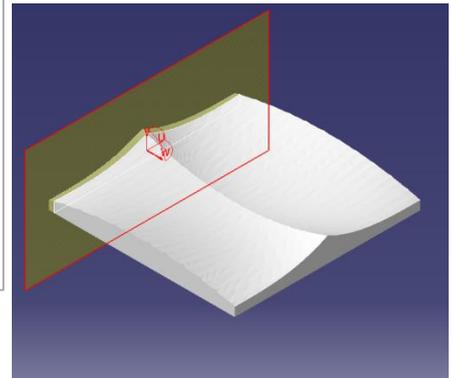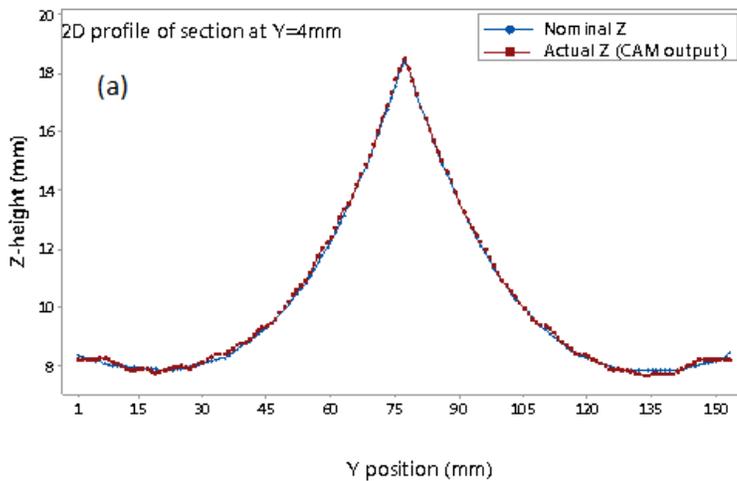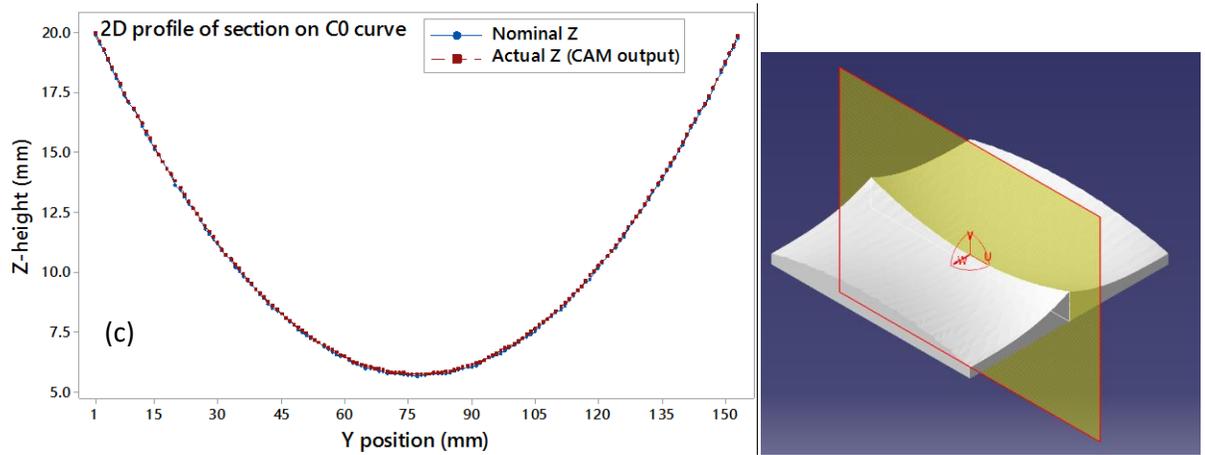
Figure 6.26: Experimental results (CMM measurements) of 2D cross section profiles for SS-2: (a) Y= 4 mm, (b) Y=149.5 mm, (c) C_0 continuous curve.

6.4   Summary and conclusions

The proposed methodology for optimising the sculptured surface CNC machining problem was applied to various benchmark sculptured surfaces to validate the results obtained by its application and perform rigorous comparisons with reference to results from other tool path planning / optimisation methods available in the literature. The intelligent part of the methodology which is the multi-objective virus-evolutionary genetic algorithm – MOVEGA is responsible for maintaining quality of optimal results. Thus, to compare its capabilities against those found to other modern stochastic algorithms a common problem-solving environment (design space) was formulated using regression models from the series of machining simulation experiments introduced in Chapter 3 to study the effect of tool path planning parameters.

As far as the algorithmic validation part is concerned the average gain by selecting the MOVEGA as the intelligent algorithm to integrate the proposed optimisation methodology is close to 17.80% when testing the various regression models as objective functions to optimise the tool paths for the benchmark sculptured surfaces examined. This percentage implies significant differences among the individual objectives i.e. mean machining error, machining time and remaining volume after finish-machining. Although such an approach cannot fully represent the sculptured surface CNC machining problem owing to lack of generality found in regression modelling, it was accepted only under the perspective of comparing results obtained by the same problem design space since linking all new algorithms examined in the proposed optimisation methodology goes far beyond the research bounds set in this work.

Machining-simulated results as well as actual experimental outputs were investigated to characterize the efficiency and quality of the properties the proposed optimisation methodology exhibits. It was found that the methodology manages to distinguish optimal tool path parameter values among other candidate solutions for the tool path applied to machine the benchmark sculptured surfaces. It was also shown that, despite its stochastic nature and the absence of mathematical definitions for representing the benchmark surfaces, the methodology can indirectly adhere to crucial elements characterizing the mechanics of multi-axis material removal operations such as the multi-point contact between the cutting tool and the surface leading thus to an efficient machining with wide tool path strips while maintaining surface quality and precision.

As far as the validation of results when studying actual CNC machining outputs is concerned the proposed methodology exhibits a gain equal to 12.48% by considering the best tool path planning / optimisation method, the multi-point machining (MPM). This percentage is referred to the average scallop height as a key objective of the MPM method whilst other optimisation criteria such as machining strip width where also found to be competitive by using the proposed methodology. From the resulting machined surfaces, the methodology seems to surpass the "Rolling ball" in a significant level whilst it produces 29% lower average scallop height that that of the graphics-assisted rolling ball method for the same number of cutting paths. By comparing the maximum undercuts of the proposed methodology and graphics-assisted rolling ball method the former produces 17.15% less gouging that the latter. Similar conclusions are drawn when examining the rest of the results corresponding to the tool path planning / optimisation methods especially when dealing with the same resources / materials with emphasis to the cutting tool's geometry and configuration.

Chapter 7

Conclusions and future recommendations

7.1 Conclusions and research assessment

Sculptured surface CNC machining is an important industrial manufacturing process to produce a variety of aesthetic, modern, and versatile complex products. Computer-aided manufacturing environment provides the one and only infrastructure to generate multi-axis surface machining tool paths by determining specific values for the parameters involved, cutting tool, stepover, lead angle, tilt angle and maximum discretization step. Advantageous sculptured surface CNC machining using optimised tool paths will ultimately result to better surface finish while maintaining competitive production times. The research aim of this work was to develop a generic methodology for the intelligent optimisation of 5-axis sculptured surface CNC machining (end-milling) tool paths. With reference to the various experimental results presented it has been shown that methodology developed in the thesis has fulfilled this research aim.

One of the most crucial objectives of this work was to determine the criteria for formulating the generalised sculptured surface CNC machining problem having in mind the independent parameters one needs to set so as to generate a swept surface multi-axis tool path. The machining parameters investigated in this research work are cutting tool, stepover, lead angle, tilt angle and maximum discretization step. Under this premise the independent parameters were studied by testing the swept surface multi-axis tool path to several sculptured surfaces with different properties to generalise the results. Machining simulations were conducted to examine the effect of important multi-axis tool path parameters on the generic criteria established for representing the generalised sculptured surface CNC machining problem. The criteria were used not only for presenting the generalised problem but also for providing a relation among tool path parameters and CAM outputs as virtually "physical" products. The elements used for determining the problem's generic criteria were examined for their validity through experiments and statistical significance tests. Apart from this important activity the parameters were also investigated for deciding the number of accuracy digits when it comes to the binary representation of tool path "chromosomes" and the data structure

needed for allowing proper interaction among the several programming modules the developed methodology comprises. It comes as a conclusion that such a methodology could only be established once the aforementioned attributes were properly examined.

After the problem formulation and investigation of the effects of swept surface multi-axis tool path parameters the next objective was to establish the generic methodology for optimising sculptured surface CNC machining tool paths. A two-fold programming framework was developed involving the part that fully automates CAM environment and its corresponding functions and the part of intelligent optimisation module embedding the multi-objective virus-evolutionary genetic algorithm (MOVEGA). The two parts interact and exchange data for performing stochastic evaluations to solve the generalised sculptured surface CNC machining problem and provide globally optimal surface machining tool paths for any sculptured surface regardless of its mathematical definition.

The optimisation methodology's functional behaviour would remain vague as well as its full potentials should an investigation for the optimal algorithm-specific parameter settings wouldn't have been conducted. Therefore, an important objective of this work was to study the effect of the parameters referring to the viral operators of the new algorithm, on the overall algorithmic performance and quality of final "optimal" result. In order to study the effect of algorithm-specific parameters the methodology was applied to a benchmark sculptured surface and the results were statistically exploited to observe the advantageous regions of parameter values so as to decide their final settings. These settings were employed to perform confirmation experiments and compare the optimisation methodology's embedded algorithm (MOVEGA) to itself when omitting the viral operators to prove that the former is prominent.

The last objective of this work was to validate the results of the optimisation methodology against those available from other competing tool path planning and/or optimisation methods for sculptured surface CNC machining. In addition, the methodology developed was compared to various intelligent algorithms using regression models correlating the independent tool path parameters to the generic optimisation criteria introduced to formulate the sculptured surface CNC machining problem. From the perspective of algorithmic evaluations, the MOVEGA was found quite promising in terms of the prerequisites needed for achieving better results to simultaneously optimise machining efficiency and surface finish. However, these algorithms should be rebuilt from scratch to become compatible with the already developed environment for their reliable implementation to solve the problem. From the perspective of process-related assessment the methodology not only outperforms other competitive tool path planning / optimisation methods but also exhibits important indications accompanying physical sculptured surface CNC machining processes, even unintentionally, with emphasis to the

simultaneous 5-axis machining and the multi-point tool-surface contact for increasing efficiency while maintaining surface finish. Even though one may not expect that a software-based system dedicated to tool path optimisation would adhere to physical process elements or mechanics of processes, the methodology presented in this work has managed to enhance the trajectory of multi-axis swept surface tool paths as well. This outcome suggests a major departure from any other method proposed for multi-axis tool path planning / optimisation while it significantly contributes to the fields of intelligent manufacturing, sculptured surface machining and engineering software development.

The need to develop the proposed optimisation methodology for sculptured surface CNC machining arises from the particularity of tool path parameters which cannot be correlated so that a generic solution can be found. The requirement for tool paths capable of simultaneously minimising surface machining error, maintaining its uniformity and minimizing the number of cutting data for the CNC program, calls for a stochastic methodology to deal with the direct exploitation of the aforementioned tool path parameters as a single candidate solution. There is also a need to automate time-consuming, repetitive tasks when it comes to tool path planning as well as trial-and-error machining simulation scenarios. Consequently, the methodology developed for addressing the generalised sculptured surface CNC machining problem contributes to the broader research field of intelligent manufacturing as follows:

1   The methodology constitutes a practically viable tool and user-friendly environment to optimise complex sculptured surface tool paths by using standard and known resources to practitioners, such as CAD/CAM systems,

2   The methodology pushes further the envelope of profitability and efficiency of intelligent manufacturing by supporting automation and optimisation,

3   The methodology handles simultaneously the parameters involved to tool path planning / optimisation for complex machining while it achieves optimisation under a global essence,

4   The methodology shares and develops new ideas for the next generation's manufacturing software development, dealing with artificial intelligence and its effective implementation,

5   It accounts for absolutely zero trial-and-error machining simulation scenarios and iterative experimental efforts for finding "optimal" values for tool path parameters.

It is reasonable to consider that any new technology aiming at facilitating industrial operations comes also with its shortcomings. The methodology developed for optimising sculptured surface tool paths for the multi-axis CNC machining has the major drawback of needing a considerable amount of time to execute the evaluations in order to end up to the optimal result since each algorithmic evaluation

corresponds to a machining simulation. This running time depends on the nominal dimensions of the surface under investigation, the settings of algorithm-specific parameters, i.e. the number of evaluations required for reaching an optimal output, the number of candidate solutions (tool path chromosome) about to be evaluated and the configuration of the computer system on which the system will be operated. The experiments required for this research were performed on a Windows 8.1 Pro., Intel® Core ™ i3-4160 CPU, 3.60 GHz 64-bit operating desktop system with 8.00 GB RAM. The average time needed to simulate the benchmark sculptured parts was about 3 to 4 hours including the system setup referring to the initial tool path planning according to the cutting strategy selection. However this running time may antagonise the actual time practically needed even by an experienced process planner in the case of complex sculptured surfaces. In addition one can imagine a reduction to an important fraction of this running time when a high-performance hardware system may be implemented to support the developed methodology.

It is very likely that the proposed methodology might not lead to optimal results for tool path parameters with regard to optimization criteria established in this work. This may occur in exceptional cases of extremely complex sculptured surfaces where the dramatic changes in curvature may not allow for a reliable tool path trajectory generation for the cutting tool to follow. It is mentioned here that the CAM system would be responsible for such a case and not the proposed optimisation methodology since the latter depends on the capabilities of CAM software. This can be addressed by integrating the optimisation methodology with a routine to search for the optimal feed direction whilst it is expected to be the one with the lowest curvature. Nevertheless, the current status of the methodology can guarantee that, at worst, the resulting near-optimal solution would be again more advantageous compared to a tool path planning scenario prepared even by a highly experienced NC programmer since it is impossible to find near-to-optimal or exact values for planning a tool path capable of simultaneously optimising all criteria involved based only in experience.

The optimization methodology proposed can only guarantee optimal tool paths under the perspective of implementing multi-axis sweeping tool paths accompanied to their corresponding recto-linear cutting paths (zig-zag cutting style). In addition, feed direction with reference to the recto-linear angle has not been under investigation by taking in advance that the optimal one would be found towards the surface region with the lowest curvature. Nevertheless, to ensure quality of results or even optimize further the second optimization criterion introduced (tool path smoothness or machining error distribution) feed direction ought to normally constitute an optimization parameter. Finally the CAM solution plays important role to the optimal results the proposed

methodology may obtain, since different software packages for CAM offer different utilities, sharing several strengths and weaknesses themselves.

7.2   Recommended future work

The generic methodology for globally optimising the sculptured surface CNC machining problem has been tested with reference to tool paths following a multi-axis swept surface cutting strategy. Despite that this cutting strategy covers almost the 90% of finish-machining operations for moulds / dies and other complex products found in industry one could automate the functions of other already existing or newly developed tool paths. In this case, the changes or amendments needed to integrate the methodology developed may be straightforwardly done once the programming instances are known and incorporated in the form of additional code using the methodology's automation function. Since this function has been externally developed its modules can accommodate the routines of other CAD/CAM packages other than the one employed in this work. This can be accomplished provided that the software development architecture (known as the application programming interface – API) of a CAD/CAM system allows for further customization via programming or the development of new code to extend its capabilities.

The work conducted leaves also room for the research concerning the optimal coordinates of NURBS control points if tool paths are to be planned by adopting a NURBS interpolation. Instead of inherently optimising the cutting tool positions for reducing machining error it is possible to fit NURBS tool paths with fewer control points and with optimal locations in the parametric space. In addition, one can envision a novel post-processing engine for turning the optimal CNC program of this work to a NURBS format according to the recommendations of noticeable contributions found in literature. Speaking of post-processor development, it is easy to take advantage of the current functions for computing sequential tool positions towards the direction of feed rate and apply an adaptive feed rate interpolator to optimise also cutting conditions including rotational speed. In addition to feed adaptation one can easily employ the functions of latest NC units found in industry as well as 5-axis machine tool configurations and support any type of CNC format and 5-axis machining kinematics with reference to the recommendations found in Fountas et al. 2017b. The formulation of optimised, complete manufacturing programs with roughing, finishing and some intermediate machining operations can be also a prosperous future research with this work as a reference. It is possible to apply the existing environment to optimise the roughing process for a sculptured surface once the

criteria have properly been modified. Obviously, one should decide to deal with cutting force components - with emphasis to the main cutting force – with cutting force variations and the material volume left for the forthcoming processes, semi-finishing and finishing. Such an effort would not be started by scratch; research has already been conducted to initialize such an idea (Fountas et al. 2015). With the progress of hardware and software new utilities are to be introduced in the next few years related to machining kinematics, servos and NC controllers as well as to 4th generation CAM systems and novel algorithms for intelligent machining/process planning to facilitate industry 4 and its corresponding elements. One should follow these trends and try to apply new knowledge to the already existing environment towards the establishment of a complete infrastructure for optimised design (CAD), optimised analysis with either the boundary element method or the finite element method (BEM-FEM) and finally optimised computer-aided manufacturing (CAM) with Step-NC commands for on-line CNC monitoring. The possibility of introducing optimal setups with automated fixturing / part positioning could be also a future work based on the current one.

References

Agrawal R.K., Pratihar D.K., Choudhury A.R. (2006). Optimization of CNC isoscallop free form surface machining using a genetic algorithm. International Journal of Machine Tools and Manufacture, 46(7-8), 811-819.

Altintas Y., Kersting P., Biermann D., Budak E., Denkena B., Lazoglu I. (2014). Virtual process systems for part machining operations, CIRP Annals - Manufacturing Technology, 63(2), 585-605.

Anderson N. (1970). Evolutionary Significance of Virus Infection. Nature, 227, 1346-1347.

Annoni M., Bardine A., Campanelli S., Foglia P., Prete C.A. (2012). A real-time configurable NURBS interpolator with bounded acceleration, jerk and chord error. Computer-Aided Design, 44(6), 509–521.

Baker J.E. (1987). Reducing bias and inefficiency in the selection algorithm. Proceedings of the 2nd International Conference on Genetic Algorithms and their application, pp. 14-21, ISBN:0-8058-0158-8, Cambridge, Massachusetts, USA.

Bedi S., Ismail F., Mahjoob M.J., Chen, Y. (1997). Toroidal versus ball nose and flat bottom end mills. International Journal of Advanced Manufacturing Technology, 13(5), 326-332.

Behera S.K., Meena H., Chakraborty S., Meikap B.C. (2018). Application of response surface methodology (RSM) for optimization of leaching parameters for ash reduction from low-grade coal. International Journal of Mining Science and Technology, 28(4), 621-629.

Benardos P.G., Vosniakos G.C. (2002). Prediction of surface roughness in CNC face milling using neural networks and Taguchi's design of experiments. Robotics and Computer Integrated Manufacturing 18(5-6),343-354.

Beudaert X., Lavernhe S., Tournier C. (2014). Direct Trajectory Interpolation on the Surface using an Open CNC. International Journal of Advanced Manufacturing Technology, 75 (1-4), 535-546.

Bhavsar S.N., Aravindan S., Rao P.V. (2015). Investigating material removal rate and surface roughness using multi-objective optimization for focused ion beam (FIB) micro-milling of cemented carbide. Precision Engineering, 40, 131–138.

Box G., Draper N.R. (1987). Empirical Model-Building and Response Surfaces. ISBN-13: 978-0471810339, Wiley, New York.

Box G.E.P., Hunter J.S., (1957). Multi-factor experimental design for exploring response surfaces. *Annals of Mathematical Statistics,* 28(1), 195–241.

Branke J., Kaußler T., Schmeck H. (2001). Guidance in evolutionary multi-objective optimization. Advances in Engineering Software, 32(6), 499–507.

Budak E., Lazoglu I., Guzel B.U. (2004). Improving Cycle Time in Sculptured Surface Machining Though Force Modeling. CIRP Annals 53(1), 103–106.

Castelino K., D'Souza R., Wright P.K. (2003). Toolpath Optimization for Minimizing Airtime during Machining. Journal of Manufacturing Systems 22(3), 173–180.

Chen J.-S., Huang Y.-K., Chen M.-S. (2005). A study of the surface scallop generating mechanism in the ball-end milling process. International Journal of Machine Tools and Manufacture, 45(9), 1077- 1084.

Chen Z.C., Khan, M.A. (2014). A new approach to generating arc length parameterized NURBS tool paths for efficient three-axis machining of smooth, accurate sculptured surfaces. International Journal of Advanced Manufacturing Technology, 70(5-8), 1355-1368.

Chen Z.T., Li S., Gan Z., Zhu Y. (2017). A highly efficient and convergent optimization method for multipoint tool orientation in five-axis machining. International Journal of Advanced Manufacturing Technology, 93(5-8), 2711-2722.

Cheng C.W. Tsai M.C. (2004). Real-time variable feed rate NURBS curve interpolator for CNC machining. International Journal of Advanced Manufacturing Technology, 23(11-12), 865–873.

Cheng M.-Y., Tsai M.-C., Kuo J.-C. (2002). Real-time NURBS command generators for CNC servo controllers. International Journal of Machine Tools and Manufacture, 42(7), 801-813.

Chipperfield A., Fleming P., Pohlheim H., Fonseca C. (1994). Genetic Algorithm Toolbox for use with Matlab – User's guide Version 1.2. Department of Automatic Control and System Engineering, University of Sheffield.

Cho H.D., Jun Y.T., Yang M.Y. (1993). Five-axis CNC milling for effective machining of sculptured surfaces. International Journal of Production Research, 31(11), 2559-2573.

Choi Y.-K., Banerjee A. (2007). Tool path generation and tolerance analysis for free-form surfaces. International Journal of Machine Tools and Manufacture 47(3-4), 689-696.

Chu C.H., Wu P.H., Lei W.T. (2012). Tool path planning for 5-axis flank milling of ruled surfaces considering CNC linear interpolation. Journal of Intelligent Manufacturing, 23(3), 471-480.

Chudasama C., Shah S.M., Panchal M. (2011). Comparison of Parents Selection Methods of Genetic Algorithm for TSP. International Journal of Computer Applications, Proceedings of the International Conference on Computer Communication and Networks CSI-COMNET-2011, pp. 85-87.

Cobb H., Grefenstette J. (1993). GA for Tracking Changing Environments. 5th International Conference on GA, pp. 1-8, Defense Technical Information Center, Morgan Kaufmann Publishers, Inc.

Coello Coello C., Lamont G.B., van Veldhuizen D.A. (2002). Evolutionary algorithms for solving multi-objective problems. eBook ISBN: 978-0-387-36797-2, Kluwer Academic Publishers, New York.

Das I., Dennis, J.E. (1997). A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. Structural Optimization 14(1), 63-69.

Dassault Systèmes CATIA V5 R18 online documentation about Sweeping tool path parameters (available on http://catiadoc.free.fr/online/smgug_C2/smgugrf0301.htm)

De Castro L.N., Timmis J. (2002). Artificial Immune Systems: A New Computational Intelligence Approach. ISBN: 978-1-85233-594-6, Springer-Verlag, London.

De Castro L.N., Von Zuben F.J. (2002). Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation, 6(3), 239-251.

De Jong K.A., Sarma J. (1993). Generation Gaps Revisited. In: Foundations of Genetic Algorithms, 2, Whitley, L.D. (Ed.), https://doi.org/10.1016/B978-0-08-094832-4.50007-6, Morgan Kaufmann Publishers, 1993.

Demuth H., Beale M. (1998). Neural network toolbox user's guide, version 3.0. Natick, MA: The Mathworks Inc., 1998.

Duvedi R.K., Bedi S., Batish A., Mann S. (2014). A multipoint method for 5-axis machining of triangulated surface models. Computer-Aided Design, 52, 17–26.

Duvedi R.K., Bedi S., Mann S. (2017). Drop and tilt method of five axis tool positioning for tensor product surfaces. International Journal of Advanced Manufacturing Technology, 93(1-4), 617-622.

El-Mounayri H., Kishawy H.A., Briceno J. (2005). Optimization of CNC ball end milling: a neural network-based model. Journal of Materials Processing Technology, 166(1), 50–62.

El-Mounayri H., Kishawy H.A., Tandon V. (2002). Optimized CNC end milling: a practical approach, International Journal of Computer Integrated Manufacturing, 15(5), 453–470.

Fan W., Ye P., Zhang H., Fang C., Wang R. (2013). Using rotary contact method for 5-axis convex sculptured surfaces machining. International Journal of Advanced Manufacturing Technology, 67(9-12), 2875-2884.

Fan W.G., Wang X.C., Cai Y.L., Jiang H. (2012). Rotary contact method for 5-axis tool positioning. Journal of Manufacturing Science and Engineering, 134 (2), 021004.1–021004.6.

Fausett L. (1994). Fundamentals of neural networks: architectures, algorithms, and applications, ISBN:0-13-334186-0, Prentice-Hall, Inc. Upper Saddle River, NJ, USA.

Feng H.-Y., Huiwen L. (2002). Constant scallop height tool path generation for three-axis sculptured surface machining. Computer-Aided Design, 34(9), 647-654.

Fisher, R. B. (1989). From surfaces to objects: Computer vision and three dimensional scene analysis. ISBN: 978-0-471-92344-2, John Wiley and Sons, Chichester, New York.

Forrest, S., Mitchell, M. (1993). Relative building-block fitness and the building-block hypothesis. In L. Darrell Whitley, editor, Foundations of Genetic Algorithms 2, pp. 109–126, Morgan Kaufmann, San Mateo, CA.

Fountas N., Vaxevanidis N., Stergiou C., Benhadj-Djilali R. (2015). Evaluation of 3- and 5-axis sculptured surface machining in CAM environment through design of experiments. International Journal of Computer Integrated Manufacturing, 28(3), 278-296.

Fountas N.A., Stergiou C.I., Vaxevanidis N.M., Benhadj-Djilali R. (2017b). A generic multi-axis post-processor engine for optimal CNC data creation and intelligent surface machining, Solid state phenomena, 261 SSP, pp. 463-469. https://www.scientific.net/SSP.261.463

Fountas N.A., Kechagias J.D., Benhadj-Djilali R., Stergiou C.I., Vaxevanidis N.M. (2014). Optimizing 5-Axis Sculptured Surface Finish Machining Through Design of Experiments and Neural Networks. Proc. ASME 12th Biennial Conference on Engineering Systems Design and Analysis,1, July 25–27, Copenhagen, Denmark, 2014. (Paper No. ESDA2014-20210).

Garg, M. P., Jain, A., & Bhushan, G. (2012). Modelling and multi-objective optimization of process parameters of wire electrical-discharge machining using non-dominated sorting genetic algorithm-II. Proceedings of Institution of Mechanical Engineers Part B: Journal of Engineering Manufacture, 226(12), 1986–2001.

Giri V., Bezbaruah D., Bubna P., Choudhury A.R. (2005). Selection of master cutter paths in sculptured surface machining by employing curvature principle. International Journal of Machine Tools and Manufacture 45(10), 1202–1209.

Gittens A., Chowdary B.V., Rao U.R.K. (2005). Investigation into the Effects of Toolpath and Feed Rate Variation on Sculptured Surface Machining. Proceedings of the 6th IASTED International Conference on Robotics and Applications, Oct. 31 – Nov. 2, 2005, Cambridge, MA.

Goldberg, D.E. (1989). Genetic algorithms in search, optimization and machine learning. ISBN:0201157675, Addison Wesley Publishing Co., Inc. Boston, MA, USA.

Gray P., Bedi S., Ismail F. (2003). Rolling ball method for 5-axis surface machining. Computer-Aided Design, 35(4), 347–357.

Gray P., Ismail F., Bedi S. (2003). Cusp modeling for 5-axis surface machining. Proceedings of the 8th ACM Symposium on Solid Modeling and Applications, pp. 340-345, Seattle, Washington, USA.

Gray P., Ismail F., Bedi S. (2004). Graphics-assisted rolling ball method for 5-axis surface machining. Computer-Aided Design, 36(7), 653-663.

Haykin S. (2009). Neural Networks and Learning Machines, 3rd Ed., ISBN-13: 978-0-13-147139-9, Prentice Hall, Upper Saddle River, New Jersey, 07458.

He Y., Chen Z., Xu R., Wu X. (2015). Reducing fluctuation of machining strip width by tool position modification for five-axis NC machining of sculptured surfaces. International Journal of Advanced Manufacturing Technology, 78(1-4), 249-257.

Holland J.H. (1975). Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. ISBN: 9780262275552, MIT Press, MA, USA.

Holland J.H. (2000). Building blocks, cohort genetic algorithms, and hyperplane-defined functions. Evolutionary Computation, 8(4), 373-391.

Hoos H.H., Stótzle T. (2004). Stochastic local search: foundations & applications. ISBN-13: 978-1558608726, Elsevier.

Jaggi S., Bhushan R., Malhotra D. (2013). Guidelines to Decide the Encoding Scheme Used For G.A. International Journal of Advanced Research in Computer Science and Software Engineering, 3(8), 1436-1440.

Jahanpour J., Alizadeh M.R. (2015). A novel acc-jerk-limited NURBS interpolation enhanced with an optimized S-shaped quintic feedrate scheduling scheme. International Journal of Advanced Manufacturing Technology, 77(9–12), 1889–1905.

Kayal P. (2007). Inverse offset method for adaptive cutter path generation from point-based surface. International journal of CAD/CAM, 7, 1-18.

Kersting P., Zabel A. (2009). Optimizing NC-tool paths for simultaneous five axis milling based on multi-population multi-objective evolutionary algorithms. Advances in Engineering Software, 40(6), 452-463.

Khalilpourazari S., Khalilpourazary S. (2018). Optimization of time, cost and surface roughness in grinding process using a robust multi-objective dragonfly algorithm. Neural Computing and Applications, pp. 1-12 https://doi.org/10.1007/s00521-018-3872-8.

Khalilpourazari S., Mohammadi M. (2016). Optimization of closed-loop supply chain network design: a Water Cycle Algorithm approach. In: 12th international conference of IEEE industrial engineering (ICIE), 25-26 Jan., pp. 41-45.

Khalilpourazari S., Pasandideh S.H.R. (2018). Multi-objective optimization of multi-item EOQ model with partial backordering and defective batches and stochastic constraints using MOWCA and MOGWO. Operational Research, pp. 1-33, https://doi.org/10.1007/s12351-018-0397-y.

Kirkpatrick S. (1984). Optimization by simulated annealing: quantitative studies. Journal of Statistical Physics, 34(5-6), 975–986.

Kirkpatrick S., Gelatt Jr C.D., Vecchi, M.P. (1983). Optimization by simulated annealing. Science, 220, (4598), 671–680.

Krimpenis A., Fousekis A., Vosniakos G. (2005). Assessment of sculptured surface milling strategies using design of experiments. International Journal of Advanced Manufacturing Technology, 25(5-6), 444–453.

Krimpenis A., Vosniakos G.C. (2004). Optimisation of roughing strategy for sculptured surface machining using genetic algorithms and neural networks. 8th International Conference on Production Engineering, Design and Control 2004 (PEDAC2004), Alexandria, Egypt, 2004.

Krimpenis A.A., Fountas N.A. (2016). Balancing multiple criteria in formulation of weighted, single-objective genetic algorithm optimization for CNC machining problems. Advances in Manufacturing, 4(2), 178-188.

Kumar A. (2013). Encoding Schemes in Genetic Algorithm. International Journal of Advanced Research in IT and Engineering, 2(3), 1-7.

Kumar V., Sambhav K., Tandon P. (2015). Force Modeling in Ball-end Milling and its Application to Sculptured Surface Machining. Journal of Material Science and Mechanical Engineering, 2(6), 4-8.

Kuriachen, B., Somashekhar, K.P., Mathew J. (2015). Multi response optimization of micro-wire electrical discharge machining process. International Journal of Advanced Manufacturing Technology, 76(1-4), 91–104.

Lamikiz A., Lopez De Lacalle L.N., Sanchez J. A., Salgado M.A. (2004). Cutting force estimation in sculptured surface milling. International Journal of Machine Tools and Manufacture, 44(14), 1511–1526.

Lartigue C., Duc E., Tournier C. (1999). Machining of free-form surfaces and geometrical specifications. Proceedings of the Institution of Mechanical Engineers Part B: Journal of Engineering Manufacture, doi.org/10.1243/0954405991516615.

Lasemi A., Xue D., Gu P. (2014). Tool path re-planning in free-form surface machining for compensation of process-related errors. International Journal of Production Research, 52(20), 5913-5931.

Lavernhe S., Tournier C., Lartigue C. (2008). Kinematical performance prediction in multi-axis machining for process planning optimization. International Journal of Advanced Manufacturing Technology, 37 (5-6), 534-544.

Lazoglu I. (2003). Sculpture Surface Machining: A Generalized Model of Ball-End Milling Force System, International Journal of Machine Tools and Manufacture, 43(5), 453-462.

Lazoglu I., Manav C., Murtezaoglu Y. (2009). Tool path optimization for free form surface machining. CIRP Annals Manufacturing Technology, 58(1), 101–104.

Lazoglu I., Liang S.Y. (1997). Analytical Modeling of Force System in Ball-End Milling. Machining Science and Technology, 1(2), 219-234.

Lee S.G., Yang Y.H. (2002). CNC tool-path planning for high speed high-resolution machining using a new tool-path calculation algorithm. International Journal of Advanced Manufacturing Technology, 20(5), 326-333.

Lee Y.-S., Chang T.-C. (1996). Automatic cutter selection for 5-axis sculptured surface machining. International Journal of Production Research, 34(4), 977-998.

Li L., Liu F., Chen B., Li B.C. (2015). Multi-objective optimization of cutting parameters in sculptured parts machining based on neural network. Journal of Intelligent Manufacturing, 26(5), 891-898.

Li S.X. Jerard R.B. (1994). 5-axis machining of sculptured surfaces with a flat-end cutter. Computer-Aided Design, 26(3), 165-178.

Li Z., Chen W. (2006). A global cutter positioning method for multi-axis machining of sculptured surfaces. International Journal of Machine Tools and Manufacture, 46(12-13), 1428-1434.

Lin R.S., Koren Y. (1996). Efficient tool-path planning for machining free-form surfaces. Journal of Engineering for Industry, 118(1), 20-28.

Lin Z., Fu J., Shen H., Gan W. (2014). An accurate surface error optimization for five-axis machining of freeform surfaces. International Journal of Advanced Manufacturing Technology 71(5-8), 1175-1185.

Liu H., Liu Q., Zhou, S., Li C., Yuan S. (2015). A NURBS interpolation method with minimal feedrate fluctuation for CNC machine tools. International Journal of Advanced Manufacturing Technology, 78(5–8), 1241–1250.

López de Lacalle L.N., Lamikiz A., Muñoa J., Sánchez J.A. (2005). The CAM as the centre of gravity of the five-axis high speed milling of complex parts. International Journal of Production Research, 43(10), 1983-1999.

Lopez De Lacalle L.N., Lamikiz A., Sanchez J.A., Salgado M.A. (2007). Toolpath selection based on the minimum deflection cutting forces in the programming of complex surfaces milling. International Journal of Machine Tools and Manufacture, 47(2), 388–400.

Lu A., Ding Y., Zhu L.M. (2017). Tool path generation via the multi-criteria optimization for flat-end milling of sculptured surfaces. International Journal of Production Research, 55(15), 4261-4282.

Lu Y.A., Ding Y., Zhu L.M. (2016). Simultaneous Optimization of the Feed Direction and Tool Orientation in five-axis Flat-end Milling. International journal of Production Research, 54(15), 4537-4546.

Makhanov S.S., Batanov D., Bohez E., Sonthipaumpoon K., Anotaipaiboon W., Tabucanon M. (2002). On the tool-path optimization of a milling robot. Computers and Industrial Engineering, 43(3), 455-472.

Manav C., Bank H.S., Lazoglu I. (2013). Intelligent tool path selection via multi-criteria optimization in complex sculptured surface milling. Journal of Intelligent Manufacturing, 24(2), 349-355.

Markopoulos A.P., Georgiopoulos S., Manolakos D.E. (2016). On the use of back propagation and radial basis function neural networks in surface roughness prediction. Journal of Industrial Engineering International, 12(3), 389–400.

Martínez M., Herrero J.M., Sanchis, J., Blasco X., García-Nieto S. (2009). Applied Pareto multi-objective optimization by stochastic solvers. Engineering Applications of Artificial Intelligence, 22(3), 455-465.

Mayor J.R., Sodemann, A.A. (2008). Intelligent Tool-Path Segmentation for Improved Stability and Reduced Machining Time in Micromilling. ASME Journal of Manufacturing Science and Engineering, 130(3), 311211-3112113.

Mendes J.M. (2013). A Comparative Study of Crossover Operators for Genetic Algorithms to Solve the Job Shop Scheduling Problem. WSEAS Transactions on computers 4(12), 164-173.

Mirjalili S. (2015). The ant-lion optimizer. Advances in Engineering Software, 83, 80-98.

Mirjalili S. (2016). Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Computing and Applications, 27(4), 1053-1073.

Mirjalili S., Jangir P., Saremi S. (2017). Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems. Applied Intelligence, 46(1), 79-95.

Mirjalili S., Mirjalili S.M., Hatamlou A. (2016). Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications, 27(2), 495–513.

Mirjalili S., Mirjalili S.M., Lewis A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46-61.

Mitchell, M., 1999. An introduction to genetic algorithms. ISBN:0262631857, MIT Press, MA, USA.

Montgomery D.C. (2017). Design and Analysis of Experiments, 9th edition, ISBN: 978-1-119-11347-8, Wiley, New York.

Myers R.H. (1971). Response surface methodology. Allyn and Bacon, Boston.

Myers R.H., Montgomery D.C., Anderson-Cook C.M. (1995). Response surface methodology: Process and Product Optimization Using Designed Experiments, 4th Ed., ISBN: 978-1-118-91601-8, John Wiley & Sons, New York.

Ortiz-Boyer D., Hervas-Martinez, C., Garcia-Pedrajas N. (2005). A Crossover Operator for Evolutionary Algorithms Based on Population Features. Journal of Artificial Intelligence Research, 24, 1-48.

Oysu C., Bingul Z. (2009). Application of heuristic and hybrid-GASA algorithms to tool-path optimization problem for minimizing airtime during machining. Engineering Applications of Artificial Intelligence, 22(3), 389-396.

Palanikumar K., Latha B., Senthilkumar V.S., Karthikeyan R. (2009). Multiple performance optimization in machining of GFRP composites by a PCD tool using non-dominated sorting genetic algorithm (NSGA-II). Metals and Materials International, 15(2), 249–258.

Pandey A.K., Dubey A.K. (2012). Simultaneous optimization of multiple quality characteristics in laser cutting of titanium alloy sheet. Optics and Laser Technology, 44(6), 1858–1865.

Quinsat Y., Sabourin L. (2006). Optimal selection of machining direction for three-axis milling of sculptured parts. International Journal of Advanced Manufacturing Technology, 27(11-12), 1132–1139.

Rao N., Ismail F., Bedi S. (1997). Tool path planning for five-axis machining using the principal axis method. International Journal of Machine Tools and Manufacture, 37(7), 1025-1040.

Rao R.V., Rai D.P, Balic J. (2016). Multi-objective optimization of machining and micro-machining processes using non-dominated sorting teaching–learning-based optimization algorithm. Journal of Intelligent Manufacturing, 29(8), 1715-1737.

Redonnet J.M., Djebali S., Segonds S., Senatore J., Rubio, W. (2013). Study of the effective cutter radius for end milling of free-form surfaces using a torus milling cutter. Computer-Aided Design, 45(6), 951-962.

Redonnet J.M., Vazquez A.G., Michel A.T., Segonds S. (2016). Optimization of free-form surface machining using parallel planes strategy and torus milling cutter. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, DOI:10.1177/0954405416640175.

Reynolds C.W. (1987). Flocks, herds and schools: a distributed behavioral model. ACM SIGGRAPH Computer Graphics, 21(4), 25-34.

Roman A., Barocio E., Huegel J.C. Bedi S. (2015). Rolling ball method applied to 3½½-axis machining for tool orientation and positioning and path planning. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 7(12), 1-12.

Romero-Hdz J., Aranda S., Toledo-Ramirez G., Segura J., Saha B. (2016). An Elitism Based Genetic Algorithm for Welding Sequence Optimization to Reduce Deformation. Research in Computing Science, 121, 17-36.

Ross J.P. (1996). Taguchi Techniques for Quality Engineering 2nd Ed., ISBN-13: 978-0070539587, McGraw-Hill, New York.

Ross, P.J. (1996). Taguchi Techniques for Quality Engineering. ISBN-13: 978-0070539587, McGraw-Hill International Editions, New York.

Russell R.W., May M.L., Soltesz K.L., Fitzpatrick J.W. (1998). Massive swarm migrations of dragonflies (Odonata) in eastern North America. *The American Midland Naturalist*, 140(2), 325-342.

Saroj A.K., Jayswal S.C. (2013). Analysis of Different Parameters on Tool Path for Machining Sculptured Surfaces. International Journal of Engineering Research and Technology, 2(10), 369-379.

Segonds S., Seitier P., Bordreuil C., Bugarin F., Rubio W., Redonnet J.M. (2017). An analytical model taking feed rate effect into consideration for scallop height calculation in milling with torus-end cutter. Journal of Intelligent Manufacturing, DOI 10.1007/s10845-017-1360-0.

Siemens$^{©}$ AG SINUMERIK, Manual, 5-axis machining, 5/2009.
https://cache.industry.siemens.com/dl/files/454/37335454/att_110322/v1/SIN_WF5_0509_en.pdf).

Stahovec J., Kandráč L. (2013). Optimization of cutting conditions for the reduction cusp height in the milling process. Transfer inovácií 25, 244-248.

Sun Y., Zhao Y., Bao Y., Guo D. (2014). A novel adaptive feedrate interpolation method for NURBS tool path with drive constraints. International Journal of Machine Tools and Manufacture, 77, 74–81.

Suresh K., Yang. D. (1994). Constant scallop-height machining of free-form surfaces. Journal of Engineering for Industry, 1(16), 273-283.

Taguchi G. (1986). Orthogonal Arrays and Linear Graphs, ASIN: B000710HIW, American Supplier Institute, Inc., Dearborn, Ml.

Thierens D. (2002). Adaptive Mutation Rate Control Schemes in Genetic Algorithms. Proceedings of the 2002 Congress on Evolutionary Computation, 12-17 May, Print ISBN: 0-7803-7282-4, Honolulu, USA.

Tournier C., Duc E. (2005). Iso-scallop tool path generation in 5-axis milling, International Journal of Advanced Manufacturing Technology, 25 (9-10), 867-875.

Ulker E., Turanalp M.E., Halkaci, H.S. (2009). An artificial immune system approach to CNC tool path generation. Journal of Intelligent Manufacturing, 20(1), 67-77.

Vickers G.W., Quan, K.W. (1989). Ball-mills versus end-mills for curved surface machining. ASME Journal of Engineering for Industry, 111 (1), 22-26.

Warkentin A., Hoskins P., Ismail F., Bedi S. (2001). *Computer Aided 5-Axis Machining*, in: Systems, Techniques and Computational Methods volume 1 of Computer-Aided Design, Engineering and Manufacturing: Systems, Techniques and Applications, Chapter 3, 3001-3034, CRC Press, Inc., Boca Raton, FL, USA.

Warkentin A., Ismail F., Bedi S. (2000). Comparison between multipoint and other 5-axis tool position strategies. International Journal of Machine Tools and Manufacture, 40(2), 185-208.

Whitley D., Kauth J. (1988). GENITOR: A different Genetic Algorithm. Proceedings of the 1988 Rocky Mountain Conference on Artificial Intelligence, pp. 118-130. Computer Science Department, Colorado State University, Denver.

Wikelski M., Moskowitz D., Adelman J.S., Cochran J., Wilcove D.S., May M.L. (2006). Simple rules guide dragonfly migration. Biology Letters, 2(3), 325-329.

Wolpert D.H., Macready W.G. (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, 1(1), 67–82.

Xu R., Zhitong C., Wuyi C., Xianzhen W., Jianjun Z. (2010). Dual drive curve tool path planning method for 5-axis NC machining of sculptured surfaces. Chinese Journal of Aeronautics, 23(4), 486-494.

Yang J., Altintas Y. (2015). A generalized on-line estimation and control of five-axis contouring errors of CNC machine tools. International Journal of Machine Tools and Manufacture, 88, 9-23.

Yang M., Hong W. (2002). Three-Dimensional Reference Pulse Linear and Circular Interpolators for CNC Systems. International Journal of Production Research, 40(2), 425–439.

Yang S. (2007). Genetic Algorithms with Elitism-Based Immigrants for Changing Optimization Problems. In Giacobini, M., et al. (Eds.): EvoWorkshops 2007, LNCS 4448, 627-636, Springer-Verlag, Berlin, Heidelberg.

Yeh S., Hsu P. (2002). Adaptive-Feedrate Interpolation for Parametric Curves With a Confined Chord Error. Computer-Aided Design, 34(3), 229–237.

Zain A.M., Haron H., Sharif S. (2010). Application of GA to optimize cutting conditions for minimizing surface roughness in end milling machining processes. Expert Systems with Applications, 37(6), 4650-4659.

Zeroudi N., Fontaine M. (2015). Prediction of tool deflection and tool path compensation in ball-end milling. Journal of Intelligent Manufacturing, 26(3), 425-445.

Zeroudi N., Fontaine M., Necib K. (2012). Prediction of cutting forces in 3-axes milling of sculptured surfaces directly from CAM tool path. Journal of Intelligent Manufacturing, 23(5), 1573-1587.

Zhang D., Yang, P. Qian X. (2009). Adaptive NC Path Generation From Massive Point Data With Bounded Error. Journal of Manufacturing Science and Engineering, 131(1), 0110011 - 01100113.

Zhou B., Zhao J., Li L. (2015). CNC double spiral tool-path generation based on parametric surface Mapping. Computer-Aided Design, 67–68, 87-106.

Appendix A

Research validation document from Hellenic Aerospace Industry

HELLENIC AEROSPACE INDUSTRY S.A.
Athens Tower, Messogion 2-4, GR 115 27 Athens, Greece
Tel: 210-77.99.622, Fax: 210- 77.97.670
Tanagra, P.O.Box 23, 320 09 Schimatari
Tel.: 22620-52000, Fax: 210-88.38.714, 22620-52170

From : AEROSTRUCTURES PLANT DIRECTORATE

Date: January 9th, 2017
Reg # : 3400-2017-01-2

To : To whom it may concern

Subject : CNC process optimization

This is to certify that Mr. Nikolaos Fountas visited our manufacturing facilities (Hellenic Aerospace Industry - HAI Greece) on October 21, 2016 to kindly ask for our support on machining a benchmark complex component for validating his PhD research, dealing with the optimization of sculptured surface CNC machining tool-paths by means of artificial intelligence technologies. The complex part that Mr. Fountas prepared for validating his scientific proposals, was an industrial component with changing curvatures to both "u" and "v" parametric directions, hence; successfully reflecting the actual manufacturing challenges faced when machining similar components.

Mr. Fountas successfully provided insight on the fundamentals of his methodology, explaining its industrial impact to actual applications; as well as mentioning to what extend current shops' practices could be improved by implementing such a methodology for optimizing surface machining parameters to create complex CNC part programs. Thereby, Mr. Fountas was supported by experienced process planners to machine and inspect the test part using HAI's high-technology equipment.

At an early stage our experts found noticeable merits on the simultaneous (full) 5-axis machining operation conducted according to Mr. Nikolaos Fountas' proposals. Both surface finish as well as machining time were found in excellent levels. Mr. Fountas took advantage of our measurements on his benchmark component to produce research results for his methodology to become even more prominent and further interpret the outputs, whilst his achievements were compared to a series of others already published for the same problem and the same level of complexity. It was clear that the final result was far more beneficial than the other previously used approaches.

Therefore, we can ensure to a large extend that, the research developed by Mr. Nikolaos Fountas has a great industrial impact on challenging manufacturing activities such as in sculptured surface CNC machining. We are positive on a collaboration with Piraeus University of Applied Sciences and Kingston University London which are involved in the aforementioned methodology development.

Sincerely yours,

....
P. Mermelas
Mechanical Engineer
CNC Programming

....
D. Chorianopoulos
Production Support Manager

....
D.-K. Kampouroglou
Aerostructures Plant Director

Research results for the benchmark sculptured parts from the literature

B1. Research results from Xu et al. (2010)
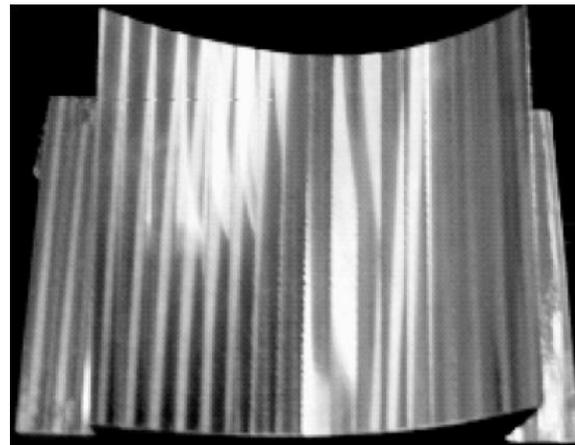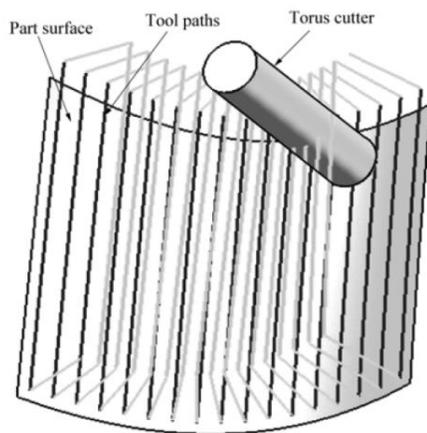
Simulated tool paths and actual part machined using the method of Xu et al. (2010)
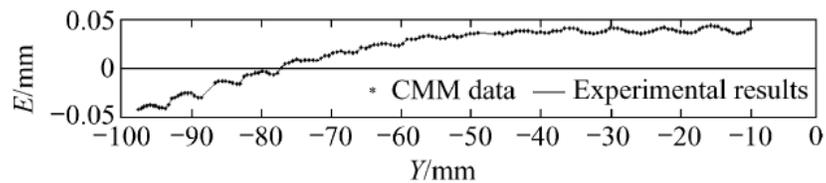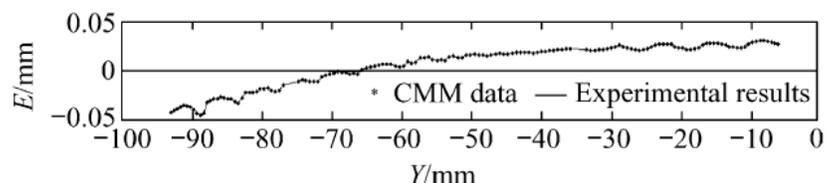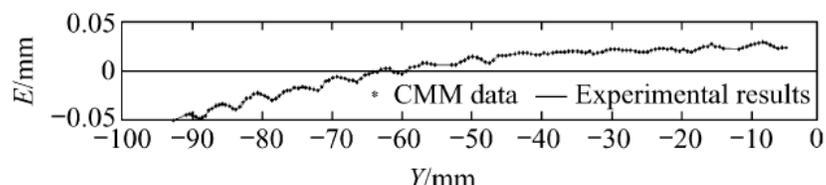
Measured results of the machine surface using the method of Xu et al. (2010).

B2. Research results from Gan et al. (2016)

Range	Color
0.05000	13:Cyan
0.05000	36:Blue Violet
0.03000	14:Green
0.02000	27:Deep Pink
0.01000	28:Dodger Blue
0.00500	6:Light Goldenrod
0.00200	15:Blue
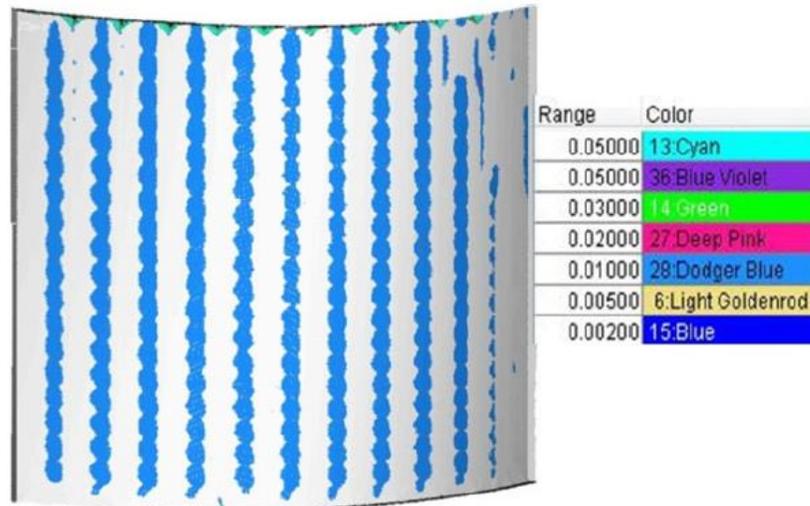
Machining simulation result and actual part machined using the "mechanical equilibrium method – MEM" of Gan et al. (2016).

B3. Research results from Warkentin et al. (2000) for "inclined tool method – ITM", "principal axis method – PAM" and "multi-point machining – MPM"
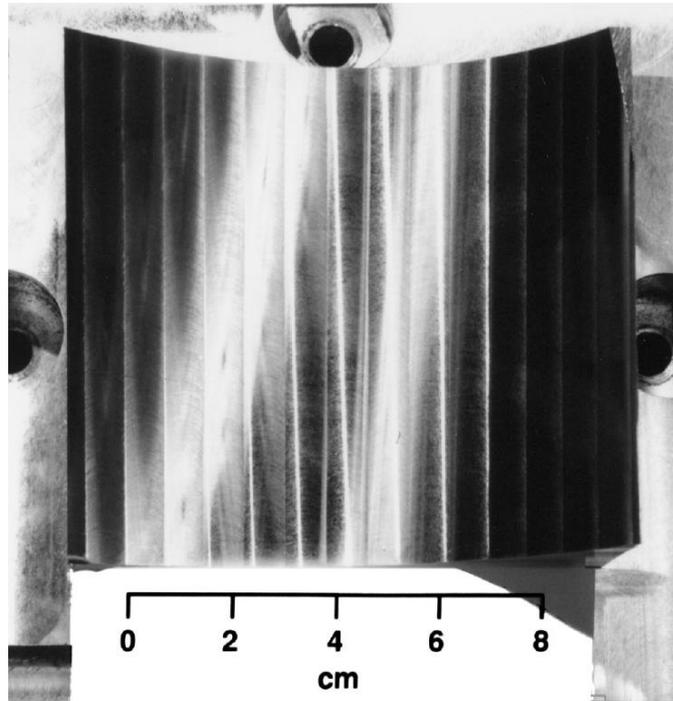
Test surface machined using the "inclined tool method – ITM" with D16Rc3 mm toroidal end-mill, 8mm tool pass interval (stepover 50%) and 6º inclination angle.

Test surface machined using the "principal axis method – PAM" with D16Rc3 mm toroidal end-mill and 8mm tool pass interval (stepover 50%).
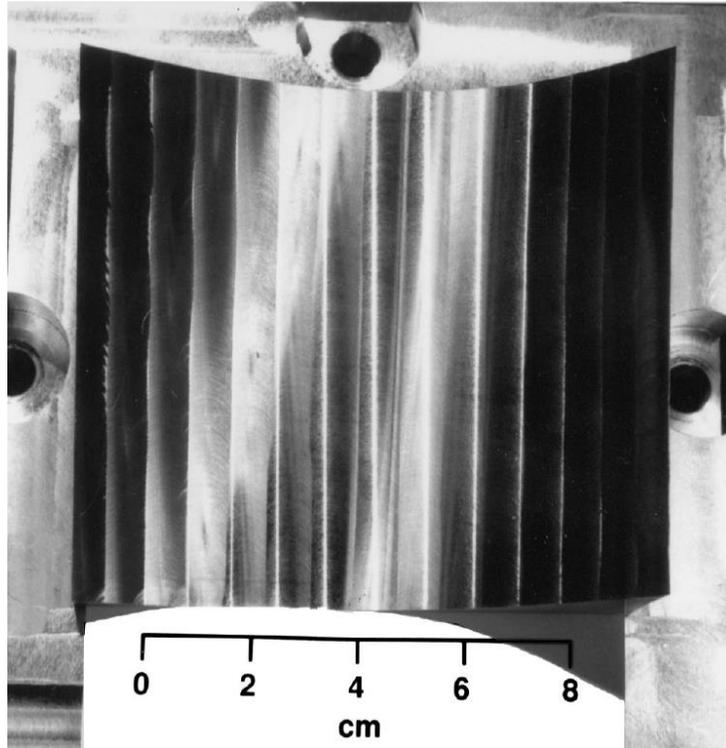
Test surface machined using the "multi-point method – MPM" with D16Rc3 mm toroidal end-mill, 8mm tool pass interval (stepover 50%) and 0.8 separation ratio for the two contact points.
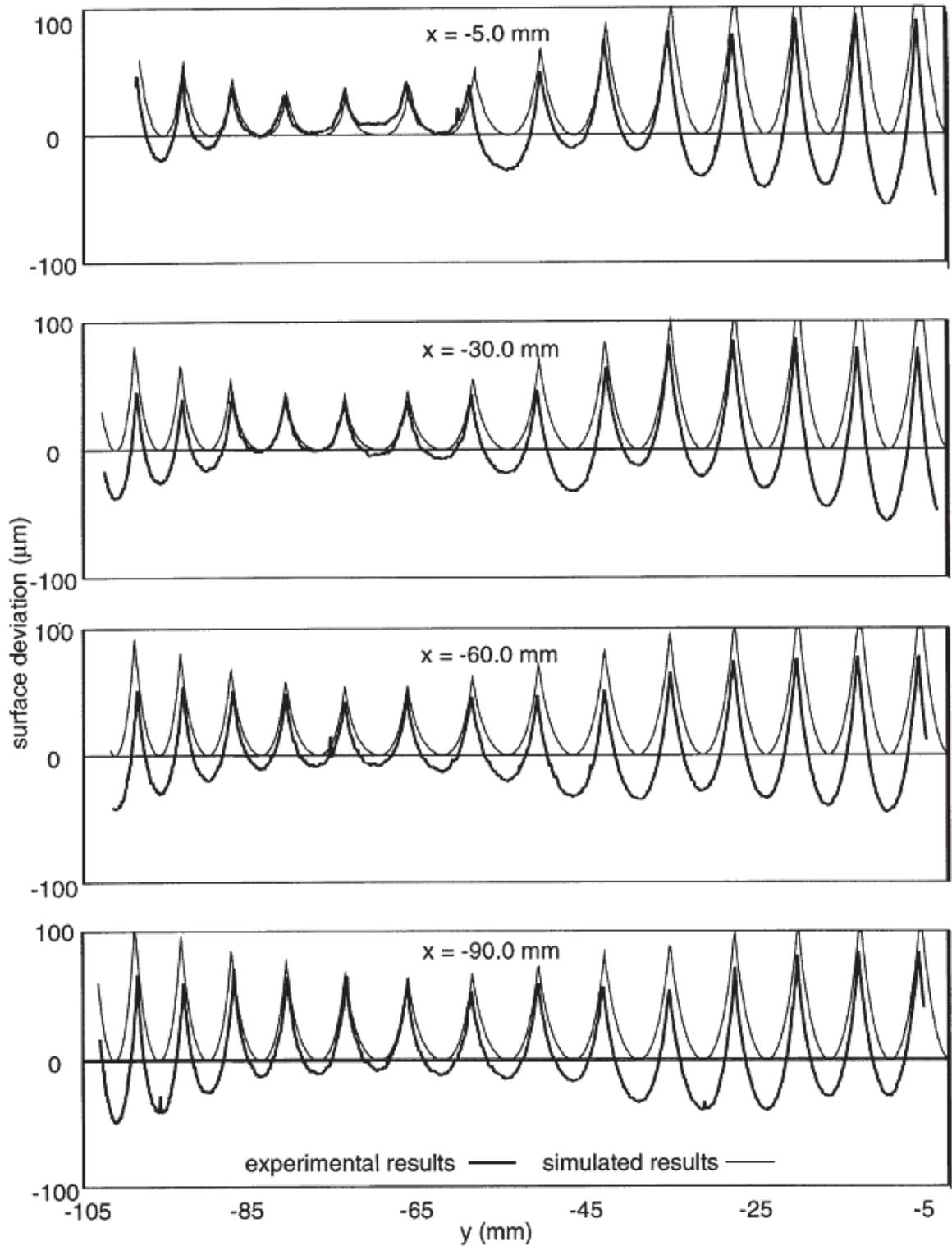
Comparison of experimental and simulated results for "inclined tool method – ITM" using a D16Rc3 toroidal end-mill with 8.0mm tool pass interval (stepover 50%) and 6° inclination angle.
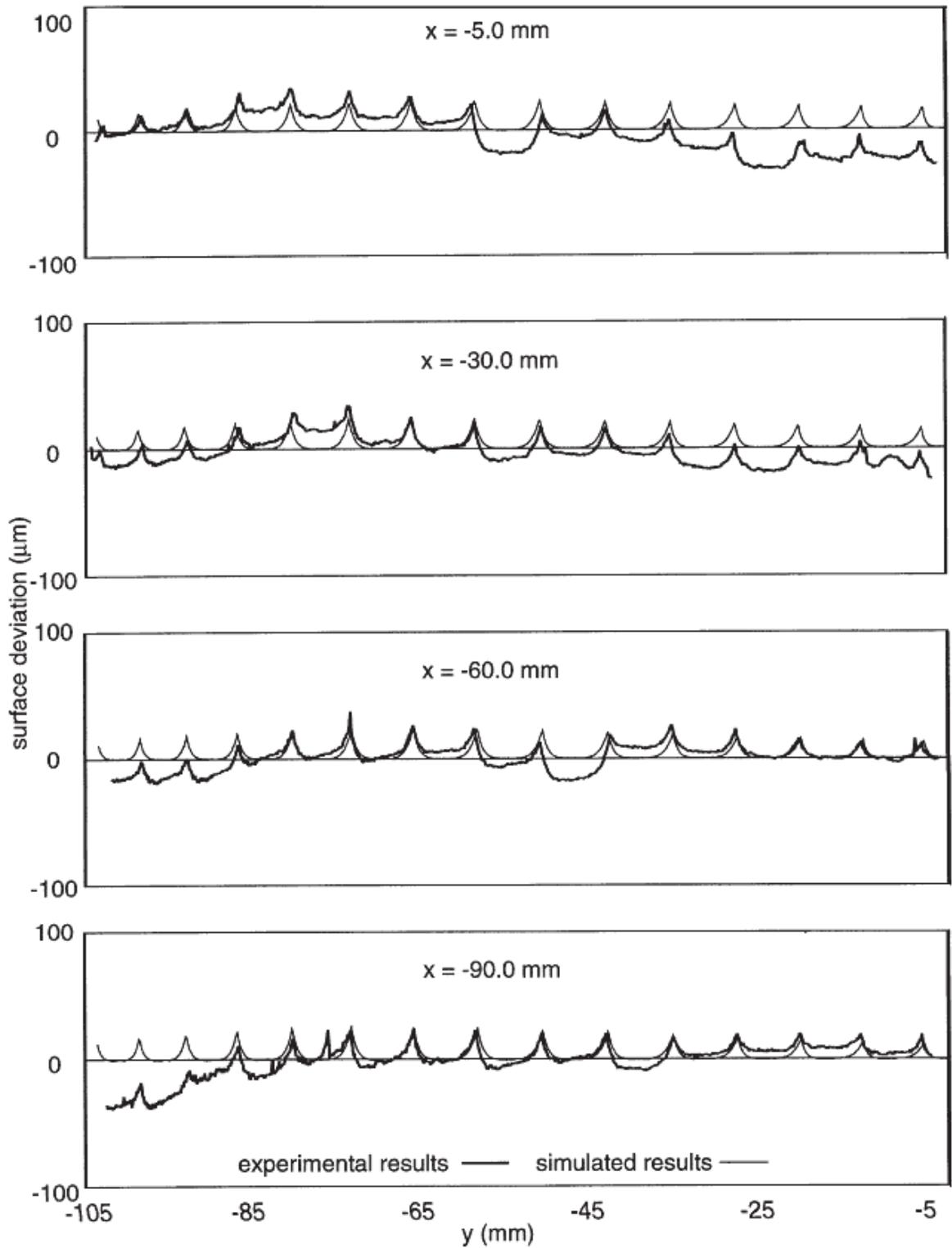
Comparison of experimental and simulated results for "principal axis method – PAM" using a D16Rc3 toroidal end-mill with 8.0mm tool pass interval (stepover 50%).
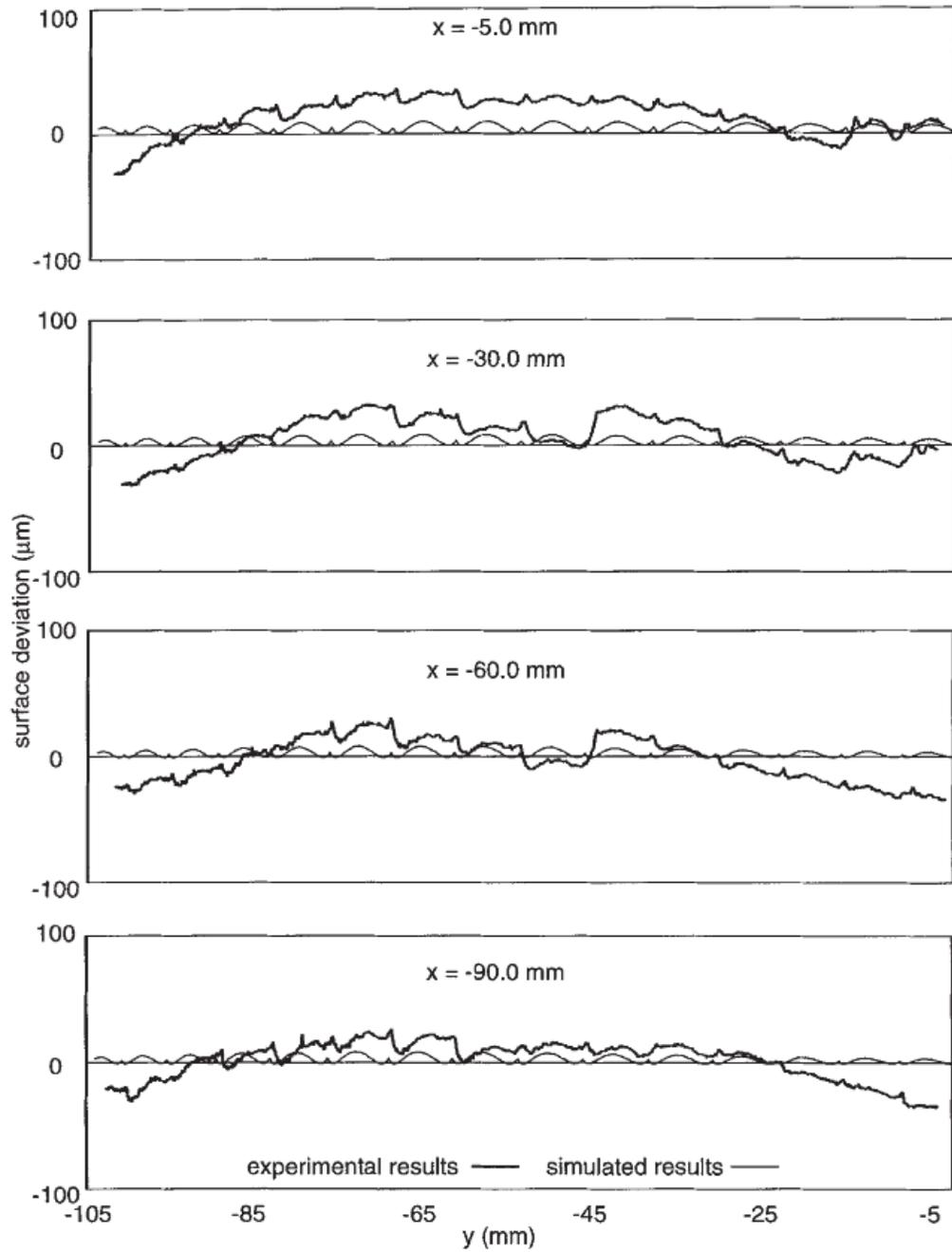
Comparison of experimental and simulated results for "multi-point method – MPM" using a D16Rc3 toroidal end-mill with 8.0mm tool pass interval (stepover 50%) and 0.8 separation ratio between the two contact points.

246

B4. Research results from Chen et al. (2017)

0.05000	36:Blue Violet
0.03500	13:Cyan
0.03000	5:Plum
0.02000	16:Yellow
0.01200	27:Deep Pink
0.00500	32:Green Yellow
0.00200	15:Blue
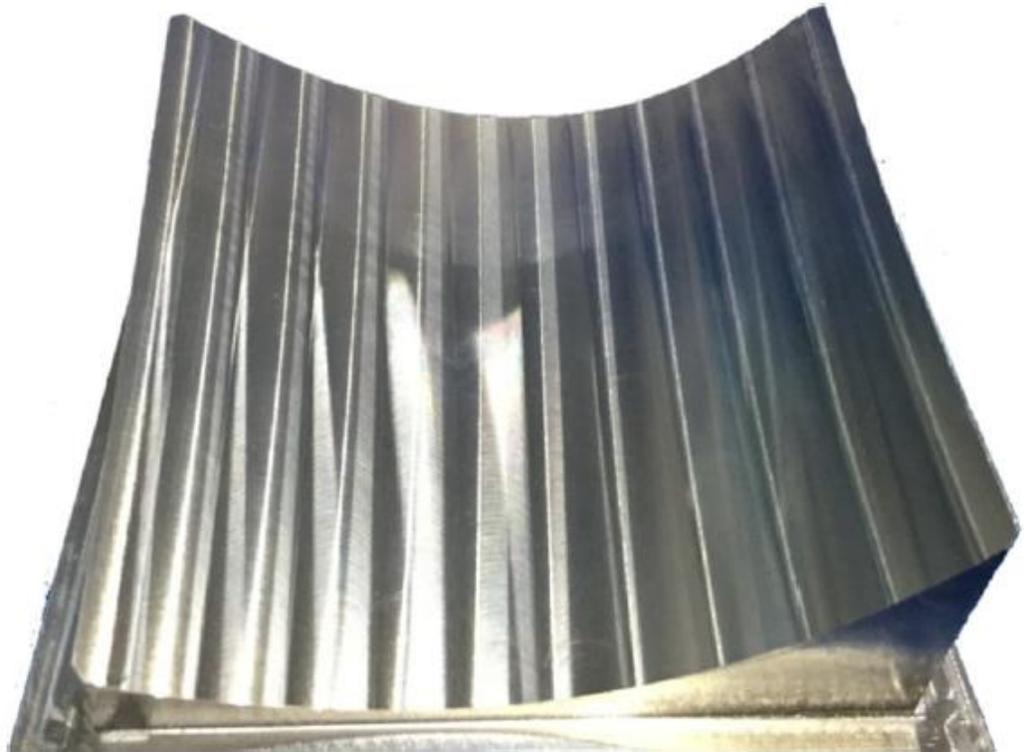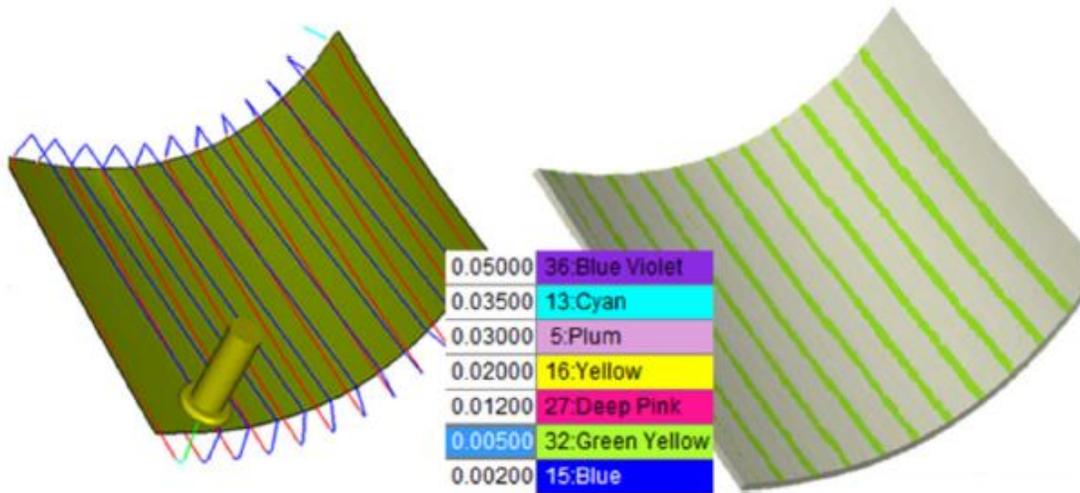
Machining simulation result and actual part machined using the "efficient convergent optimization method – FCO" of Chen et al. (2017).

B5. Resulting benchmark sculptured surface from Fountas et al. (2017)
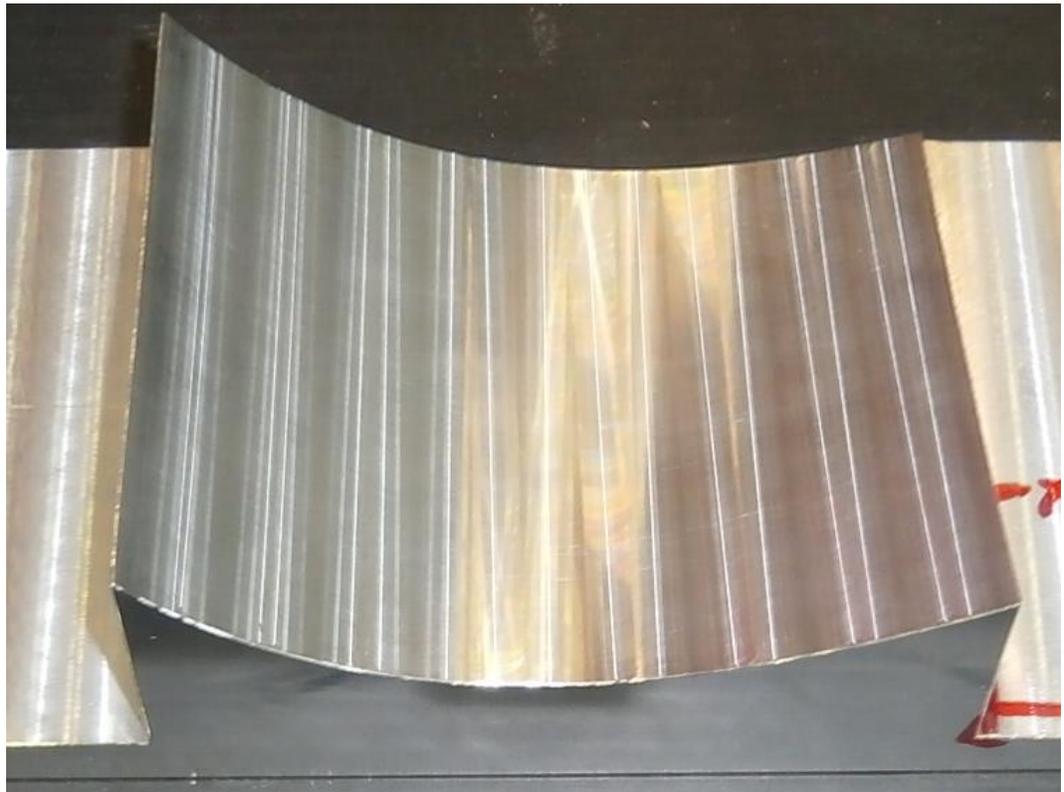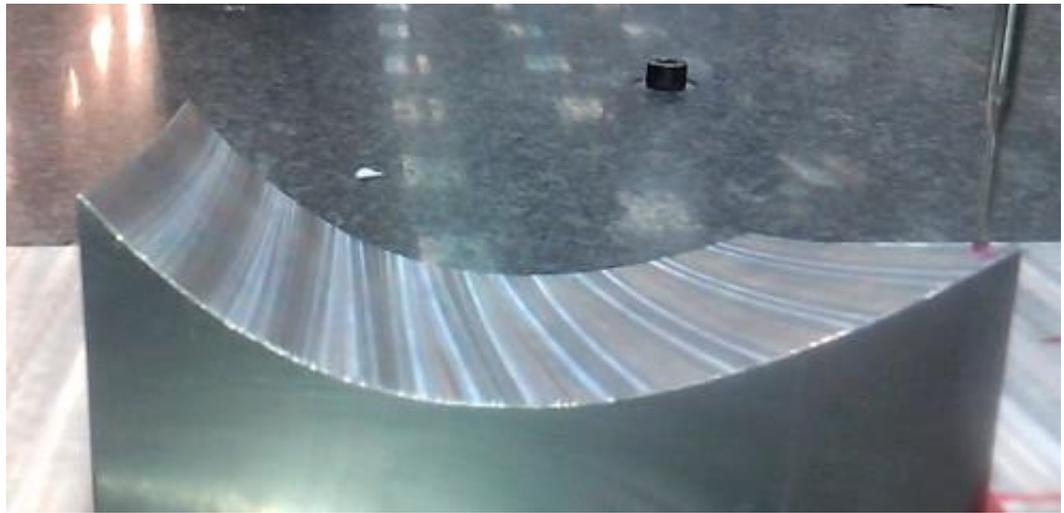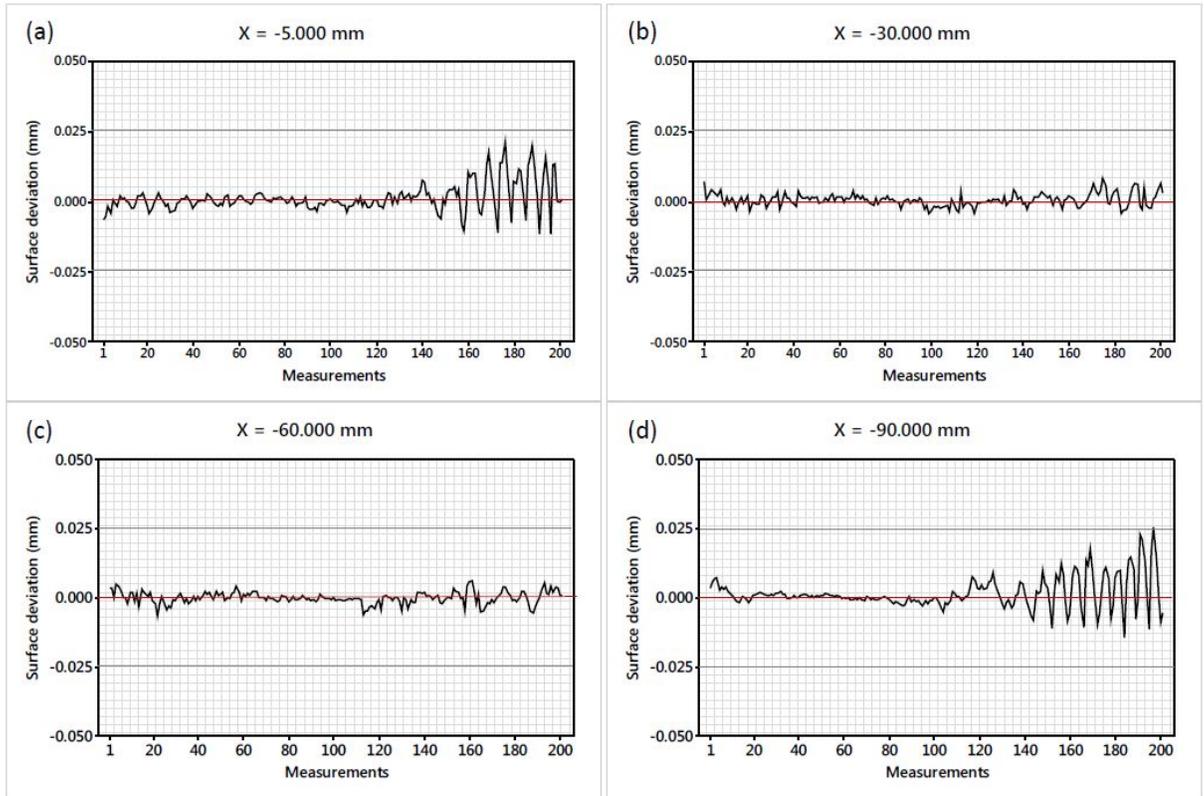
Actual machining result and measured outputs using the proposed intelligent optimisation methodology for sculptured surface CNC machining tool paths.