2001

Automatic sampling for CMM inspection planning of free form surfaces.

Diaa F. ElKott
University of Windsor

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

AUTOMATIC SAMPLING FOR CMM INSPECTION PLANNING OF FREE FORM SURFACES

By

Diaa F. ElKott

A Thesis Submitted to
the Faculty of Graduate Studies and Research
through the Program of Industrial and Manufacturing Systems Engineering
in partial fulfillment of the requirements for
the degree of Master in Applied Science
at the University of Windsor.

Windsor, Ontario, Canada

2001

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-62210-X

Canada

ABSTRACT

The advance in design, and manufacturing technologies has made it possible to design, and manufacture products with high degrees of irregularity, such as free form surfaces. These products are utilized in the aircraft, shipbuilding, automotive, biomedical, and tool and die industries. Coordinate Measuring Machines (CMMs) are used to examine the conformity of the produced parts with the designer's intention. The inspection of free form surfaces is a difficult process due to their complexity, and irregularity.

Many tasks are performed to ensure a reliable and efficient inspection using CMMs. Sampling is an essential and vital step in inspection planning. It is a major contributor to the CMM measurement uncertainty. This research focuses on developing efficient and reliable approaches to determine the locations of the points to be sampled from free form surfaces using the CMM.

Four heuristic algorithms for the sampling of free form surfaces have been developed. Optimal sampling of free form surfaces using Genetic Algorithms has been

introduced. An algorithm for the automatic selection of sampling algorithm based on the surface complexity is developed. The developed sampling algorithms have been implemented, and integrated into a computer-aided system for the sampling of free form surfaces.

Extensive simulations have been performed on the developed methodologies to test their performance using free form surfaces with different degrees of complexity. Within the simulation study, the CMM measurement error, and the manufacturing form error have been simulated. The developed sampling approaches were compared to the uniform sampling pattern.

The heuristic methodologies proved to be efficient in producing reliable sampling strategies to a wide range of surface complexity. When the surface reaches a high complexity level, finding an optimal sampling plan becomes crucial. In this case, Genetic Algorithms demonstrated superiority over the various sampling methodologies.

The developed algorithms are intended to fill voids which exist in previous works in sampling of free form surfaces. These are: offering alternative solutions to the sampling problem, optimal sampling of free form surfaces, and automatic selection of sampling algorithms. They are intended to be candidates for integration with computer aided tactile inspection planning systems.

To The Memory of Mohammad AlDurrah

ACKNOWLEDGMENTS

LIST OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION AND BACKGROUND

This chapter presents a background to the principal concepts, and methodologies utilized throughout the course of this study. The first section discusses various aspects of coordinate metrology. The second section is a brief discussion of inspection planning, and a literature survey of the work done in the field of automation of inspection planning. The third section introduces an overview of geometric modeling of free form surfaces, with emphasis on Non Uniform Rational B-Splines (NURBS). Section four is a brief review of optimization, with emphasis on Genetic Algorithms (GA's). Section five presents a problem statement. Finally, section six presents an overview of this thesis.

1.1. Coordinate Metrology

The basic function of coordinate metrology consists of the measurement of the actual shape of a workpiece, its comparison with the desired shape, and the evaluation of the metrological information such as size, form, location and orientation (Ni and Wäldele, 1995). The actual shape of a workpiece is obtained by probing its surface at certain points with the CMM touch probe which returns the locations in space of each point, processing

the data collected to extract the required substitute geometry of the workpiece, and then comparing the substitute geometry with the workpiece design data to determine whether the workpiece features lie within the specified design tolerances. Two techniques are widely used for the measurement of a workpiece using a CMM. Those are contact, and non-contact measurements. In contact measurement, an electronic device, known as the probe, tactile, or touch trigger, touches the surface of the workpiece. Upon touching the workpiece, the probe returns an electrical pulse that tells the machine the location in space of the contact point. Probed points are then processed to obtain the workpiece substitute geometry. Non-contact methods are based on optical systems which use electron or laser beams to measure the dimensions of various features. Non-contact methods return clouds of data that are to be processed to generate the substitute geometry.

Non-contact measurement methods are fast and capable of collecting a large amount of data. However, the capital cost of non-contact measurement equipment is significant, and elaborate software techniques are needed to efficiently handle the large amount of data scanned by these methods.

Measurement accuracy in the range of 2.50-25.0 μm with excellent repeatability is achievable with contact measurement methods (ElMaraghy, H. A., and ElMaraghy, W. H., 1994). This gave inspection with touch probes an obvious advantage over the non-

contact measurement methods which provide measurement accuracy in the range of 25-50 μm. Aside from the high accuracy of touch probes, they are also cheaper than non-contact techniques. Both methods are complementary. Each has its own pros and cons. The most promising approach is a combination of both tactile and non-contact coordinate measurement methods (ElMaraghy, H. A., and ElMaraghy, W. H., 1994). Figure 1.1 illustrates a CMM measuring a mechanical part. The inset of figure 1.1 shows a P-9 Ranishaw touch probe.
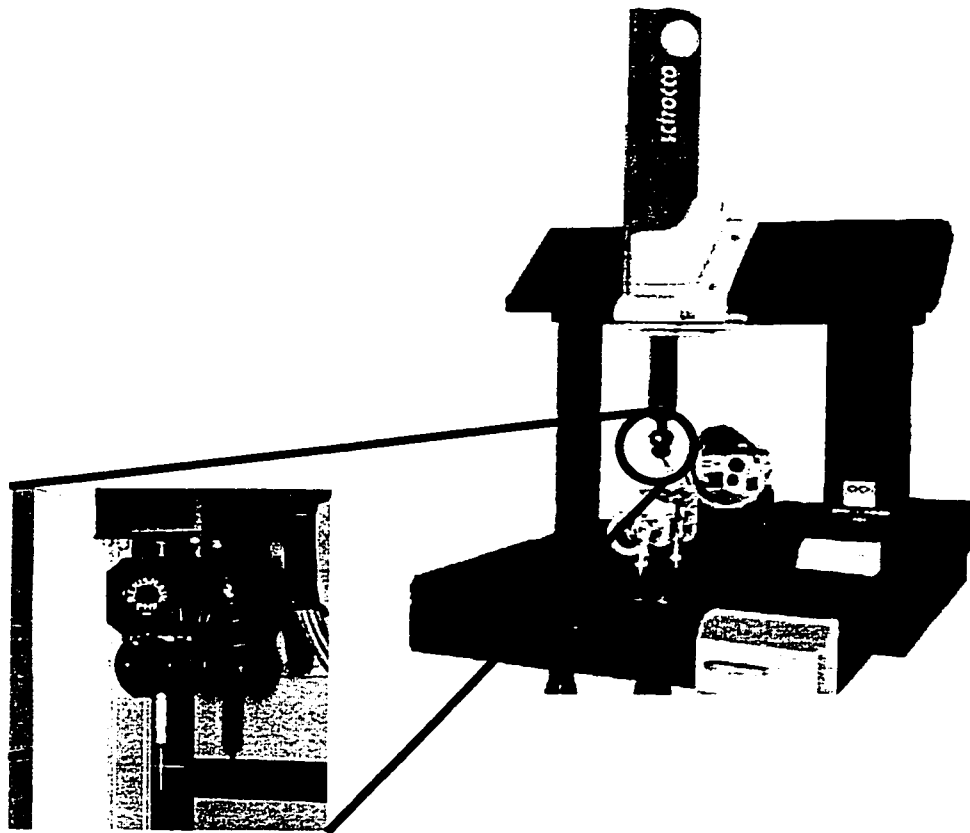
Figure 1.1. Part Inspection Using A CMM

1.1.1. CMM Main Components

The coordinate measuring machine is basically a Cartesian robot that is produced in various configurations. The CMM configuration plays an important role in meeting measurement requirements such as accuracy, flexibility, time (as synonymous of throughput), and cost (Slocum, A. H., 1992). The main CMM components fall in two categories, i.e., hardware components and application software. Hardware components include all the physical components that are responsible for the transmission, sensing, and control of the CMM motion. The main CMM hardware components are:

1. Mechanical setup with the three axes, servomotors, motion transmission components, and distance transducers.

2. Probe head to touch the workpiece at the measurement points.

3. Control unit, and control pad to control the motion of the CMM moving parts.

4. Digital computer with peripheral equipment and software to calculate and represent results.

The function of CMM application software is to facilitate the measuring process and provide users with measurement data in the most useful format. The main types of CMM application software are (Krejci 1995):

4

1. CMM core software:

This is the minimum software needed for the machine to run. It performs the tasks of calibration, coordinate system manipulation, feature construction, and some basic tolerance analysis.

2. Programming software:

This is the software used to facilitate the job of the CMM operator. This category of software improves the operator interface, and assists in the programming of CMM. It is subdivided into two broad categories, namely, online, offline programming software.

3. Post inspection software:

This category deals with the manipulation and presentation of the inspection data collected by the CMM. It performs such tasks as statistical analysis, data fitting and evaluation, and graphical representation of form errors.

4. Reverse engineering software:

This category of software is used to define the geometry of a measured part or prototype starting from the data digitized by the CMM.

Figure 1.2 illustrates the main CMM components.

```
                    ┌─────────────────┐
                    │    CMM MAIN     │
                    │   COMPONENTS    │
                    └────────┬────────┘
              ┌──────────────┴──────────────┐
     ┌────────┴────────┐           ┌─────────┴────────┐
     │    Hardware     │           │   Application    │
     │   Components    │           │    Software      │
     └────────┬────────┘           └─────────┬────────┘
   ┌──────┬───┴──┬───────┐       ┌──────┬─────┴────┬────────┐
┌──┴──────┐  ┌───┴────┐      ┌───┴─────┐      ┌────┴────────┐
│Structural│  │ Probe  │      │  Core   │      │ Programming │
│ Elements │  │  Head  │      │Software │      │  Software   │
└──────────┘  └────────┘      └─────────┘      └─────────────┘
┌──────────┐  ┌────────┐      ┌──────────────┐ ┌─────────────────────┐
│Control Unit│ │Computer│     │Post-Inspection│ │      Reverse        │
│Control Pad │ │and Peripherals││  Software    │ │Engineering Software │
└──────────┘  └────────┘      └──────────────┘ └─────────────────────┘
```
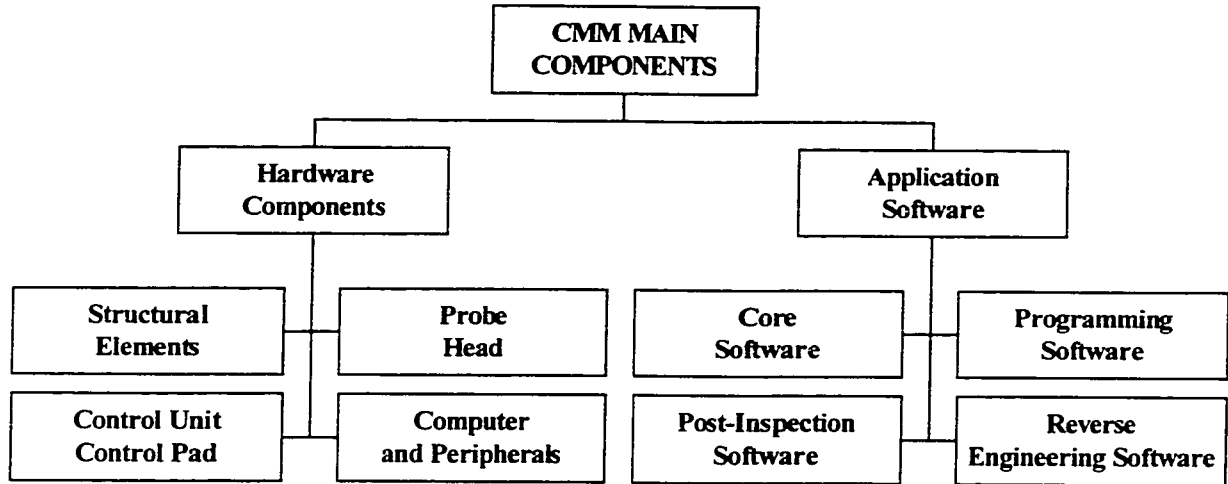
Figure 1.2. Main CMM Components

Basic CMM configurations, illustrated in Figure 1.3, are:

1. Fixed table cantilever arm.

2. Moving bridge.

3. Moving ram horizontal arm.

4. Gantry-type.

1.1.2. Programming of CMM

The following techniques are used to drive the CMM probe head to measure workpiece surface:

1. Manual operation:

This takes place by controlling the machine probe head to touch the

6

workpiece at the specified points. The encoders, then, determine the locations of the probed points in the Cartesian space.

2. *Teaching mode:*

The part inspection steps are carried out through driving the CMM probe using the control pad. Meanwhile, the machine programming software records these steps on-line in an inspection program. Once the program recording is complete, the program can be recalled at any time to perform the part inspection steps. This is the most commonly used CMM programming method. However, it is time consuming, since the machine is to stay idle during the programming process. In an environment where large variety of parts are produced in medium size batches, this method creates a bottleneck in the production.

3. *Off-line programming:*

In this case, the CMM program is generated without the need for any physical involvement of the machine. The workpiece design data and the CMM planner experience are used to generate the inspection plan (sampling points, approach vectors, probe orientations, probing path, etc.) which is then translated into the machine language.

4. Automatic CMM planning and program generation:

The CMM inspection plan and program are automatically generated based on the CAD data and the coded CMM planner experience. Sophisticated calculations are included, e.g., inspection points allocation, points grouping and clustering, probe orientations planning, and collision-free optimum probe path planning.

Cantilever type   Horizontal type

Gantry type   Bridge type
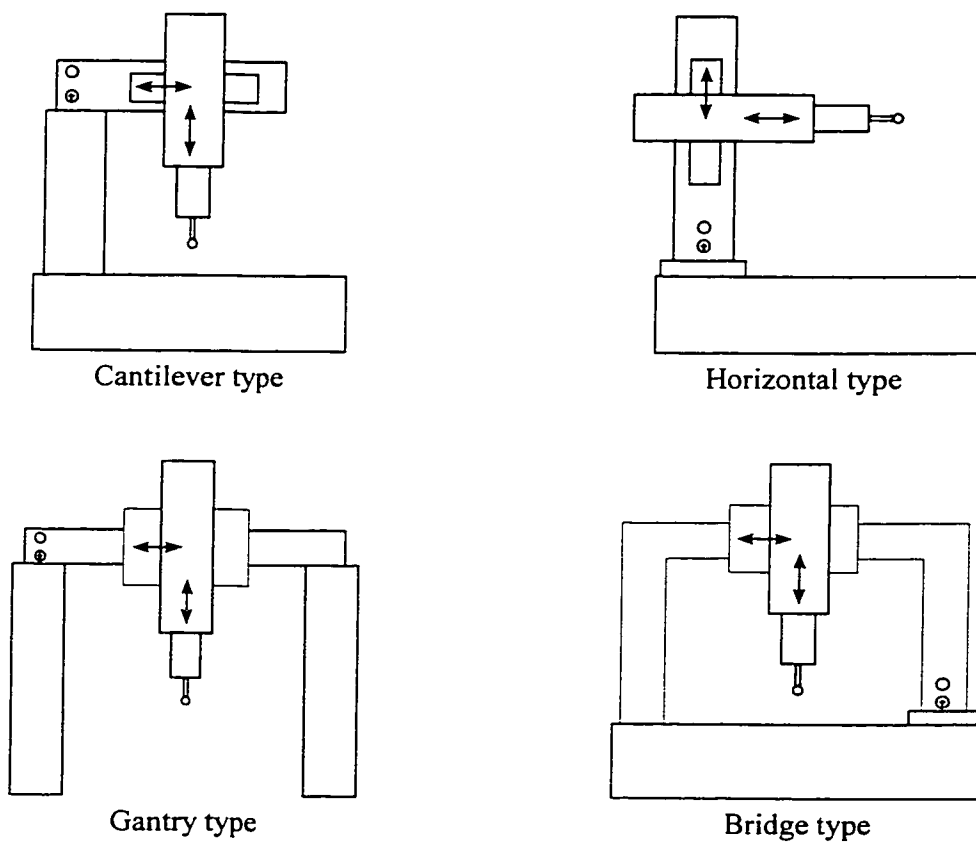
Figure 1.3. Various CMM Configurations

1.1.3. CMM measurement error and measurement uncertainty

CMM measurement error means the difference between the true coordinate location of the point probed with the CMM and the coordinate values measured by the CMM. In CMM performance evaluation, precisely calibrated artifacts are used. In this case, the form error of an artifact is negligible so that the measurement error can be calculated as the difference between the CMM measurement and the artifact dimension.

In the case of measuring well-calibrated artifacts, measuring the same artifact feature using the same procedures, and in the same environment more than one time still leads to different values, and hence different measurement errors. CMM measurement uncertainty can be defined as the collection of all possible measurement errors when measuring the same feature, following the same procedure, in the same environment. Measurement uncertainty is positioned about the measured value and is usually centred symmetrically, as illustrated in Figure 1.4 (Philips, 1995).
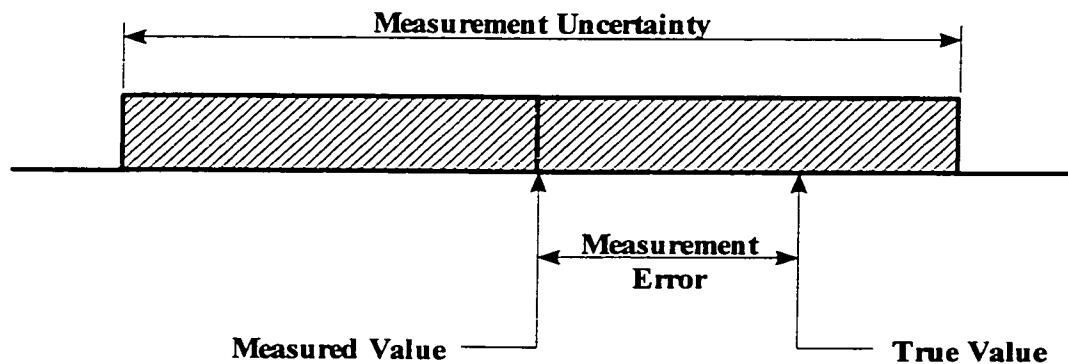
Figure 1.4: CMM Measurement Uncertainty And Measurement Error

Sources of uncertainty in CMM measurement are grouped in four main categories (Philips et al., 1993):

1. *Probe properties:*

Probe lobing, probe repeatability, stylus bending, indexable probe head repeatability, stylus changer repeatability, and probe spatial frequency response and frictional effects.

2. *CMM properties:*

Errors in rigid body geometry at standard temperature, non-rigid body geometry errors, CMM part loading effects, CMM dynamic behavior, CMM repeatability, algorithm accuracy, and thermally induced errors.

3. *Part properties:*

Part dynamics, part fixturing, and part thermal properties.

4. *User-selected properties:*

Algorithm selection, sampling strategy, and part location and orientation.

1.2. Overview Of Inspection Planning

Coordinate measuring machines (CMMs) are used to discretely sample points from a

10

manufactured part. Geometric features are then extracted from the sampled points to represent different characteristics of the produced part. The deviation between the produced part and the geometric model (ideal part) can then be obtained and evaluated.

In today's world of manufacturing, the CMM has become a critical device for quality control. The CMM has emerged as one of the systems of choice in measurement, and inspection. In order to meet the demands of very high production throughput, the inspection process using CMMs should be well defined and planned such that it is performed effectively and economically.

Inspection using CMM is the central activity in the inspection planning and verification procedure shown in Figure 1.5 (Limaiem, A. and ElMaraghy, H. 1997). The downstream activity is tolerance analysis and verification, which often takes place online. The upstream activity is the CMM inspection planning which usually takes place off-line.

Although the use of computer controlled CMMs eliminates costs associated with the manual inspection operations, it requires significant planning efforts for the CMM to function without human intervention. Feature-based CMM inspection verifies the conformity of the manufactured part with the design intent generally expressed in the form of dimensional and geometric tolerances, and decides whether to accept or reject it

accordingly (Limaiem, A. and ElMaraghy, H. 1999). Planning for CMM inspection involves a series of activities such as setup planning, operation planning, sample point determination, measurement sequencing, and probe trajectory planning, as illustrated by Figure 1.6 (Ziemian and Medeiros 1998).

The expertise of the planner and the CMM operator has to be complemented with mathematical tools which can handle efficiently and optimally critical and time consuming tasks in inspection planning for CMMs. The ever increasing need for automating the CMM inspection planning activities has emerged as CMMs started to find their place on the shop floors. Several attempts to automate inspection planning activities were reported in the literature. This section overviews those attempts.

The ever-increasing need for using CMMs on the shop floor, and difficulty and complexity of the CMM inspection planning process has urged researchers to work towards the automation of the CMM inspection planning activities. The need for CMMs can best be experienced in modern manufacturing environments with low production volumes, high variety production, close tolerances and high quality products. In such production environments, conventional dimensional inspection and verification methods can create production bottlenecks due to their limited speed, accuracy, and lack of integration with CAD tools. Therefore, more automated inspection process planning and

better decision support tools for human planners are needed. This can be achieved by

adopting and developing an integrated Computer-Aided Inspection Planning (CAIP)

approach as proposed by ElMaraghy H.A., and ElMaraghy, W. H. (1994).
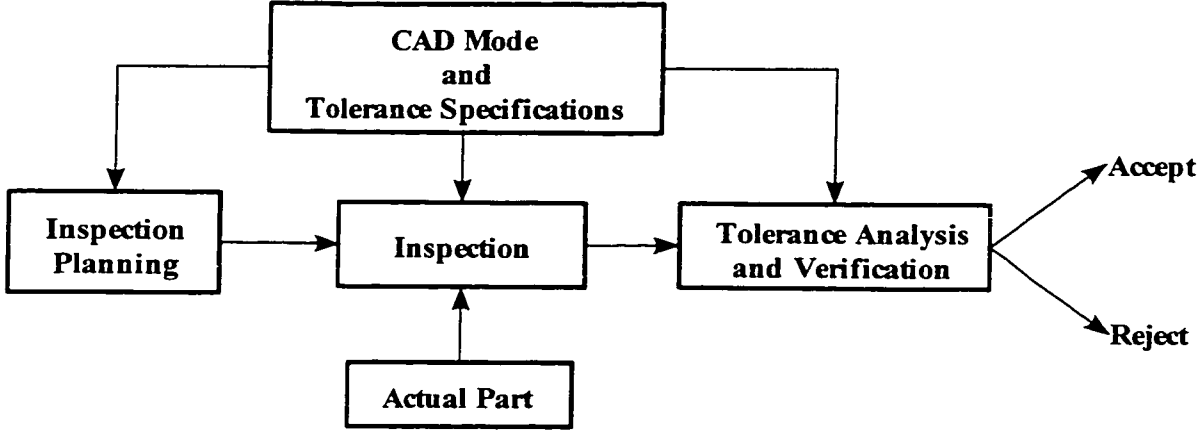
Figure 1.5: Inspection Planning And Verification Procedure
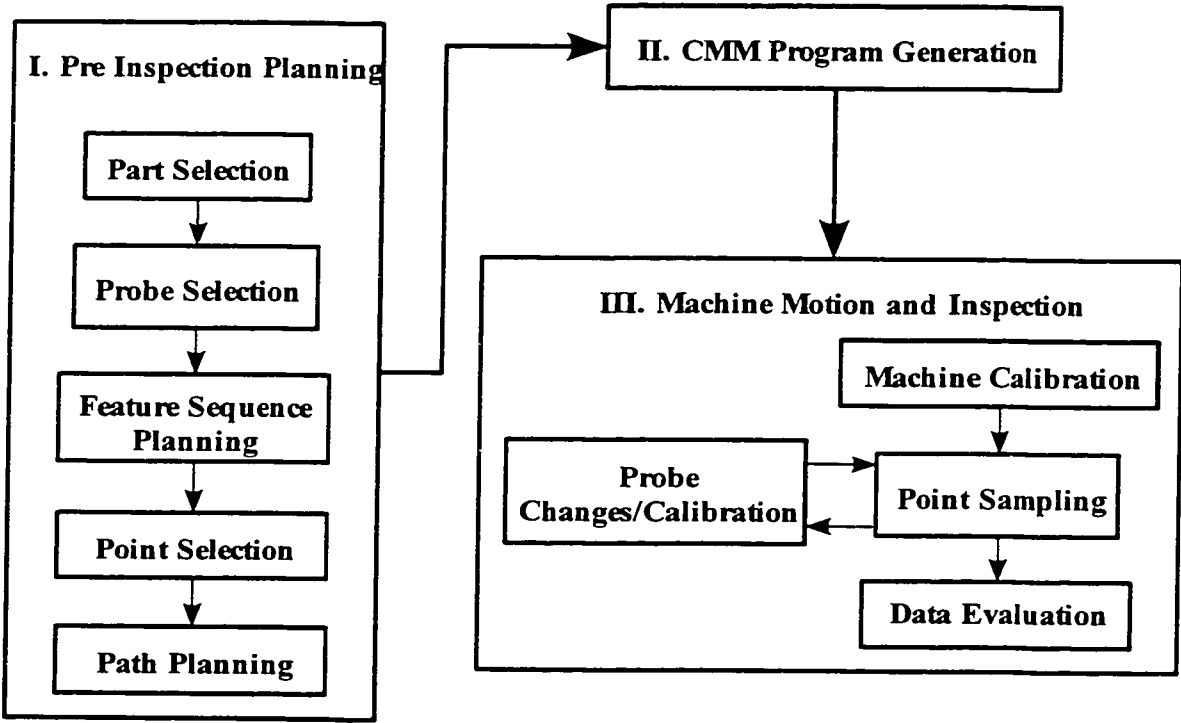
Figure 1.6: CMM Inspection Procedure

ElMaraghy and Gu (1987) presented the first expert system for inspection planning. In this system, inspection features are grouped, and prioritized. The expertise of human inspection planners has been transcribed into expert rules and used for clustering features to be inspected.

CMM inspection planning automation was approached through the automation of the low-level inspection planning tasks (Limaiem, A., 1996, Limaiem, A. and ElMaraghy, H., 1997 and 2000, and Vafeesefat, A., and ElMaraghy, H. A., 2000). Such tasks include: measurement points accessibility analysis, clustering of measurement points, sequencing of measurement points, generation of collision-free probe paths, and the integration of these tasks into a computer-aided inspection planning system. A Computer-Aided Tactile Inspection Planning (CATIP) system was developed (Limaiem, A., and ElMaraghy, H. A. 1999), which provides a means of integrating the previously mentioned inspection planning tasks into a computer-aided inspection planning system. This system provides tools for robust accessibility analysis, an optimization algorithm to provide different plans for clustering and sequencing the inspection points. It generates the shortest collision free probing path.

The automation of the high level inspection planning is approached through the automation of part setups determination. Spyridi and Requicha (1994) used a state space

approach to develop a system for automatic generation of inspection plans. The output of that system is a collection of setups, probes, and probe orientations associates with the surface features to be inspected.

Kweon and Medeiros (1998) presented a methodology for part setup planning based on the part visibility maps. Since a part frequently needs more than one setup for inspection, such a system presented a methodology to determine a set of part orientations such that all tolerances under consideration can be inspected with a small number of part orientations.

Ziemian and Medeiros (1998) presented a probe and workpiece setup planner for CMMs that defines a set of feasible probe orientations for the inspection of each feature requiring measurement and defining the associated recommended part setups.

Other researchers focused on developing efficient sampling plans for CMM inspection planning. The problem involves two parts, namely, the determination of the number of measurement samples that best represent the geometric feature, and the best distribution of this number of sample points. The later is the focus of the present work and will be discussed in details in Chapter 2 (Literature Survey).

1.3. Geometric Modeling Of Free Form Surfaces

The data probed with a CMM is manipulated to construct the substitute geometry, which is to be evaluated against a geometric model. In the case of reverse engineering, the data is fitted to produce a geometric model. In CNC machining, a mathematical representation of a part is essential to program a path for the cutting tool. For automatic CMM inspection planning, a part geometric model is used to plan the probe path, and probe orientations.

1.3.1. Parametric and non-parametric representation

There are two methods that are most common in representing curves and surfaces. These are implicit equations and parametric functions (Piegl, L., and Tiller, W., 1997). The implicit equations describe the relationship between the coordinates of a point lying on the geometric entity. They take the form $y = f(x)$ or $f(x, y) = 0$ for a planar curve, or $z = f(x, y)$ or $f(x,y,z) = 0$ for a 3D surface.

In parametric functions, each of the coordinates of a point on the geometric entity is expressed as a function in a parametric space (u for a point on a curve, or u-v for a point on a surface), while each of u and v varies between minimum and maximum real positive values. For each value of u and v, the position of a point on the geometric entity is calculated. These parametric functions take the form: $P = [x \ y]^T = [x(u) \ y(u)]^T$ for a point

16

on a parametric planar curve, and $\mathbf{P} = [x \; y \; z]^T = [\; x(u, v) \; y(u, v) \; z(u, v)]^T$ for a point on

a 3D surface. A circle can be implicitly represented as $f(x, y) = x^2 + y^2 - 1 = 0$, or

parametrically as: $C(u)=[x(u) \; y(u)]^T = [\cos(u) \; \sin(u)]^T$, $-p/2 \leq u \leq p/2$.

An implicit sphere is expressed as $f(x, y, z) = x^2 + y^2 + z^2 - 1 = 0$. A parametric sphere is:

$S(u,v) = [x(u, v) \; y(u, v) \; z(u, v)]^T = [\sin(u) \cos(v) \quad \sin(u) \sin(v) \quad \cos(u)]^T$.

where $-p/2 \leq u \leq p/2$ and $-p/2 \leq v \leq p/2$.

Parametric functions have several advantages over implicit equations that made

parametric functions the mathematical representation method of choice for CAD

packages (Piegl, L., and Tiller, W., 1997).

Geometric entities are not only cylinders, cones, spheres, and planar surfaces. Other

more complex shapes are the so-called sculptured or free-form surfaces which arise

extensively in such industries as aerospace, automotive, and die and mold making. A

sculptured surface is defined as a collection or sum of interconnected and bounded

parametric patches together with blending and interpolation formulas (Zeid 1991).

Free-form shapes have to be geometrically represented to be handled in the CAD,

CAM, and inspection applications packages. Parametric functions provide an efficient

and powerful representation tool. Several parametric representation methods are available to represent free form shapes, e.g., Splines, Bézier, B-Splines, and NURBS. Both B-Splines and NURBS are widely used in CAD packages for their flexibility and power in representing, and manipulating complex shapes.

1.3.2. NURBS review

NURBS is a parametric representation of a curve or a surface. A curve is defined by an approximate polygon (control polygon), the corner points of which are known as the control points (see Figure 1.7). Each control point "P_i" is associated with a positive real number, known as the control point weight "w_i", which pulls the curve towards, or pushes it away from the control point. The NURBS equation is a function of an independent parameter "u" which ranges from zero to one.

At a parameter value of zero, the first control point lies on the curve. At a parameter value of one, lies the last control point. For NURBS surfaces, the surface equation is a function of two independent parameters "u" and "v"; both range from zero to one. The surface passes through the border control points (u=0, v=0; u=0, v=1; u=1, v=0; u=1, v=1). Figure 1.8 illustrates the control polyhedron, and the data points of a NURBS surface.
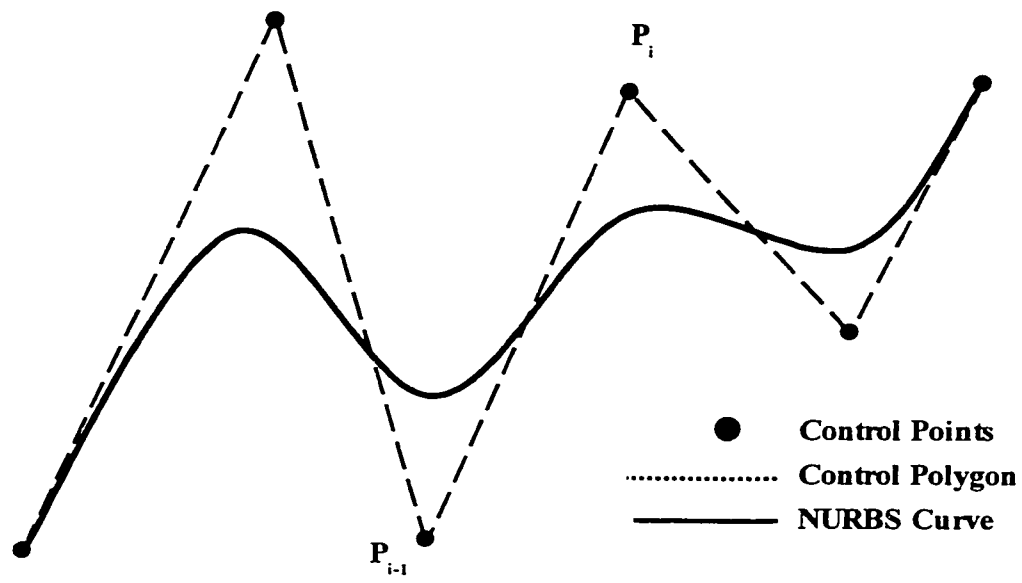
Figure 1.7: A Sample NURBS Curve
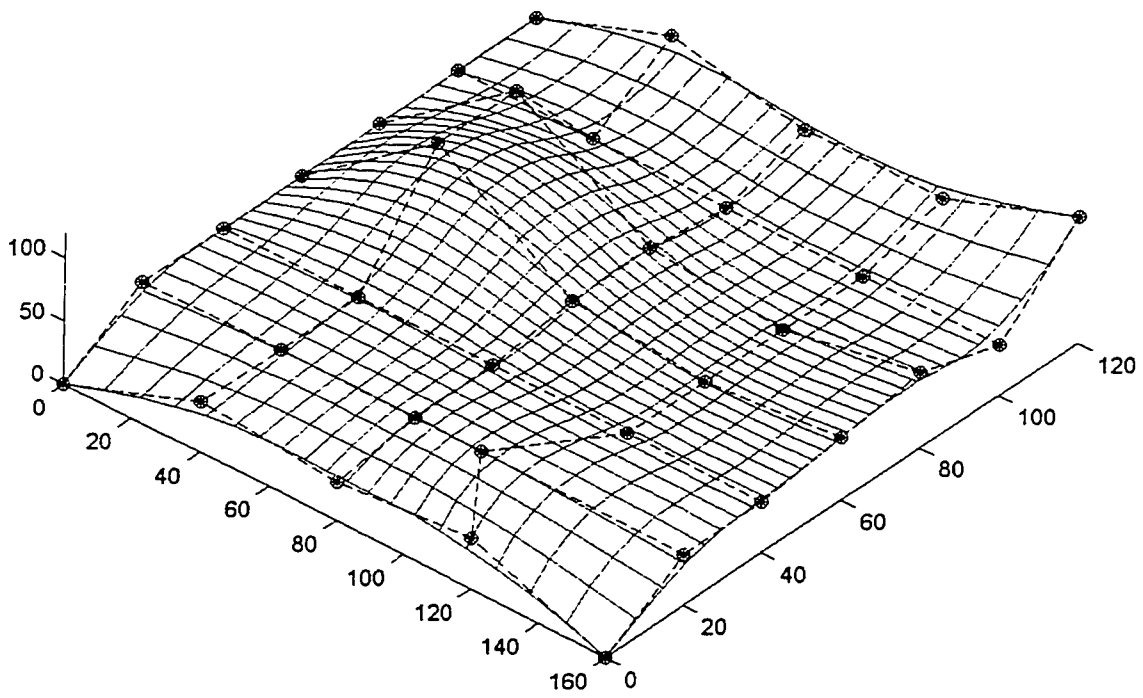
Figure 1.8: A Sample NURBS Surface

19

The major advantage of NURBS representation of curves and surfaces is the capability of introducing local changes to the curve/surface through changing the locations and/or weights of the control points. This local control comes from the segmentation of the curve/surface into spans/patches through subdividing the independent parameter(s) range of 0-1 into spans. Each span is controlled by a group of control points. This segmentation is achieved through the use of a non-decreasing real numbered sequence known as the knot vector. For a NURBS curve, only one knot vector is defined, whereas for a NURBS surface, two knot vectors are used to divide the surface into patches. Let $U = \left[u_1^k,\ u_2^k,\ \cdots\ u_i^k \cdots,\ u_m^k \right]$ be the knot vector of the independent parameter "u". A NURBS curve is defined by accompanying each of its control points "P_i" by a corresponding B-Spline function "N_i", defined recursively as:

$$N_{i,p}(u) = \frac{u - u_i^k}{u_{i+p}^k - u_i^k} N_{i,p-1}(u) + \frac{u_{i+p+1}^k - u}{u_{i+p+1}^k - u_{i+1}^k} N_{i+1,p-1}(u)$$

$$N_{1,0} = \begin{cases} 1 & \text{if } u_i^k \leq u \leq u_{i+1}^k \\ 0 & \text{otherwise} \end{cases} \qquad (1.1)$$

The NURBS equation of a curve is given by:

$$C(u) = \sum_{i=0}^{n} R_{i,p}(u) P_i$$

$$R_{i,p}(u) = \frac{w_i\, N_{i,p}(u)}{\sum_{i=0}^{n} w_i\, N_{i,p}(u)} \qquad (1.2)$$

20

where:

n = number of control points+1.

$R_{i,p}(u)$ = Rational basis function.

The rational basis function is non-zero in the interval (u_i^k , u_{i+p+1}^k], and thus, the change in any of the control point P$_i$ parameters (location, weight) will affect the segment of the curve where $u \in \{u_i^k , u_{i+p+1}^k\}$. This property is known as "local shape control" which gave NURBS an advantage over other mathematical representations of curves and surfaces. The same principles are applied for a NURBS surface, which is represented in terms of two independent parameters u, and v. The segmentation of a NURBS surface into patches is achieved through the use of two knot vectors, U and V, given by

$U = \begin{bmatrix} u_{1,}^k & u_{2,}^k & \cdots & u_{i,}^k \cdots, & u_m^k \end{bmatrix}$,and $V = \begin{bmatrix} v_{1,}^k & v_{2,}^k & \cdots & v_{i,}^k \cdots, & v_m^k \end{bmatrix}$ respectively. A NURBS surface of degree "p" in the "u" direction and degree "q" in the "v" direction is given by:

$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} R_{i,j}(u,v) \ P_{i,j}$$

$$R_{i,j}(u,v) = \frac{N_{i,p}(u) \ N_{j,q}(v) \ w_{i,j}}{\sum_{k=0}^{n} \sum_{l=0}^{m} N_{k,p}(u) \ N_{l,q}(v) \ w_{k,l}} \tag{1.3}$$

where:

n = number of control points+1 in the u-direction.

m = number of control points+1 in the v-direction.

$R_{i,j}(u,v)$ = Rational basis function.

NURBS provides a flexible and powerful tool for designers to represent a wide variety

of shapes, from the very simple curves/planes to even very complex free-form

curves/surfaces, as illustrated in Figure 1.9. Down stream applications of the NURBS

representation of surfaces are reverse engineering, and inspection. In reverse engineering,

a designer starts with a prototype, digitizes points on that prototype, and ends up with a

geometric CAD model that represents the digitized surface. An inspection person starts

with a product, digitizes points on the product surface, fits these points to a NURBS

surface that simulates the CAD model, and compares the fitted surface to the CAD model

to determine whether the produced surface is within the designated tolerance zone. As

previously mentioned, the number and locations of sampled points have a significant

effect on the shape of the surface fitted to them, and, therefore, on the inspection results.

1.4. Overview Of Optimization

Optimization is the application of mathematical and numerical methods for finding

and identifying the best candidate from a collection of alternatives without having to

explicitly enumerate and evaluate all possible alternatives (Reklaitis et al., 1983). In order

to implement any mathematical technique to achieve the optimal state for a certain

system, four main elements should be defined:

1. The system boundaries:

These are the limits that separate the system under consideration from the surrounding universe. For instance, to optimize the parameter values of points to be sampled from a free form surface, the limits of the system are the minimum and maximum values of the parameters u and v.

2. The performance measure:

This is the criterion based on which the performance of the system can be evaluated. It is usually a cost or profit function. For example, the objective may be to minimize the surface roughness, or to maximize the system efficiency.

3. The independent variables:

The independent system variables are those whose interaction characterizes the system behaviour. In machining for instance, the cutting speed, feed rate, and depth of cut are independent variables that affect the surface finish. Similarly, in inspection the locations of the sample points in the parametric space are independent variables that affect the accuracy of the free form surface fitted to these points.

4. The system model:

This is the formulation of the behavior of the system as a consequence of the interaction between different values of the independent variables.

23
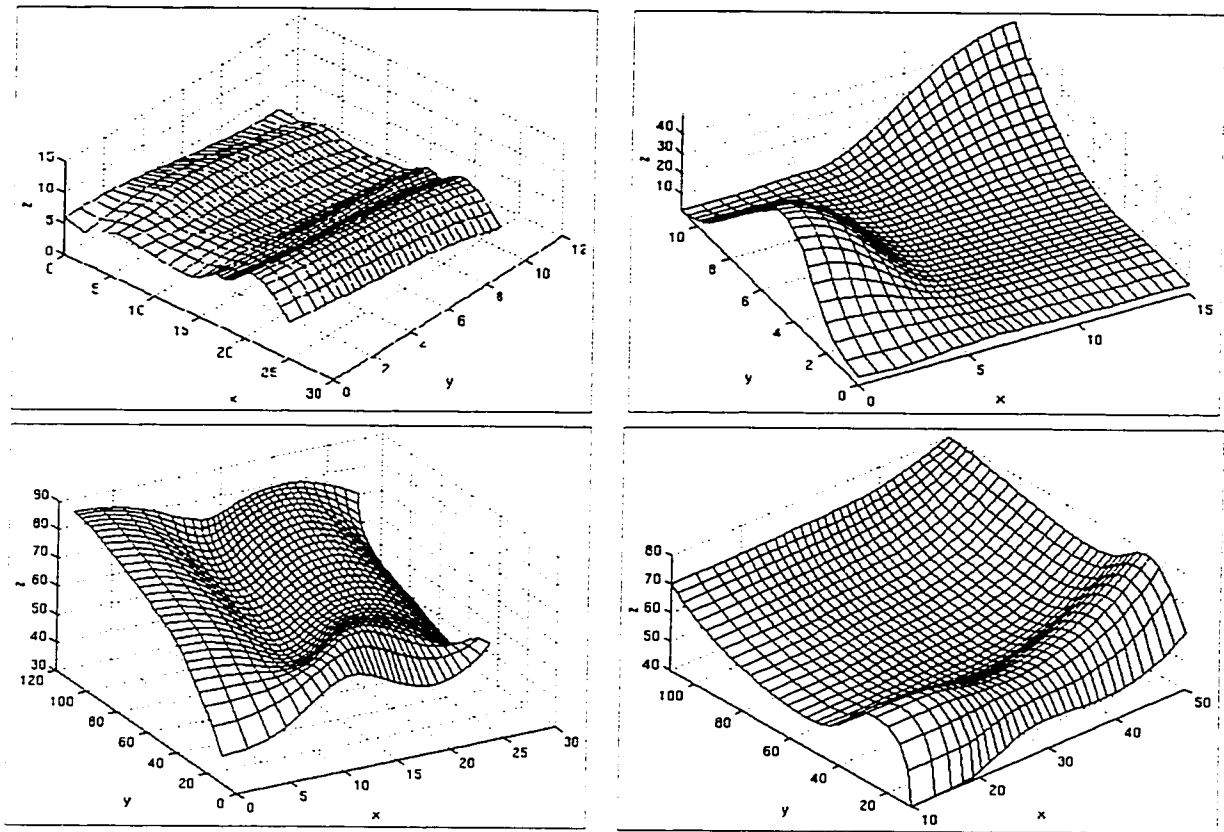
Figure 1.9. Samples Of NURBS Surfaces

Optimization is either constrained or unconstrained (Chong, E., and Zak, S., 1996). In unconstrained optimization, the problem takes the form:

Minimize $f(x)$, subject to: $x \in \Omega$ (1.4)

where:

$f(x)$ = The objective function.

$x = [x_1, x_2, x_3, \cdots x_n]$ = A set of real numbers defining the independent variables of the system.

Ω = The constraint set.

24

In constrained optimization, the problem takes the form:

Minimize $f(x)$, subject to
$$h_i(x) = 0 \ ; \quad i = 1, 2, \cdots, m$$
$$g_j(x) \le 0 \ ; \quad j = 1, 2, \cdots, p$$
(1.5)

where:

$$x \in R^n , \quad f : R^n \rightarrow R , \quad g_j : R^n \rightarrow R \tag{1.6}$$

Different optimization problems require different optimization methods (Reklaitis et al., 1983 and Chong, E., and Zak, S., 1993). Several techniques work on solving both constrained and unconstrained optimization.

1.4.1. Genetic Algorithms (GA's) optimization

Genetic Algorithms (GAs) are stochastic search techniques based on the mechanism of natural selection and natural genetics (Gen, M., and Cheng, R., 1997). GAs start with an initial set of random solutions for the problem under consideration. This set of solutions is known as the population. The individuals of the population are called chromosomes. The chromosomes of the population are evaluated according to a pre-defined fitness function. The chromosomes evolve through successive iterations called generations. During each generation, a new population is created by either merging, or modifying chromosomes of the present population. Merging chromosomes is known as "crossover", while modifying an existing one is known as "mutation". The selection of chromosomes

to merge/modify to create the new population is based on their fitness function. Once a new generation is created, a new population is formed by deleting members of the present population to make room for the new generation. The process is iterative. A step-by-step GA process proceeds as follows (Karr, C. L., and Freeman, L. M., 1999):

1. Generate random initial population of candidate solutions in the form of chromosomes.

2. Evaluate each chromosome in the population according to a pre-defined fitness function (usually, the value of the objective function corresponding to this chromosome).

3. Employ a selection operator to create new chromosomes. The selection operator biases the new generation of chromosomes towards higher quality "fitter" solutions.

4. Members of the present population are deleted to make room for the new candidates.

5. New chromosomes are evaluated and inserted into the population so that the population size remains constant.

6. If a satisfactory solution (or a termination condition) has been achieved, the process stops; otherwise, process is repeated from step 3.

Figure 1.10 (Chong, E., and Zak, S., 1996) shows a flow chart of the process described

above. Within the flow chart, P(k) represents the kth population, M(k) is the kth mating

pool of chromosomes, Pc is the probability of choosing two chromosomes for crossover

operation and Pm is the mutation probability within a chosen chromosome.
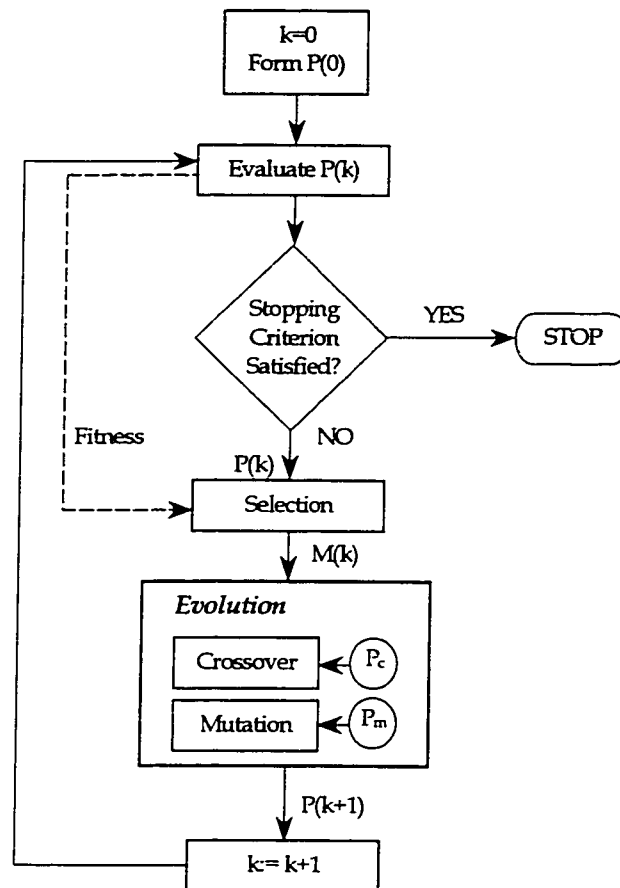
Figure 1.10. Flow Chart For Genetic Algorithms Optimization

Genetic algorithms are applied to a wide variety of engineering problems, e.g., design,

process planning, job-shop scheduling, robotics, software testing (Karr, C. L., and

Freeman, L. M., 1999 and Gen, M., and Cheng, R., 1997).

1.5. Problem Definition

The sampling strategy affects the accuracy of the inspection process results. Features inspection is usually carried out using a machine measurement system that contains systematic and pseudo-random errors. Furthermore, the actual form of the manufactured feature deviates from the ideal defined by CAD model giving rise to form errors.

Mathematical algorithms are used to fit the measurement data to construct the substitute geometry, which should resemble the real geometric feature. It may be intuitive that increasing the number of sample points decreases the error in measurement. However, the time spent on inspection is an important factor that affects the inspection cost, and hence, the product cost. Therefore, it is desirable to perform the inspection task satisfactorily with the lowest possible number of sample points (sample size). Second, it is of great importance to select the sample points at suitable locations to best represent the feature being inspected, and reduce errors due to the sampling strategy. The more complex the geometric feature, the more difficult to solve is the sampling problem. The problem is further complicated when free form shapes are encountered.

Many attempts to tackle the issue of sampling of free form features have been reported in the literature. Many of these attempts adopt methodologies that offer a single solution to the sampling problem (e.g. Uniform pattern sampling, Bezier patches curvature based

sampling, etc.). Besides, these methodologies do not seek optimal solution to the sampling problem. Furthermore, they do not provide an automatic methodology to sample free form surfaces.

The objective of this research is to develop a system for the sampling of free form surfaces. The characteristics sought in this system are:

1. NURBS-based representation of free form surfaces.

2. Providing alternate sampling strategies.

3. Offering an optimality criterion.

4. Offering automatic selection of the sampling methodology, depending on the surface complexity.

5. Ease and flexibility of integration with computer-aided tactile inspection planning systems.

Sampling algorithms utilize the features of NURBS representation of free form surfaces to allocate points on the surface. The sampling algorithms presented in this thesis are:

1. Equi-parametric sampling:

This algorithm equally distributes the sampling points on the surface in the

parametric space, regardless of the complexity of surface features.

2. Patch size based sampling:

This algorithm resolves the surface into its patches, ranks the patches according to their sizes, and then use this ranking as a criterion to distribute sample points, i.e., larger patches get larger share of sample points.

3. Patch mean Gaussian curvature based sampling:

This algorithm resolves the surface into its patches, ranks them according to the values of their mean Gaussian curvature, and use this ranking as a criterion for sampling.

4. Hybrid sampling:

This algorithm combines the latter two algorithms into one. Each of the former two methodologies is assigned a weight, and then the hybrid algorithm performs the sampling.

5. Genetic Algoritms (GA's) based sampling:

This algorithm uses Genetic Algorithms to optimize the location of sample points on the feature being inspected. The optimization criterion is the minimization of the maximum deviation between the substitute geometry, and

the CAD surface.

6. Automatic Selection of sampling algorithm:

This algorithm performs a series of complexity checks on the surface, and ,
accordingly, selects a sampling algorithm that is suitable to the surface under
consideration.

Each sampling algorithm was developed, implemented, and tested for NURBS curves,
and then generalized for NURBS surfaces. Furthermore, all the developed algorithms
were integrated into a computer aided system for sampling of free form surfaces.

1.6. Overview Of The Thesis

This thesis is divided into six chapters, and one appendix.

Chapter one presented the background to this research, as well as a problem definition,
and overview.

Chapter two provides a survey of the literature in the field of sampling for CMM
inspection planning. It also presents the objective, and motivation of the research.

Chapter three presents the sampling algorithms developed in this research. Flow

diagram for each sampling algorithm is provided, as well as the result on a NURBS curve, and a NURBS surface.

Chapter four presents a simulation study conducted to validate the efficiency of the developed sampling algorithms, and compare them to the uniform sampling pattern presented by Yao and Menq (1992).

Chapter five presents a computer-aided system for sampling of free form surfaces (CASampler) that integrates the developed sampling algorithms.

Chapter six concludes the thesis, and discusses the achievements, and future research, and development issues.

Appendix A of this thesis provides the pseudo code for the sampling algorithms developed throughout the course of this study, illustrated in chapter three.

Appendix B is a guide for the reader to setup, and run the Computer-Aided Free Form Sampling System (CASampler), illustrated in chapter five.

CHAPTER TWO

LITERATURE SURVEY

Many attempts to achieve the goal of automating the CMM inspection planning have been reported in the literature, as discussed in chapter 1. Some researchers presented methods to automate the sampling phase of the inspection planning for CMMs which involve prismatic and conical parts. Others attempted to automate the sampling of free form surfaces, applied algorithms to calculate the sample size, and automated the distribution of sampling points with a pre-determined number (sample size). Within this chapter, those attempts are discussed and compared. The type of part feature, the sampling criteria, and the optimality criteria, if any, are discussed. This chapter also contains an overview of sampling for CMM inspection planning. A summary table is provided to summarize the discussed literature.

2.1. Overview of Sampling for CMM Inspection

As discussed in the previous chapter, CMM acquires data on a point-by-point basis. These data need to be analyzed to create a geometric model, usually called the substitute geometry, for the feature being measured. The computed results are greatly affected by

33

different factors. Figure 2.1 demonstrates those effects (Philips 1995).
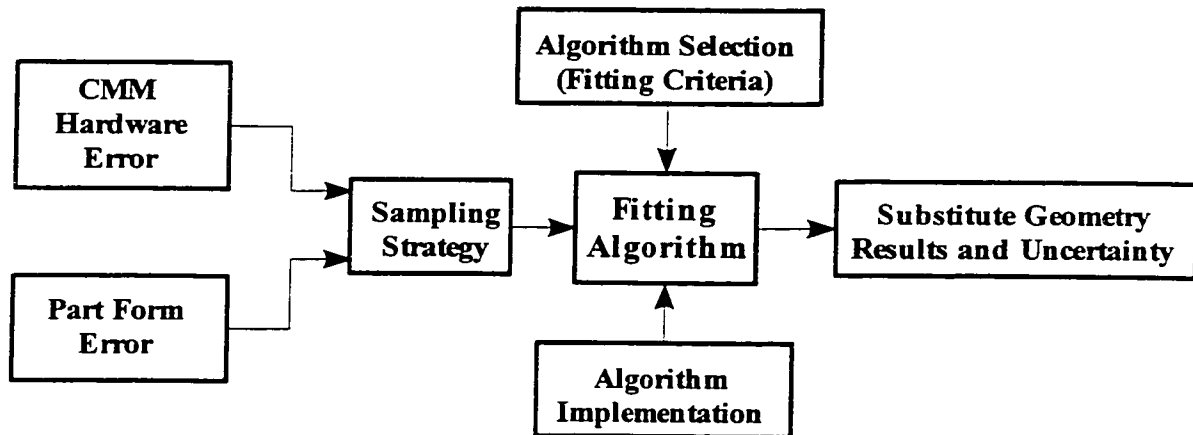
Fig. 2.1: Various Factors Affecting CMM Measurement

The effect of selecting a measurement sampling strategy has been recognized as a major component of measurement uncertainty (Philips et al., 1993). This effect comes from the fact that measuring a feature is usually carried out using a machine measurement system containing systematic and pseudo-random error (Hocken et al., 1993). Furthermore, the form of the feature deviates from ideal. This deviation, usually known as the form error, is always unknown. Mathematical algorithms will be used to fit these measurement data to construct the substitute geometry, which is supposed to resemble the real geometric feature. The more complex the features to be inspected, the more the need for a higher number of sample points. However, this increases the inspection time, and hence, the cost of the product. It is desirable to produce efficient sampling plans that utilize the lowest possible number of sample points. Second, it is of great importance to

34

locate the sample points in suitable locations so as to represent the feature being inspected, and reduce the error component due to the sampling strategy. Extensive research on sampling for CMM inspection planning has been reported in the literature. The more complex the geometric feature, the more difficult to solve is the sampling problem. The problem becomes very difficult when free form shapes are concerned. It is needed to fit the data sampled to a free form surface that best represents the real features such as surface complexion and continuity.

Three sampling schema are most commonly used in the literature, i.e., uniform or equidistant sampling, completely randomized sampling, and stratified sampling (Dowling et al., 1997 and Hocken et al., 1993). When the geometric feature is a free form feature, the need arises for sampling methodologies that take into consideration the complexity of that feature.

2.2. The Work of Menq and His Team

2.2.1. Menq et al. (1990), and Yau and Menq (1992)

Menq et al. (1990), and Yau and Menq (1992) presented a statistical approach for the evaluation of geometric tolerances using discrete CMM measurement data. The system presented involves three phases, namely, the generation of the sampling plan, optimal match, and comparative analysis. In this system, statistical formulation was

applied to solve the problem of determining the sample size. The formulation depends on the manufacturing process capability, and the profile tolerance assigned to the surface.

Introduced in the formula for sample size determination are two factors "d", and "a" that represent the probability of types "I", and "II" errors, i.e., the acceptance of bad parts or the rejection of good parts. After determining the sample size "N", the samples are distributed on the geometric feature to be inspected. In the case of free form surfaces, the "N" samples are uniformly distributed. The optimal match of the profile/feature is then applied to the sampled data. This is the stage of fitting the sampled data to obtain the substitute geometry. Comparative analysis is conducted to determine whether the substitute geometry satisfies the tolerance specifications.

2.2.2. Wong et al. (1991), and Menq et al. (1992)

Wong et al. (1991), and Menq et al. (1992) presented a computer integrated. dimensional inspection system for objects having sculptured surfaces. They presented a system for CAD/CMM integration, CAD/CMM inspection planning and comparative analysis.

The CAD/CMM inspection planning module is the phase responsible for generating the inspection plan. Various sampling methods were examined for free form curves (Wong et al 1991). The curve sampling approaches involved methods such as equal-

length spacing, fixed-length spacing, curvature-gradient spacing, and equal-parametric spacing. For free form surfaces, the sampling methods applied are the equal parameter spacing, equal length spacing, and fixed length spacing. For each inspection point, the probing vector is calculated. The generated measurement points are then fed to the CMM which probes them, and feeds them back to the system.

Comparative analysis of the inspected surface takes place to investigate whether the measured part satisfies the tolerance specifications in the CAD model.

The systems and approaches presented by Menq and his team take into consideration the effect of the manufacturing process on determining the sample size, and the tolerance specifications. They also provide sampling plans for free form surfaces. However, these systems still do not offer an optimal distribution of sample points, and do not take into consideration the free form surface complexity levels, i.e., curvature changes, parameterization, etc.

2.3. The Work of Hocken and His Team
2.3.1. Caskey et al. (1992)

Caskey et al. (1992) proposed and conducted a simulation study to examine the interaction between various procedures involved in measurement of mechanical parts using CMMs. The system presented in this study examines the interaction between the

37

sampling strategy, the systematic and random errors, and the algorithms used to analyze the coordinate data to produce mathematical descriptions of geometric features such as lines, planes, circles, spheres, cylinders, and cones. To achieve their goal, Caskey et al. proceeded as follows:

1. Perfect geometric entity was constructed.

2. Systematic and random error components were obtained from a data generator algorithm, and added to the perfect data.

3. Different sampling techniques were applied.

4. Fitting algorithms were used to fit the sampled data.

5. Fitted data was compared to perfect data.

In this research, Caskey et el. emphasized the great importance of sampling plans. They concluded that more research was needed in the area of sampling, especially the minimax algorithms. The also predicted that non contact probing will be demanded, for its ability to scan large amounts of data in short time spans.

2.3.2. Machireddy et al. (1993)

Machireddy et al. (1993) conducted a simulation study on the interaction between the sample size and the fitting algorithms applied for sampling of plane surfaces. They used the same principle used by Caskey et al. (1992) to generate the plane surface and

add the random and systematic error components. The stratified sampling technique was applied to sample data from the plane surface. The mini-max, and least square algorithms were applied to fit the sampled data to a plane surface. The fitted data was then compared to the data obtained from the plane surface generator. They used two criteria as accuracy measures, namely, the perpendicular distance normal to the true plane and the simulated plane, and the angle between the normal of the two planes. Results of their simulation experiments proved the advantage of the least square algorithm over the mini-max algorithm. They also concluded that a sample size of at least 49 points is needed for a planar surface to be sampled within an accuracy of 5 micro radians.

2.3.3. Uppliappan et al. (1997)

Uppliappan et al. (1997) conducted a similar study on the sampling of cylinders. They investigated the influence of the form and measurement errors on the sampling methods and substitute geometry algorithms. The sampling strategies applied in this study are the equi-distant sampling at multiple sections, and the spiral sampling. The fitting algorithms compared are the least square, and mini-max fitting algorithms. The results of their experiments revealed a general relationship between the form error, sampling strategy, measurement uncertainty, and the choice of the fitting algorithms. They concluded that for the sampling of cylinders, equi-distant sampling produces satisfactory results, when compared to the other sampling strategies. Generally, least

square fitting algorithm proved to be more stable than the mini-max algorithm. However, the results vary with the nature of the form error.

2.4. The Work of Woo and Liang

Woo and Liang (1993), presented a method to determine the sample size "N" and the sample locations for the inspection of machined planar surfaces. The sampling size calculation method presented is based on the discrepancy of the surface measurement, and the number of dimensions. Woo and Liang chose the Hammersley sequences to determine the sample locations. This method was applied to Gaussian and Weiner surfaces and compared against uniform sampling. The results of the implementation of this approach show a great advantage for the Hammersley sampling over uniform sampling. However, some limitations were detected including:

1) This method is applicable only to planar surfaces.

2) This method dramatically reduces the number of points needed to be sampled from a surface, but it does not offer an optimality criterion for the distribution of sample points.

2.5. The Work of Cho and Kim

Cho and Kim (1995) presented a CMM inspection planning strategy for free form

surfaces. Their strategy involves two major parts: the distribution of a predetermined sample grid over the sculptured surface, and the determination of the probe path to visit those inspection points. The strategy they followed to distribute sample points is based on the mean curvatures of the surface sub-regions and is outlined as follows:

1) Divide the surface into a predetermined number of regions.

2) Allocate a sample point to the centre of each region.

3) Divide each region into a grid of sub-regions.

4) Calculate the mean curvature per curve sub-region.

5) Rank the surface sub-regions according to their mean curvatures, and define factor "a" which defines the % of sample points to be assigned to the sub-region with the highest ranking.

6) Sample points are uniformly distributed within each sub-region.

The sampling strategy proposed by Cho and Kim ensured the representation of all the surface regions, with emphasis on surface regions with curvature changes. However, this method has the following limitations:

1) Assigning sample points at the centre of surface regions could lead to the need for a larger sample size, since it is likely to assign more sample points to regions where less points are needed, i.e., relatively flat surface regions. In other words, this technique equally treats regions with sharp curvature changes, and those whose

41

features are relatively flat.

2) The user-defined factor "a" imposes another limitation, since any change in this factor would lead to a significant change in the sampling plan. Moreover, it makes the system more dependent on the user's skills.

3) This system does not provide an optimal solution for the sampling planning problem.

4) To use this system as a part of an optimization objective function, two nested optimization loops would result; i.e., to optimize the regions/sub-regions grid sizes, and the value of the factor "a". This leads to a tedious, and lengthy process, that is difficult to implement in a system for inspection planning.

2.6. The Work of Zhang et al.

Zhang et al (1996)presented a method to optimally determine the sample size needed for the CMM inspection of drilled holes. They applied an Artificial Neural Networks (ANNs) approach to achieve this goal. Zhang et al. Considered the effects of the manufacturing process, the tolerance band, and the manufactured hole size on the sample size. Figure 2.2 shows the structure they suggested for their ANN. Back propagation learning was applied to teach the ANN to output the optimal sample size given the manufacturing process, the hole size, and the tolerance bad as inputs.

This approach has the following limitations:

1) It does not take into account the effect of the sample points locations on the

CMM measurement uncertainty (Phillips et al. 1993).

2) It is limited to specific hole manufacturing processes.

3) The ANN output does not guarantee the optimality of the sampling distribution.

Manufacturing
Process

Tolerance
Band

Hole
Diameter

Input Layers

Hidden
Layers

Output Layer

Sample Size
"N"
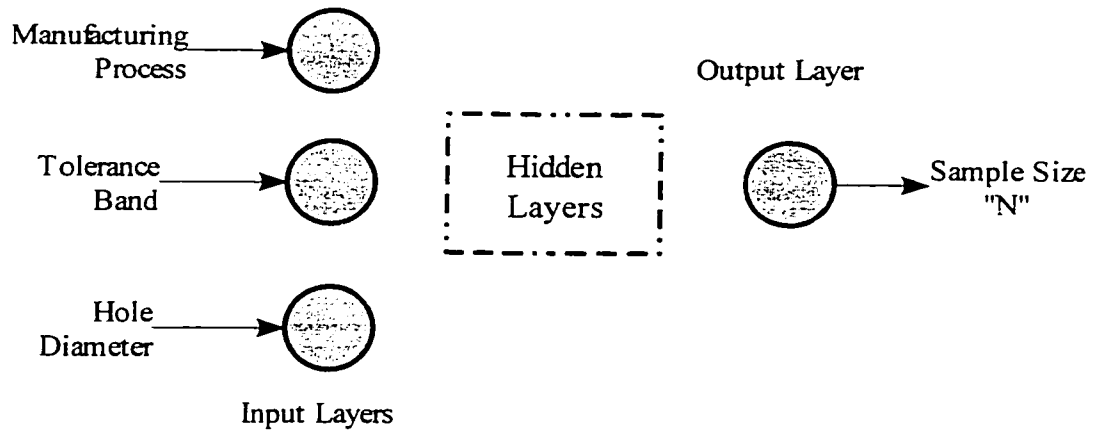
Fig. 2.2: ANN Architecture Proposed By Zhang Et Al.

2.7. Other Work

Other attempts to automate sampling for CMM inspection planning were reported in the literature for specific cases.

2.7.1. Pahk et al. (1995)

Pahk et al (1995) presented a system for the inspection of molds. Their system involved data sampling (distribution of predetermined sample grid), sampling path generation, and error evaluation. This system represented the mold surface

parametrically. Different sampling strategies were demonstrated, i.e., uniform sampling, curvature-based sampling, and a hybrid approach of both uniform and curvature-based sampling. Uniform sampling distributes the sample grid of points uniformly regardless the surface complexity. Curvature based sampling distributes sample points among the surface sections so that sections with high curvatures get larger shares of sample points. A hybrid approach combines both uniform and curvature based sampling.

2.7.2. Fan and Leu (1998)

Fan and Leu (1998) presented an AutoCAD-based system for the inspection planning of 3D objects. The sampling module of their system provides the user with two options, i.e., automatic, and user-defined sampling. The automatic sampling routine samples four points from a plane surface, and eight from a cylindrical feature. The distribution of the sample points is based on the feature dimensions, i.e., the plane width and height, and the cylinder length and diameter. The sampling method presented by Fan and Leu is limited in three ways:

1) The automatic sampling is dependent only on the feature dimensions. It does not take into account the tolerance requirements, the form errors, or the effect of the fitting algorithms.

2) This method does not provide alternate sampling methodologies.

3) No optimality criterion is applied.

44

2.7.3. Lee and Mou (1996)

Lee and Mou (1996) combined the Hammersley sampling with stratified sampling to different geometric features. They applied this combined sampling methodology to circular, hemispherical, and conical features. Their major result was the superiority of the stratified Hammersley over the stratified random and uniform sampling. However, this study did not offer an optimal solution for the distribution of sample points. Moreover, they did not address the effect of different factors on the choice of the sample size, i.e., process capability, and tolerance bands.

2.7.4. The work of Kim, and Ozsoy (1999)

Kim, and Ozsoy (1999) presented a technique to distribute the sample points for the form evaluation of free form surfaces. Their technique allocates the sample points on the constant curvature contour lines. In their method, they used uniformly parameterized cubic B-Spline surfaces. First, the maximum curvatures each point on the surface grid are computed. Second, the contour lines along which the curvatures are constant are generated. Third, for each contour, two values are computed, i.e., its length, and its rank (depending on the curvature value). Both parameters are used to weigh the contour line, and determine its share of the sample points. Fourth, the locations of the sample points are determined along the contour lines using linear interpolation.

Kim, and Ozsoy performed an experimental study to verify their technique, and concluded that it is effective, and better than such techniques as the uniform sampling pattern, the equi-parametric pattern, and the region selection method (Cho, and Kim, 1995). They also designed their technique to offer flexibility to its user to combine the two criteria used to weigh the constant curvature contour lines, and thus, tighten, or loosen the quality specifications.

The method presented by Kim, and Ozsoy provides a flexible tool for the inspection planner to allocate sample points on a free form surface. However, it does not take into account the curvature change as a factor that might affect the accuracy of the sampling plan, and therefore, resulting substitute geometry. Furthermore, their method does not take into consideration the effect of non-uniform parameterization of free form surfaces. Aside from the former two drawbacks, their method does not provide the user with an alternate method to optimize the sampling plan.

2.7.5. The work of Orady et al. (2000)

Orady at al (2000) presented a fuzzy-based decision making system for determining the CMM sampling strategies. They based their system on relating the machine tool accuracy to the part measurement accuracy. They composed their system of four main components: 1) knowledge bases, 2) fuzzy interface based decision making engine, 3)

online feed back adaptation module, and 4) feature fitting software.

The knowledge bases are collections of information, and rules related to design, manufacturing, metrology methods, and previous experiences. Based on the knowledge bases, the decision making module uses fuzzy rules to predict a CMM sampling strategy. The CMM carries out the sampling strategy, and returns the locations of the measured points. The sampled points are, then, fitted using the feature fitting software module. The feed back and adaptation module processes the online data according to the related data stored in the knowledge base. According to this analysis, the module commands the system to continue the measurement process, or adapts the system by updating the fuzzy rules, and the related data in the knowledge base.

This system was implemented, and tested on determining the sampling strategies for measuring the straightness of a produced part. The feature fitting software fits the measurement data to circles, lines, or cylinders.

2.7.6. The work of Fang et al. (2001)

Fang et al. (2001) presented a methodology for the sampling of spherical features. They proposed a linear model for the stratified sampling technique. They performed a theoretical analysis study to compare their method with the random sampling method.

The analysis proved the linear model proposed for stratified sampling to surpass random sampling scheme in the accuracy of the sampling plan.

2.7.7. The work of Edgeworth, and Wilhelm (1999).

Edgeworth, and Wilhelm (1999) presented an iterative sampling process for dimensional measurement of free form surfaces. The strategy proposed by Edgeworth, and Wilhelm uses the surface normal measurement data to develop an interpolating curve between sample points. The developed interpolant between every two points is evaluated to determine a new target point. The process is repeated until the measurement converges to a complete, and accurate evaluation of the surface.

2.7.8. The work of Kim, and Raman (2000)

Kim, and Raman (2000) investigated the sampling for the CMM measurement of flatness of planar surfaces. In their study, they investigated the Hammersely sampling, the Halton-Zaremba sampling, and the aligned systematic sampling. In addition, they proposed the systematic random sampling methodology. They performed a simulation study to compare the four sampling strategies. The criteria used were the accuracy of the sampling plan, and the length of the generated CMM probe path.

Table 2.1 summarizes the different attempts to automate sampling for CMM inspection planning.

48

Table 2.1. Summary Of Automated Sampling For Cmm Inspection

	Menq And His Team (1990-1992)	Woo And Liang (1993)	Hocken And His Team (1992-1997)	Cho And Kim (1995)
Type	Automatic.	Automatic.	Automatic.	Automatic.
CAD Model	Free form curves/surfaces.	Caussian and Weiner surfaces.	Lines, cylinders, and planes.	Free form surfaces.
Sample Size Calculation	Yes.	Yes.	No.	No.
Samples Distribution Criterion	Equal parameter, Equal length, Fixed length.	Hammersley.	Alternate (Uniform, stratified, etc.)	Mean curvature of surface sub-regions.
Alternate Plans	Yes.	No.	Yes.	No.
Optimization	No.	Sample size.	No.	No.

Table 2.1-continued.

	Zhang Et Al. (1996)	Pahk Et Al. (1995)	Fan And Leu (1998)	Lee And Mou (1996)
Type	ANN-Based.	Automatic.	Automatic.	Automatic.
CAD Model	Internal drilled holes.	Free form mould surfaces.	Plane surfaces and cylinders.	Circular, hemispherical, and conical.
Sample Size Calculation	Yes.	No.	No.	No.
Samples Distribution Criterion	Uniform.	Uniform, curvature-based, and hybrid	Based on feature dimensions.	Multiple.
Alternate Plans	No.	No.	Yes.	Yes.
Optimization	No.	Sample size.	No.	No.

49

2.8. Motivations And Objectives

Review of the previous work in automatic sampling for CMM inspection planning shows that many areas have been extensively researched in specific cases. For instance, sampling of prismatic and conical shapes has received a lot of attention from the researchers. However, many other problems remain unsolved. Among these problems are the optimization of sampling plans, and sampling of free form surfaces. The latter is the focus of this study.

From the literature summary given in Table 2.1, one can infer that most of the attempts to automate sampling for CMM inspection of free form surfaces adopt specific scenarios for sampling points distribution. For instance, some use curvature-based sampling, some use uniform sampling, others apply a hybrid approach of both scenarios. Most of these attempts do not provide alternate scenarios for sample points distribution. They also do not apply an optimality criterion for the sampling problem.

Motivated by the above discussion and the limitations of the reviewed literature, the objectives of this work are to:

1. Examine the effect of different sampling scenarios on the effectiveness of sampling of free form surfaces. This is to be carried out through the development

of a set of algorithms for the sampling of free form surfaces using different sampling scenarios. The sampling scenarios are to take into account the various features of parametric free form surfaces, i.e., the surface patch size, the surface curvature, and the surface parameterization.

2. Develop an algorithm to optimize the locations of the sample points for CMM inspection planning of free form surfaces, such that to minimize the deviation of the substitute geometry from the nominal CAD data. This is a desirable option that helps obtain efficient sampling plans that reduce the inspection cost.

3. Develop a methodology for automatic sampling of free form surfaces. This is to be accomplished through the implementation of the various algorithms for free form surfaces sampling in the form of computer programs. This methodology is to have the following characteristics:

    a) Integration of the various sampling algorithms into a computer-aided system for sampling of free form surfaces.

    b) The ability to provide alternate sampling scenarios.

    c) The ability to provide optimum sampling plans that minimize the deviation of the substitute geometry from the nominal CAD data.

    d) The ability to effectively handle the complex mathematics of free form

51

surfaces and rid the user of getting into deep details of the construction of free form surfaces.

4. The developed sampling methodology is to be tested using various free form shapes. The tested shapes are to be designed to simulate various levels of surface shape complexity. The various sources of error are to be simulated, i.e., the CMM measurement error, and the surface form error. The interaction between different sampling scenarios and the surface complexity is to be investigated.

5. A final step is to compare the developed and proposed sampling methodology with similar methodologies reported in the literature.

Next section reviews the work published throughout the course of this study towards accomplishing the objectives mentioned above.

2.9. The Work of ElMaraghy and Team
2.9.1. ElKott, D., ElMaraghy, H., and Nassef, A. (1998)

ElKott, D., ElMaraghy, H., and Nassef, A. (1998) presented methodologies for the sampling of free form curves. The methodologies presented utilized the NURBS representation of curves to compute sampling plans that reduce the maximum deviation between the design curve, and the curve fitted to the sampled points. They also presented

a methodology for the optimal sampling of free form curves. The criterion for optimization is to minimize the maximum deviation between the design curve, and the curve fitted to the sample points. In this work, Genetic Algorithms (GA's) were applied to achieve an optimal solution to the sampling problem.

Extensive simulations were applied to test these methodologies. This work was presented as a candidate for further generalization to NURBS surfaces.

2.9.2. ElKott, D., ElMaraghy, H., and Nassef, A. (1999)

ElKott, D., ElMaraghy, H., and Nassef, A. (1999) extended the work presented in the previous subsection to NURBS surfaces. They presented four methodologies for the sampling of free form surfaces. Each methodology emphasized a certain feature of the NURBS surface features. The algorithms developed in this work are, namely, equi-parametric sampling, patch size based sampling, patch mean curvature sampling, and hybrid sampling. In addition to the mentioned heuristics, they introduced an algorithm for the optimization of sampling of free form surfaces using Genetic Algorithms (GA's).

The algorithms presented were implemented, and tested on NURBS surfaces with different levels of complexity.

2.9.3. ElKott, D., ElMaraghy, H., and ElMaraghy, W. (2000)

ElKott, D., ElMaraghy, H., and ElMaraghy, W. (2000) presented a system that integrates the algorithms developed in the latter two subsections into a computer-aided system for inspection planning (CASampler). This system was developed to:

1. provide the user with alternate solutions to the sampling problem,

2. provide an optimality criterion,

3. provide the user with a utility to automatically select the sampling algorithm, based on the surface complexity,

4. be easy to use, and user-friendly, and

5. be feasible to integrate with CATIP (Limaiem, A., and ElMaraghy, H., 1999).

The algorithms mentioned in this section are described in details in chapter three. The simulation study performed on the sampling algorithms is presented in chapter four. Details of CASampler are presented in chapter five.

2.10. Discussion

This chapter provided a survey of the literature available on the sampling of free form surfaces. It also outlined the work published throughout the course of this thesis. The work presented in sections 2.2 through 2.8 pointed out the problems in sampling for

CMM inspection planning. It is obvious that various problems related to sampling have been extensively covered in the literature (e.g. sampling of different primitives). However, more work needs to be done in the field of sampling of free form surfaces. Section 2.8 presented the objectives of the research presented in this thesis. Section 2.9 highlighted the work published throughout the course of this research. The next chapters discuss the details of this work.

CHAPTER THREE

SAMPLING ALGORITHMS DESCRIPTION

This chapter demonstrates the sampling algorithms developed throughout the course of this work. The implementation of Genetic Algorithms (GA's) in the sampling of free form surfaces is demonstrated as well. The parameter used to measure the effectiveness of sampling scenarios is the maximum deviation between the substitute geometry and the design data. MATLAB 5.3 (Math Works Inc.) was used to code the sampling algorithms. Before demonstrating the different sampling algorithms, some common topics should be highlighted, i.e., the methods used for free form surfaces representation within the algorithms, how sampling is simulated and the method used for data fitting.

3.1. Approach

3.1.1. Representation of free form surfaces

NURBS are chosen in this work to represent free form surfaces as was shown in chapter I. For simplicity of computation, a slight modification was introduced by using homogeneous coordinates (Piegl, L., and Tiller, W., 1997). A NURBS curve is represented in four-dimensional space as:

$$C^W(u) = \sum_{i=0}^{n} N_{i,p}(u) \; P_i^W \qquad\qquad (3.1)$$

where P_i^W is the ith weighted control point:

$$P_i^W = (\; w_i x_i \;,\; w_i \, y_i \;,\; w_i z_i \;,\; w_i \;) \qquad\qquad (3.2)$$

$C^W(u)$ is a tensor vector that represents a point on the curve, calculated as:

$$C^W(u) = (\; w x(u) \;,\; w \, y(u) \;,\; w z(u) \;,\; w \;) \qquad\qquad (3.3)$$

Similarly, a NURBS surface is represented in the same fashion as:

$$S^W(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,p}(u) \; N_{j,q}(v) \; P_{i,j}^W \qquad\qquad (3.4)$$

where:

$$P_{i,j}^W = (\; w_{i,j} x_{i,j} \;,\; w_{i,j} \, y_{i,j} \;,\; w_{i,j} z_{i,j} \;,\; w_{i,j} \;) \qquad\qquad (3.5)$$

Also, $S^W(u,v)$ is a tensor vector that represents a point on the surface,

$$S^W(u,v) = (\; wx(u,v), wy(u,v), wz(u,v), w \;) \qquad\qquad (3.6)$$

For more simplification, all the weights for the control points were set to unity. This simplified representation made the coding of NURBS surfaces easier, more compact, and shortened the CPU runtime. For modeling the original data (the CAD model), the NURBS curve is identified by a matrix representing its control polygon, curve degree, and knot vector. Similarly, a NURBS surface is identified by a 3D matrix representing

the control polyhedron, surface degree in both u, and v parameter directions, and knot vectors in for both u, and v directions.

3.1.2. Simulation of the sampling process

The work done in this research deals with the problem of allocating a predetermined set of sample points so that the surface fitted to that point set is as close as possible to the CAD surface. To simulate the process of point probing, some factors have to be considered, i.e., the part form error due to the manufacturing process, and the measurement uncertainty. Different form error patterns were simulated and imposed on the surface CAD data. The simulation of measurement uncertainty was accomplished by adding a random value to the coordinates of each sampled point. This random measurement error value is obtained from the calibration sheet of the DEA Mistral CMM at the IMS Centre, which is ± 0.004 mm. The selection of the random measurement error is based on a 6σ uniform distribution, with -0.004 mm. located at -3σ, and +0.004 mm. at $+3\sigma$, respectively. The sample points resulting from the algorithms are, then, fitted to obtain the substitute geometry. The final step is to compare the substitute geometry to the CAD surface data, and compute the deviation between both surfaces. The maximum deviation between the two surfaces in each algorithm is used as a parameter to determine the effectiveness of that algorithm. For each of the developed algorithms, the implementation was carried out for NURBS curves, and then generalized for NURBS surfaces.

58

3.1.3. Data fitting

The problem of fitting the data obtained to a free form surface is formulated in this way: given a set of data points **Q** that represents the sample points, determine the free form surface that best fits this set of data. Since the design information, the ideal geometric feature data, is available, the problem becomes simpler than in the case of reverse engineering where no information is available about the geometric feature. The u and v parameters for each sample point are known from the sampling plan. The number of control points in both directions, and the knot vectors are also known. Furthermore, the surface degree in both u and v directions is known. The problem then becomes to determine the locations of the control points that minimize the least square error between the fitted data and the ideal geometric feature. The fitting problem is then reduced to solving a set of linear equations. As per Eq. 3.4, The original format of a NURBS surface

is $S^w(u,v) = \sum\limits_{i=0}^{n} \sum\limits_{j=0}^{m} N_{i,p}(u) \ N_{j,q}(v) \ P_{i,j}^w$. Let P_{SG}^w be the set of control points for the

substitute geometry. To determine , P_{SG}^w the following formula is used:

$$P_{SG}^w = \sum\limits_{i=0}^{n} \sum\limits_{j=0}^{m} N_{i,p}(u) \ \left\{S_{i,j}^w(u,v)\right\}^{-1} \ \left[N_{j,q}(v)\right]^{-1}$$

(3.7)

Once P_{SG}^w is determined, the construction of the substitute geometry is straightforward. To construct the substitute geometry, Eq. 3.4 is used. Let $S_{SG}^w(u,v)$ be the equation of the substitute geometry. Using equation 3.4, $S_{SG}^w(u,v)$ is computed as

59

$$S_{SG}^{w}(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,p}(u) \ N_{j,q}(v) \ P_{SG_{i,j}}^{w} \ .$$

<div align="right">(3.8)</div>

Within Eq. 3.8, the knot vectors, and degrees of the CAD surface are used. Once the

substitute geometry is constructed, its deviation from the original data is carried out

through computing the normal distance between each point on the substitute geometry,

and the adjacent point on the CAD model. The maximum deviation between the two

surfaces is taken to be the measure of the sampling scenario effectiveness.

3.1.4. Curve and surface samples.

Extensive experimentation was performed on different NURBS curves and surfaces to

test the different sampling scenarios. A sample NURBS curve, and a sample NURBS

surface are used for all methods, for demonstration within this text. Figures 3.1 and 3.2

show the NURBS curve together with its control polygon, and the NURBS surface

together with its control polyhedron respectively.

For the sample NURBS curve shown in Figure 3.1: curve degree "p" = 3, and the

curve knot vector "U" is [0, 0, 0, 0, 0.1454, 0.3129, 0.4974, 1, 1, 1, 1]. The curve control

polygon is shown in dashed lines.

The NURBS surface of Figure 3.2 is of the third degree in both u- and v- directions,

and has knot vectors in both directions of U = [0, 0, 0, 0, 0.1406, 0.3035, 0.5132, 1, 1, 1,

1], and V = [0, 0, 0, 0, 0.1242, 0.3322, 1, 1, 1, 1]. The control polyhedron of the surface is

plotted in dashed lines.

To demonstrate the behavior of the sampling algorithms, they were applied to sample

points from both the sample surface, and curve, respectively. For the curve, the sample

size was fixed at 8 points, whereas the sample grid for the surface is set to

$7_{u-direction} \times 8_{v-direction}$.
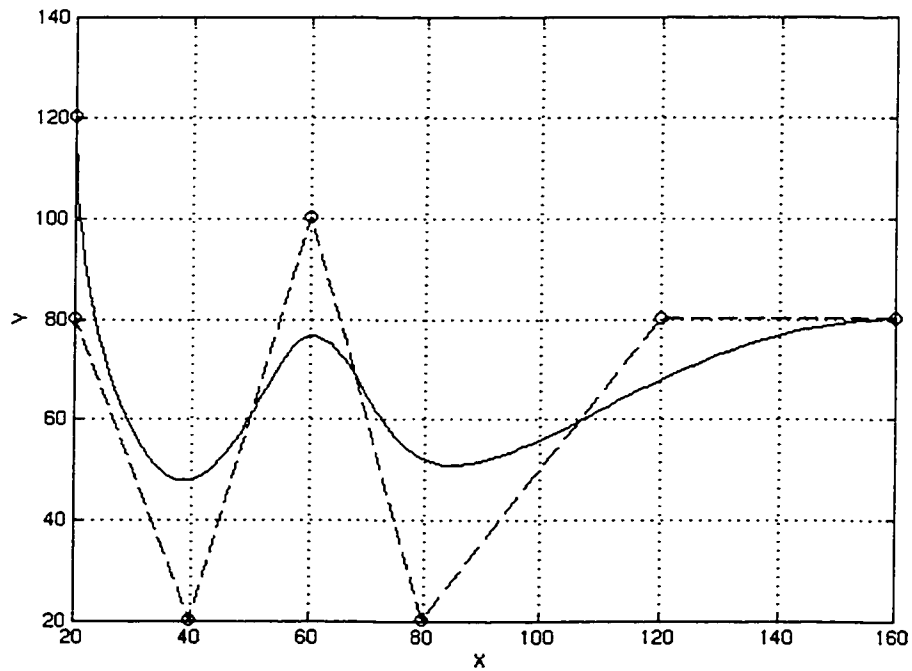
Figure 3.1. Sample NURBS Curve
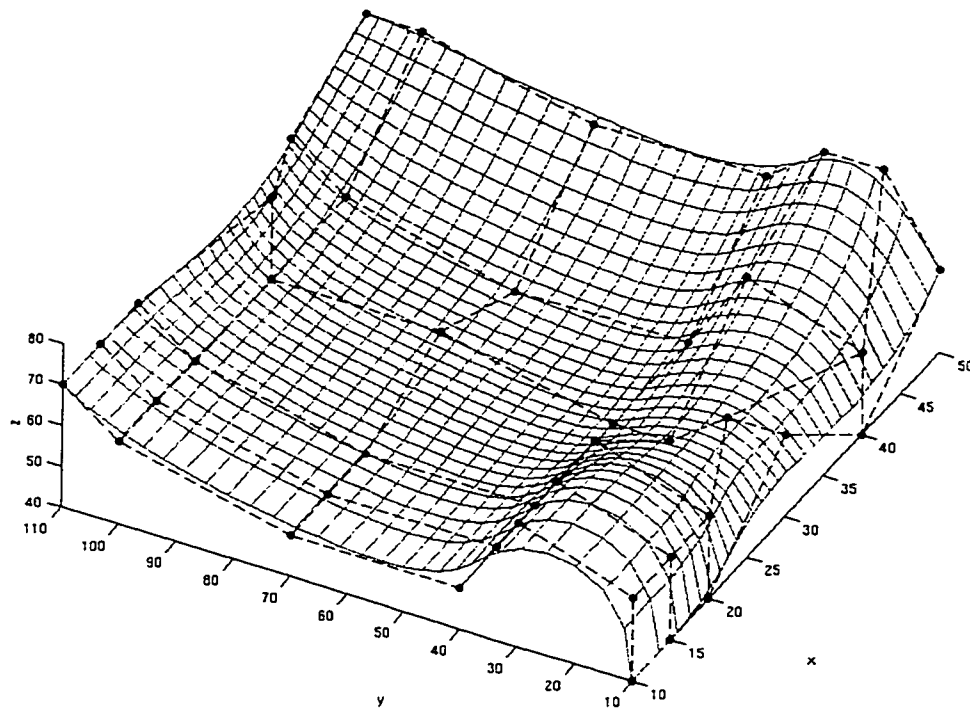
Figure 3.2. Sample NURBS Surface

3.2. Equi-Parametric Sampling

This is a simple method of equally distributing the sample points in the u-v space, i.e. allocating the sample points at equal distances on both the u-, and v- axis. This method is the simplest amongst all the sampling methods. It successfully represents simple, relatively flat NURBS surfaces with relatively no significant changes in curvatures. It has the drawback of being insensitive to different complexity levels that might exist in free form surfaces such as surfaces with sharp curvature changes and unequal surface-patch sizes. Equi-parametric sampling is used to provide an approximate the sample size for the surface. Figure 3.3 shows a flow diagram which explains the equi-parametric sampling method.
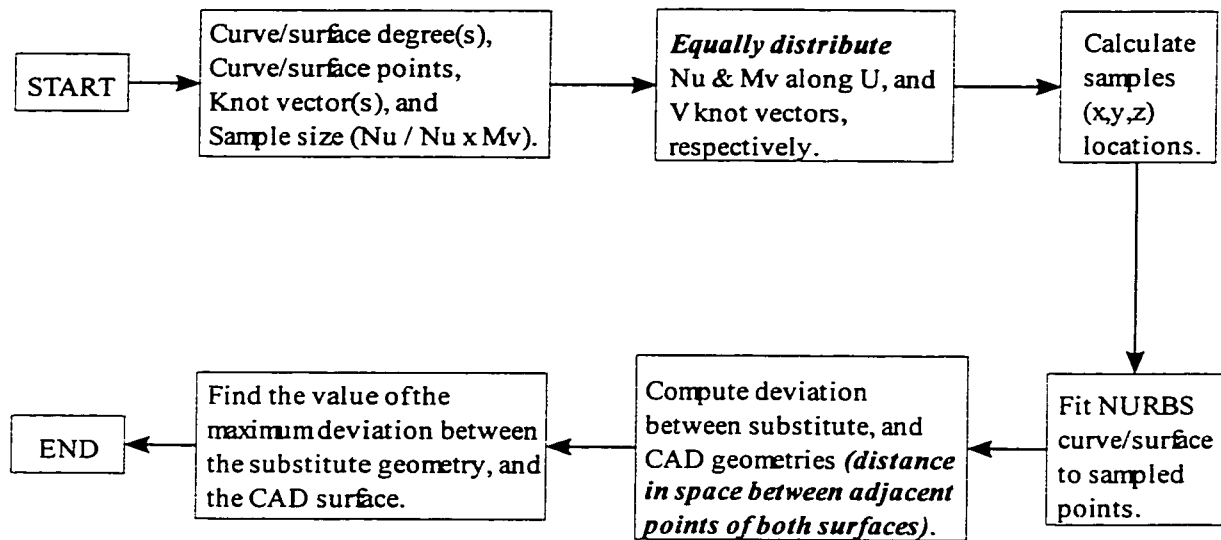
Figure 3.3. Flow Diagram Of Equi-Parametric NURBS Sampling

Figure 3.4 shows the result of applying the equi-parametric sampling method to selected NURBS curve and surface. The sample size needs to be dramatically increased to achieve better results using this method. For instance, in the case of the sample NURBS surface illustrated in figure 3.4, a sample size of $9_{u-direction} \times 11_{v-direction}$ is needed to reduce the maximum deviation between the substitute geometry, and the CAD model to 0.009 mm. (72% reduction). This means a 77% increase in the sample size and, therefore, in the inspection time. Moreover, this method is not sensitive to complex areas of a NURBS surface. Other sampling algorithms need to be developed that are able to manipulate NURBS curves/surfaces with higher levels of shape complexity. The alternate algorithms are based on the knot vector(s) spans, and the NURBS curve/surface mean curvature.
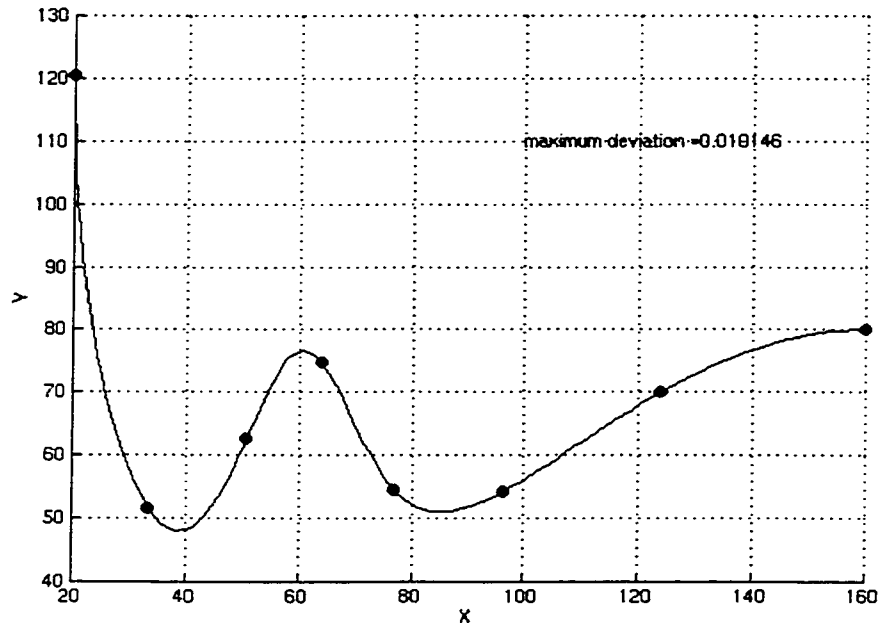
Sample size = 8 points
Maximum deviation from CAD data = 0.010 mm.
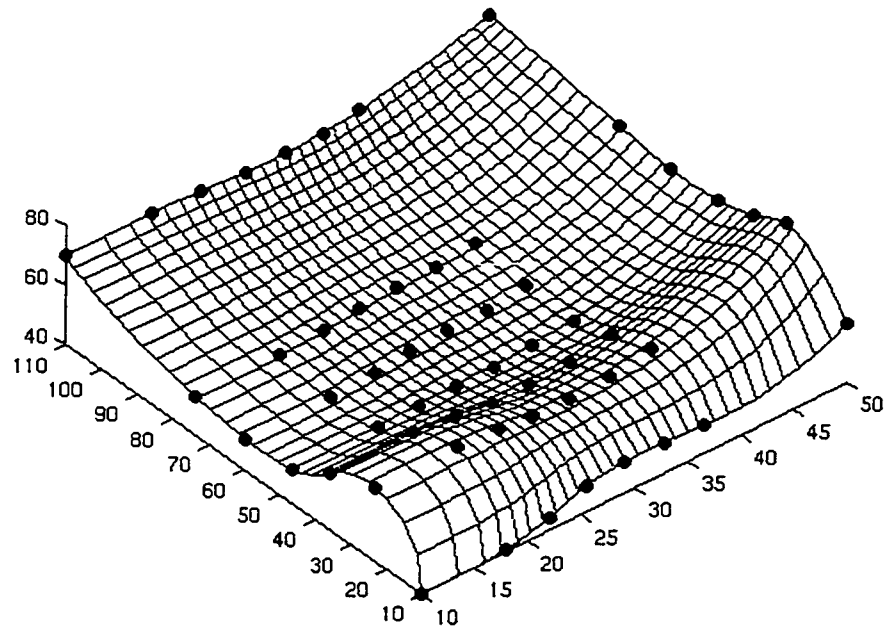
Sample size = $7_{u-direction} \times 8_{v-direction}$.
Maximum deviation from CAD data = 0.032 mm.

Figure 3.4. Equi-parametric Sampling Applied To The Sample Curve, And Surface

3.3. Span/Patch Size Based Sampling

In this methodology, the NURBS surface is resolved into its patches (obtained from the knot vectors). In this section of the algorithm, the knot vectors play the major role. First, the span lengths are computed in the parameter u direction. Second, span lengths in the parameter v direction are computed. The surface patch in the uv parametric space is a rectangle, the edges of which are composed of u-spans, and v-spans. The patch size is, then, calculated as the multiplication of the length of a u-span by the length of the v-span coinciding with it.

Figure 3.5 demonstrates how a surface patch size is computed. The surface patches are, then, ranked according to their sizes. Given a predetermined sample size, the samples are distributed amongst the surface patches according to the rank of each patch; i.e., the higher the patch rank, the larger its share of the samples in both the u- and v-directions. This method produces sampling plans that successfully represent NURBS surfaces with variations in patch sizes (resulting from the use of non-uniform knot vectors). However, patches that are relatively very small can be ignored. In some cases, those relatively small patches can include portions of the surface with sharp variation in the surface curvature. Therefore, this methodology can produce inaccurate sampling plans, in the case of sharp curvature changes, especially when they occur within relatively very small surface patches. Figures 3.7 and 3.8 show the flow diagram of the Patch Size Based Sampling

65

Methodology, and the results of applying it to sample NURBS curve, and surface respectively.
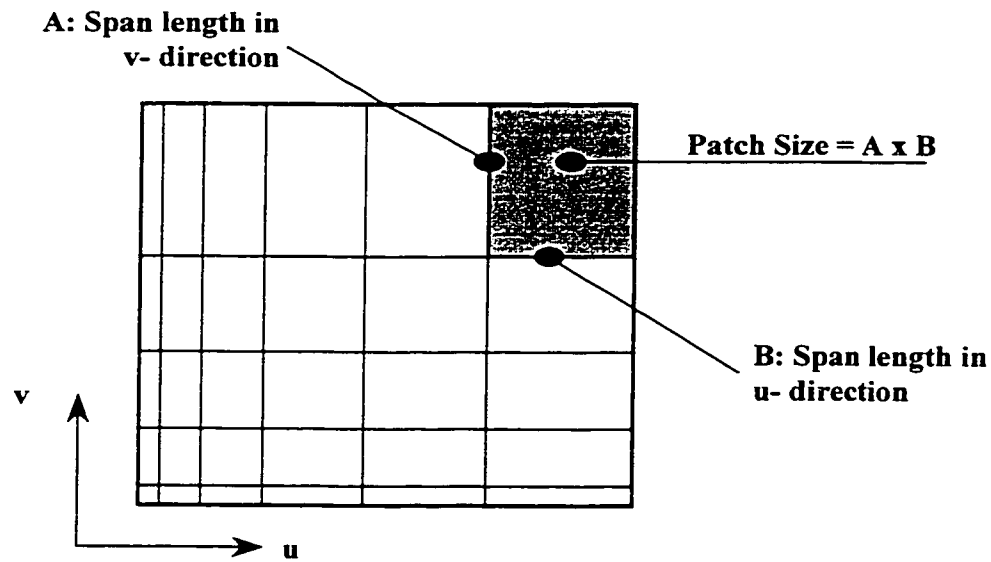
Figure 3.5. Computation Of NURBS Surface Patch Size
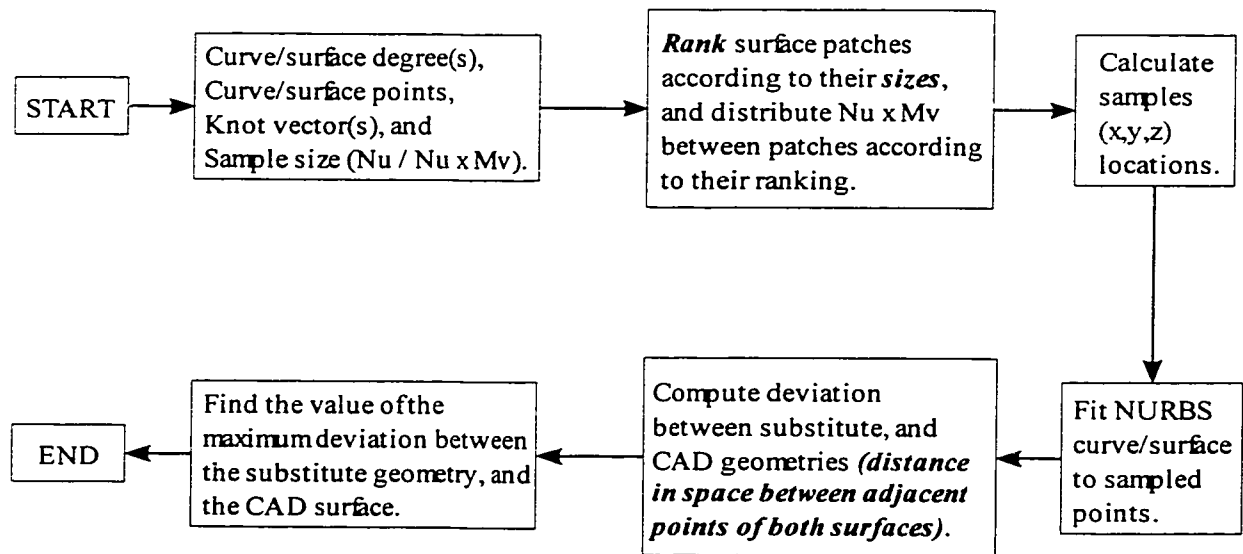
Figure 3.6. Flow Diagram Of Patch Size-Based NURBS Sampling

66
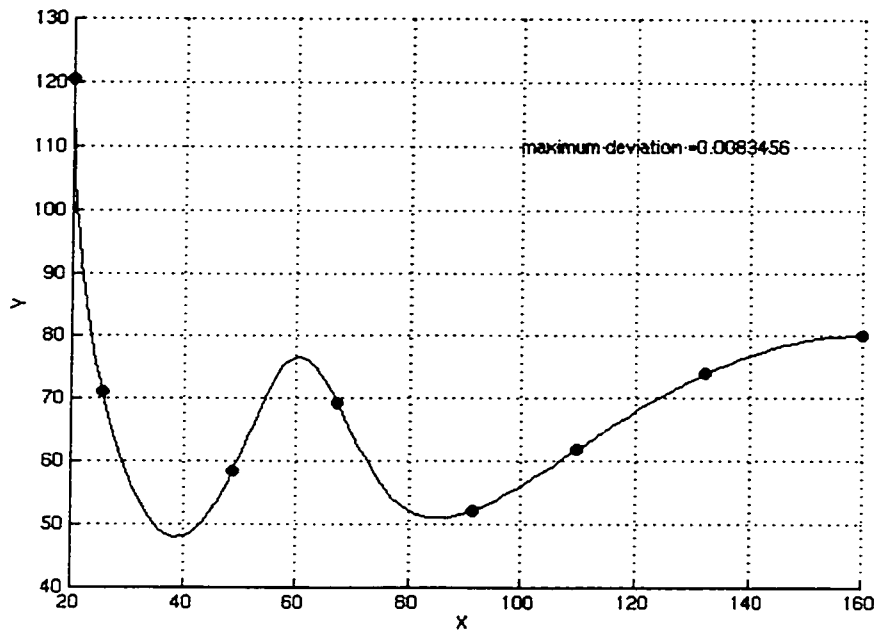
Sample size = 8 points
Maximum deviation from CAD data = 0.008 mm.
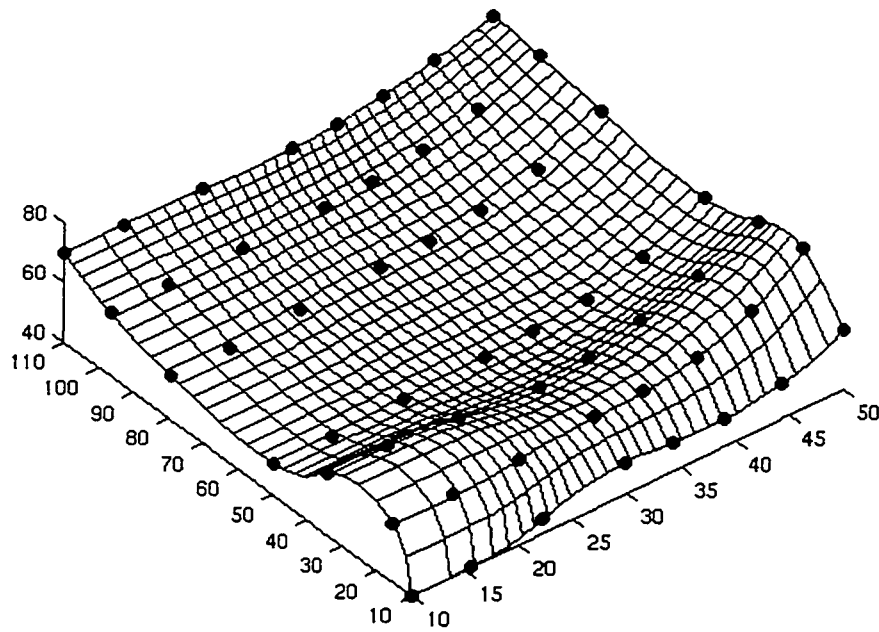
Sample size = $7_{u-direction} \times 8_{v-direction}$.
Maximum deviation from CAD data = 0.008 mm.

Figure 3.7. Span/Patch Size Based Sampling Applied To Sample Curve, And Surface

3.4. Span/Patch Mean Gaussian Curvature-Based Sampling

This methodology ranks surface patches based on the patch mean Gaussian curvature. For each point on the surface, the Gaussian curvature is calculated. The formula used for Gaussian curvature at each point on the surface is (Mortenson, 1985):

$$K(u,v) = \frac{L(u,v)N(u,v) - M^2(u,v)}{E(u,v)G(u,v) - F^2(u,v)} \tag{3.9}$$

N(u,v) is the surface normal at (u,v), calculated as:

$$\mathbf{N}(u,v) = \frac{\partial \mathbf{P}}{\partial u} \times \frac{\partial \mathbf{P}}{\partial v} = \mathbf{P}^u \times \mathbf{P}^v \text{ where } \begin{bmatrix} \mathbf{P}^u \\ \mathbf{P}^v \end{bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial u} & \dfrac{\partial y}{\partial u} & \dfrac{\partial z}{\partial u} \\ \dfrac{\partial x}{\partial v} & \dfrac{\partial y}{\partial v} & \dfrac{\partial z}{\partial v} \end{bmatrix} \tag{3.10}$$

where $\mathbf{P}^u$, $\mathbf{P}^v$ are the partial derivatives of the surface in the u, and v directions respectively. Other partial derivative vectors that are used within the Gaussian curvature formula are $\mathbf{P}^{uu}$, $\mathbf{P}^{vv}$, and $\mathbf{P}^{uv}$, which are calculated as:

$$\begin{bmatrix} \mathbf{P}^{uu} \\ \mathbf{P}^{vv} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial^2 x}{\partial u^2} & \dfrac{\partial^2 y}{\partial u^2} & \dfrac{\partial^2 z}{\partial u^2} \\ \dfrac{\partial^2 x}{\partial v^2} & \dfrac{\partial^2 y}{\partial v^2} & \dfrac{\partial^2 z}{\partial v^2} \end{bmatrix} \text{ and } \mathbf{P}^{uv} = \begin{bmatrix} \dfrac{\partial^2 x}{\partial u \partial v} & \dfrac{\partial^2 y}{\partial u \partial v} & \dfrac{\partial^2 z}{\partial u \partial v} \end{bmatrix}^T \tag{3.11}$$

At any point (u,v) on the surface, a unit normal vector is calculated as:

$$\hat{n} = \frac{N}{|N|} = \frac{P^u \times P^v}{|P^u \times P^v|} \tag{3.12}$$

The rest of the terms used in the Gaussian curvature formula are calculated using the following formulae:

68

$$L(u,v)=\hat{n}\cdot P^{uu} \quad M(u,v)=\hat{n}\cdot P^{uv} \quad N(u,v)=\hat{n}\cdot P^{vv}$$

$$E(u,v)=P^u\cdot P^u \quad F(u,v)=P^u\cdot P^v \quad G(u,v)=P^v\cdot P^v \tag{3.13}$$

The surface is divided into patches, as in the previous method, the mean Gaussian curvature per surface patch is calculated, and the patches are ranked according to their mean Gaussian curvatures. The sample points grid is then distributed amongst the surface patches according to their ranking, i.e., the higher the patch ranking, the larger its share of sample points. Unlike the former two, this method places emphasis on the zones with the highest curvatures. However, it might concentrate the majority of the sample points in the patch with the highest mean Gaussian curvature, no matter how small or large that patch is. Therefore, larger patches could be represented with a relatively small number of sample points, which may lead to an inefficient sampling plan. Figures 3.8 and 3.9 show the flow diagram, and the sample runs of the mean Gaussian curvature-based sampling methodology.

3.5. Combined Span/patch Size And Span/patch Mean Gaussian Curvature Based Sampling

In order to satisfy the need for an effective and efficient sampling plan that properly represents a NURBS surface which has both the characteristics of different patch sizes and curvature changes, the former two methods were combined into a new method. It

allows the user to specify a weight for each of the combined criteria. For instance, if a sharp change in the mean Gaussian curvatures is present, the user might prefer to emphasize the role of the mean Gaussian curvature based sampling. On the other hand, for a surface with no significant changes in its mean Gaussian curvatures, but the surface patches vary greatly in size, the user might emphasize more the effect of the variation in the size of surface patches. Figure 3.10 shows the sampling plan resulting from applying the combined method to a sample NURBS curve and surface. In the example used in this section, equal weights are given to the span/patch mean Gaussian curvature-based, and patch/span size-based sampling.
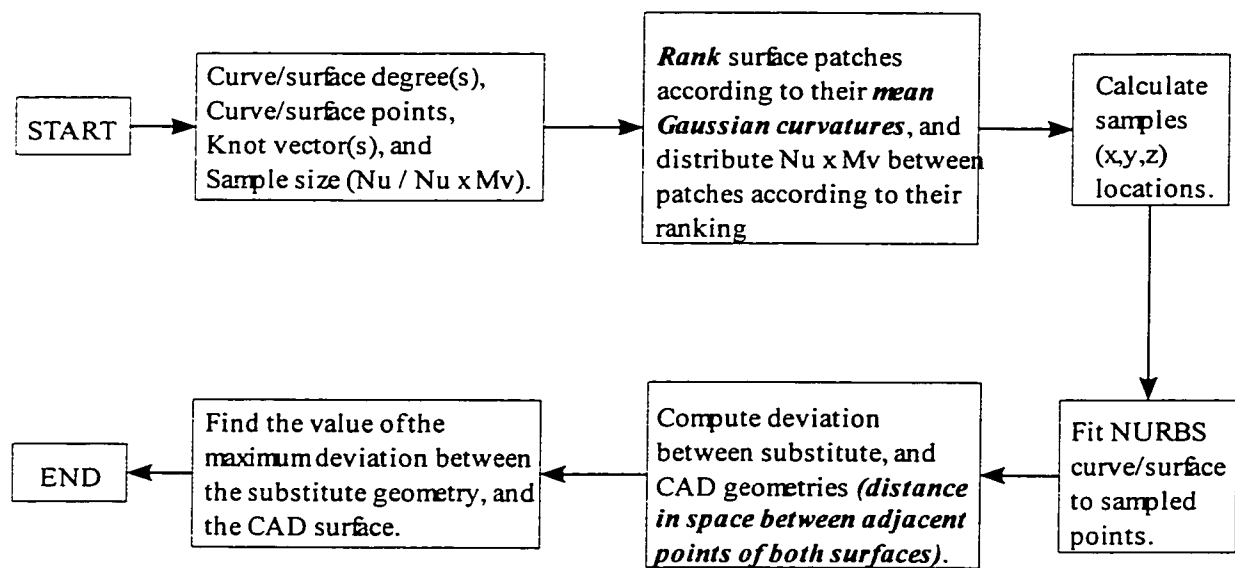
Figure 3.8. Flow Diagram Of Patch Mean Gaussian Curvature-Based Sampling

70
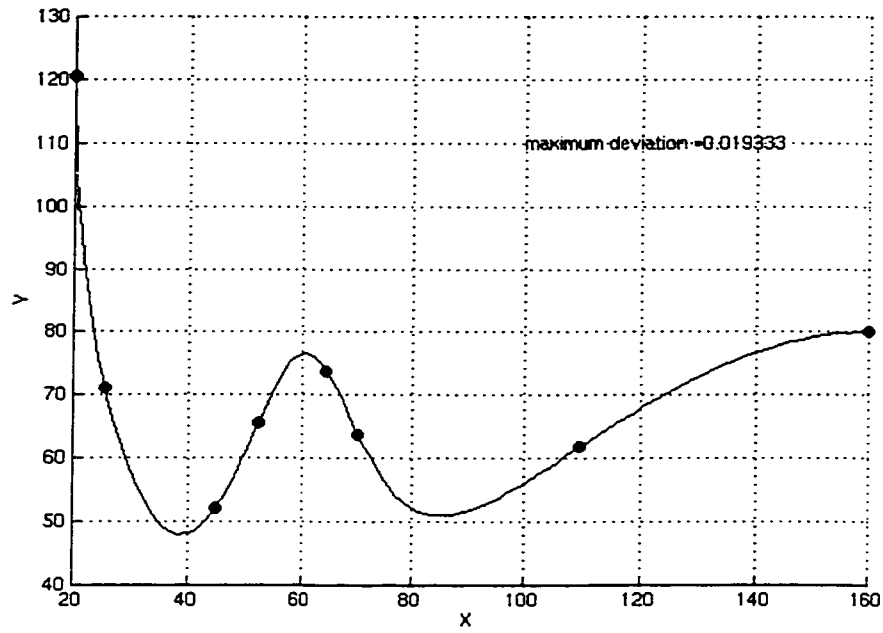
Sample size = 8 points
Maximum deviation from CAD data = 0.019 mm.
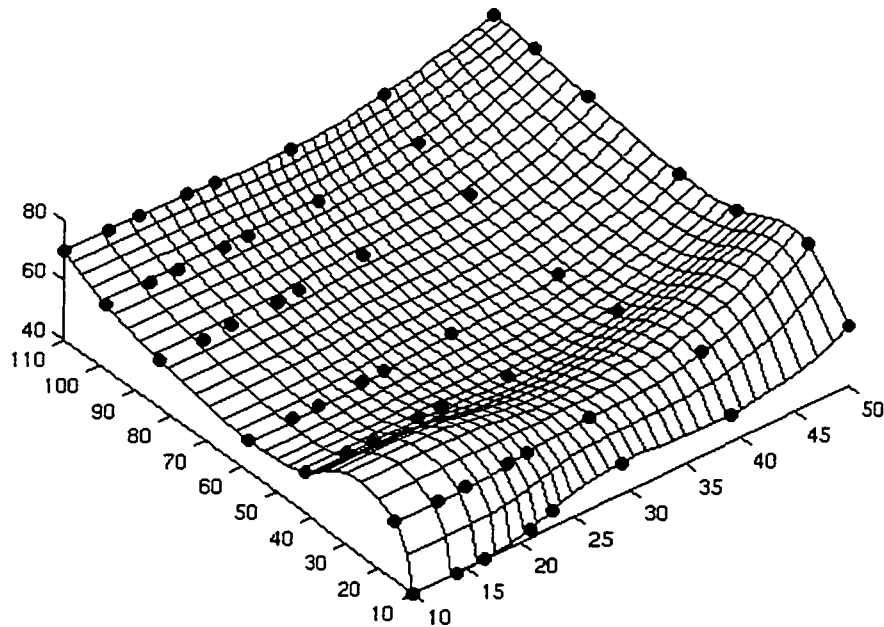
Sample size = $7_{u-direction} \times 8_{v-direction}$.
Maximum deviation from CAD data = 0.073 mm.

Figure 3.9. Sample NURBS Curve And Surface Sampled Using Span/Patch Mean

Gaussian Curvature-Based Algorithm

71
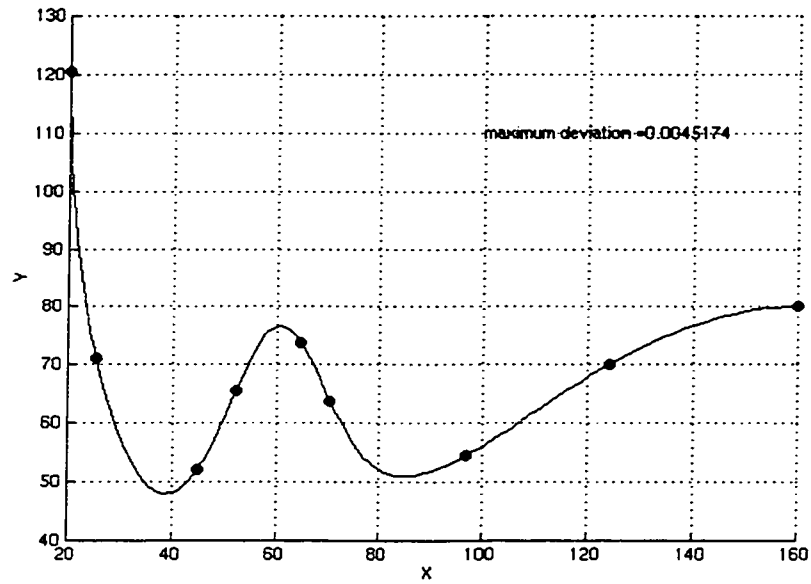
Sample size = 8 points.

Combining factors = $Weight_{Curvature}$: $Weight_{Patch}$ = $1 : 1$.

Maximum deviation from CAD data = 0.004 mm.
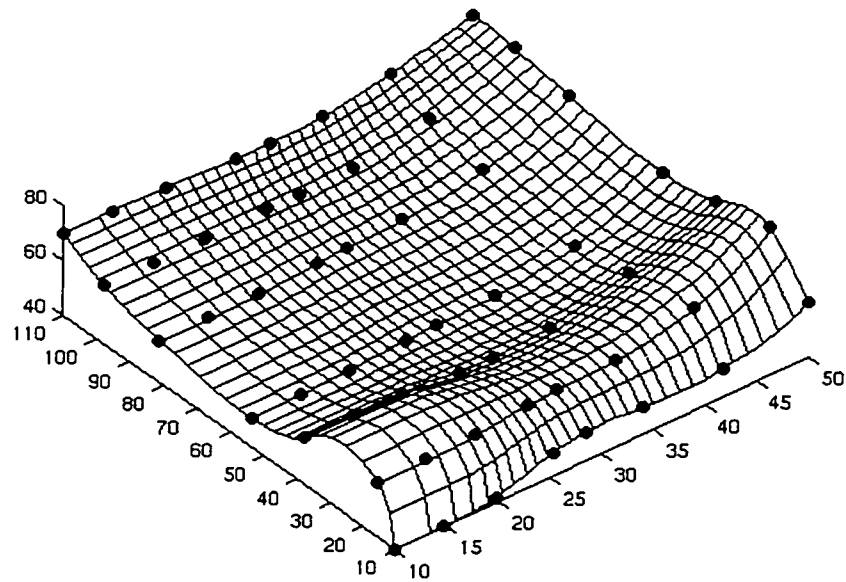
Sample size = $7_{u-direction} \times 8_{v-direction}$.

Combining factors = $Weight_{Curvature}$: $Weight_{Patch}$ = $1 : 1$.

Maximum deviation from CAD data = 0.018 mm.

Figure 3.10. NURBS Curve And Surface Sampled Using The Hybrid Sampling

Approach

72

3.6 Automatic Selection Of The Surface Sampling Algorithm

As stated earlier, the goal of the work presented in this thesis is to develop a system for automatic sampling of NURBS surfaces that provides the user with the flexibility to choose a sampling method that is suitable for the features of the surface under consideration. The hybrid approach provides an efficient means of combining the patch size, and the patch mean curvature methods. However, the selection of the weights for each method dependent on the user's skills. To increase the flexibility of the system, an algorithm was developed to automatically select the sampling algorithm that is suitable for the NURBS surface.

The algorithm performs a complexity check on the NURBS surface to determine which sampling algorithm is the most suitable to apply. The algorithm performs complexity check on the NURBS surface, which results in two factors, namely, the curvature factor, and the patch size factor. The curvature factor determines whether the surface change in curvature is of significance to be taken into consideration, or not. Likewise, the patch size factor determines if the surface patch size is to be taken into consideration in the sampling process or not.

3.6.1. Surface curvature analysis

This is the first step taken by the algorithm for automatic selection of sampling scenarios. This module, first, computes the curvature at each point on the surface grid.

73

This is done using equations 3.9 through 3.13. The result of this step is two matrices, representing the surface curvature in each of the two parametric spaces u, and v. Let the surface curvatures in the u, and v directions at an arbitrary point (u_i, v_j) be given as:

$$C^u(u_i, v_j) \quad \text{and} \quad C^v(u_i, v_j) \quad , \quad i \in \{1, 2, .., K\} \quad \text{and} \quad j \in \{1, 2, .., L\} \tag{3.14}$$

Where:

C^u = value of surface curvature in u-direction.

C^v = value of surface curvature in v-direction.

i = integer index.

j = integer index.

K = size of surface grid in u-direction.

L = size of surface grid in v-direction.

The two curvature matrices are constructed as:

$$CM^u = \begin{bmatrix} C^u(u_1, v_1) & C^u(u_2, v_1) & C^u(u_3, v_1) & \ldots & C^u(u_K, v_1) \\ C^u(u_1, v_2) & C^u(u_2, v_2) & C^u(u_3, v_2) & \ldots & C^u(u_K, v_2) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ C^u(u_1, v_L) & C^u(u_2, v_L) & C^u(u_3, v_L) & \ldots & C^u(u_K, v_L) \end{bmatrix} \tag{3.15}$$

$$CM^v = \begin{bmatrix} C^v(u_1, v_1) & C^v(u_2, v_1) & C^v(u_3, v_1) & \ldots & C^v(u_K, v_1) \\ C^v(u_1, v_2) & C^v(u_2, v_2) & C^v(u_3, v_2) & \ldots & C^v(u_K, v_2) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ C^v(u_1, v_L) & C^v(u_2, v_L) & C^v(u_3, v_L) & \ldots & C^v(u_K, v_L) \end{bmatrix} \tag{3.16}$$

Where:

CM^u = Matrix of surface curvatures in u-direction.

74

CM^v = Matrix of surface curvatures in v-direction.

Second, the percent change in curvature between every two neighboring points is calculated in both u, and v axes. The result is two curvature change matrices. The curvature change at at an arbitrary point (u_i, v_j) is given as:

$$CC^u(u_i, v_j) \quad \text{and} \quad CC^v(u_i, v_j) \quad , \quad i \in \{1, 2, .., K\} \quad \text{and} \quad j \in \{1, 2, .., L\} \tag{3.17}$$

where:

$$CC^u(u_i, v_j) = \frac{C^u(u_i, v_j) - C^u(u_{i-1}, v_j)}{C^u(u_{i-1}, v_j)} \times 100$$

$$CC^u(u_1, v_j) = 0$$
$$CC^u(u_K, v_j) = 0$$
$$CC^u(u_i, v_1) = 0$$
$$CC^u(u_i, v_L) = 0 \tag{3.18}$$

and

$$CC^v(u_i, v_j) = \frac{C^v(u_i, v_j) - C^v(u_i, v_{j-1})}{C^u(u_i, v_{j-1})} \times 100$$

$$CC^v(u_1, v_j) = 0$$
$$CC^v(u_K, v_j) = 0$$
$$CC^v(u_i, v_1) = 0$$
$$CC^v(u_i, v_L) = 0 \tag{3.19}$$

The curvature change matrices are constructed as:

$$CCM^u = \begin{bmatrix} CC^u(u_1, v_1) & CC^u(u_2, v_1) & CC^u(u_3, v_1) & \dots & CC^u(u_K, v_1) \\ CC^u(u_1, v_2) & CC^u(u_2, v_2) & CC^u(u_3, v_2) & \dots & CC^u(u_K, v_2) \\ \dots & \dots & \dots & \dots & \dots \\ CC^u(u_1, v_L) & CC^u(u_2, v_L) & CC^u(u_3, v_L) & \dots & CC^u(u_K, v_L) \end{bmatrix} \tag{3.20}$$

$$CCM^v = \begin{bmatrix} CC^v(u_1,v_1) & CC^v(u_2,v_1) & CC^v(u_3,v_1) & ... & CC^v(u_K,v_1) \\ CC^v(u_1,v_2) & CC^v(u_2,v_2) & CC^v(u_3,v_2) & ... & CC^v(u_K,v_2) \\ ... & ... & ... & ... & ... \\ CC^v(u_1,v_L) & CC^v(u_2,v_L) & CC^v(u_3,v_L) & ... & CC^v(u_K,v_L) \end{bmatrix} \qquad (3.21)$$

Where:

CCM^u = Matrix of surface curvatures' changes in u-direction.

CCM^v = Matrix of surface curvatures' changes in v-direction.

Third, the curvature change matrices are combined into one matrix. This is done through the following formula:

$$CCM(i,j) = \frac{CCM^u(i,j) \times K + CCM^v(i,j) \times L}{K+L}$$
$$i \in \{1,2,..,K\}$$
$$j \in \{1,2,..,L\} \qquad (3.22)$$

where:

CCM = Final curvature change matrix.

The elements of the final curvature change matrix (CCM) are then divided by the highest value in the matrix.

$$CCM^1(i,j) = \frac{CCM(i,j)}{|CCM(i,j)|_{MAXIMUM}}$$
$$i \in \{1,2,..,K\}$$
$$j \in \{1,2,..,L\} \qquad (3.23)$$

The curvature factor (referred to as CFactor) is calculated as the percentage change between the minimum, and the maximum elements in the Curvature Change Matrix CCM^1. The following formula is used:

$$CFactor = \frac{(|CCM^1|_{MAXIMUM} - |CCM^1|_{MINIMUM})}{|CCM^1|_{MINIMUM}} \times 100 \qquad (3.24)$$

The curvature factor is compared to a threshold value (CLimit). The value of CLimit is specified by the user of the algorithm. The value of CLimit is inversely proportional to the algorithm sensitivity to the change of the surface curvature; i.e., the higher the value of CLimit, the more "tolerant" the algorithm to the changes in the surface curvatures, and vise versa. In other words, if a high accuracy of the inspection process is needed, a lower value of CLimit is set, and vise versa. If CFactor was less than CLimit, the curvature change is considered insignificant in affecting the sampling strategy. If, otherwise, CFactor lied beyond CLimit, the surface curvature change is considered of a significant effect on the sampling strategy and, therefore, taken into account. Taking the change in the surface curvatures into account means applying the patch mean curvature based sampling. The next step is to test the change in the surface patch sizes.

3.6.2. Surface patch size analysis

In this section of the algorithm, the surface patch sizes are computed, as explained in section 3.3. After the surface patches' sizes are computed, the percent change between the minimum, and the maximum surface patches is calculated. This percent change in patch sizes is referred to as the patch factor (PFactor). As in the curvature analysis section, PFactor is compared to a threshold (PLimit), which is used to determine whether the effect of surface patch size changes is significant or not. Like CLimit, the value of PLimit

is set by the user of the algorithm. PLimit is inversely proportional to the sensitivity of the algorithm to the change in the surface patch sizes. If PFactor was less than PLimit, the change in surface patch sizes is not considered of a significant effect on the sampling scenario, and vise versa. Taking the effect of surface patch sizes into account in the sampling process means applying the surface patch size based algorithm.

3.6.3. Making the decision

Having the surface complexity check carried out, the automatic selection algorithm decides which sampling algorithm to apply. First, the algorithm checks the CFactor value, then it checks the PFactor value. As mentioned earlier, those values determine the significance of the curvature change, and the patch size changes on the sampling scenario, respectively. If both values exceeded the threshold values (CLimit, and PLimit), then the automatic selection algorithm performs hybrid sampling, and uses CFactor, and PFactor to combine the Curvature-Based, and Patch Size-Based sampling scenarios, respectively. Figure 3.11 demonstrates a flow diagram of the logic applied within the algorithm for automatic selection of sampling scenario.

One issue that is of great importance to emphasize is the values of the thresholds CLimit, and FLimit. The values of those limits are based on how tolerant the program user is to changes in surface curvatures, and patch sizes. In other words, the user adjusts

the values of these limits to push the program to pay more attention to either or both factors. If both limits were set to 1, this means that any change in the surface curvatures, or patch sizes will be taken into consideration. The higher the value of the limits, the more the tolerance given to the adjacent surface complexity feature.

The determination of the thresholds was left to the program user to decide to provide the user with more flexibility, and allow him/her to set the limits according the type of the surface, the manufacturing operation, or the inspection requirements. The algorithm described in this section is integrated with the sampling algorithms that, together, provide an integrated system for the sampling of free form surfaces. This integration will be discussed in details in the next chapter.

3.7. Optimal Sampling Of Free Form Surfaces

When the sampling problem is too complex to resolve, or when the heuristics discussed in this chapter can not reach a satisfactory solution, or when an optimal allocation of sampling points is required, the inspection planner may need a utility to produce an optimal sampling plan. Such an alternative is made possible through the application of Genetic Algorithms (GA's) in the sampling of free form curves, and surfaces (ElKott, D., ElMaraghy, H., and Nassef, A., 1998, and 1999). A GAs MATLAB toolbox was integrated with the surface construction, sampling, and fitting functions. The

objective function of the optimization problem was set to minimize the maximum deviation between the original NURBS surface (geometric model), and the NURBS surface fitted to the sampled points. Within the GA's objective function, the sampling process is simulated, a NURBS surface is fitted to the sampled data, and the maximum deviation between the fitted surface and the geometric model is calculated and minimized. Figure 3.12 demonstrates the block diagram of GAs optimization of NUBS curves and surfaces samples locations. The GA's parameters used in this study are:

- Population size: 70

- Number of generations: 100

- Number of times to apply boundary mutation, uniform, and non uniform mutations: 4

- Number of times to apply simple and whole arithmetic cross over: 2

- Number of times to apply whole non-uniform mutation: 4

- Number of times to apply heuristic cross over: 2

Genetic Algorithms-based NURBS surfaces sampling optimization was applied to a group of NURBS surfaces which represented different levels of complexity, with variety of sample sizes. Results of GA's optimization for the sample NURBS curve, and surface are demonstrated in figure 3.13.

80
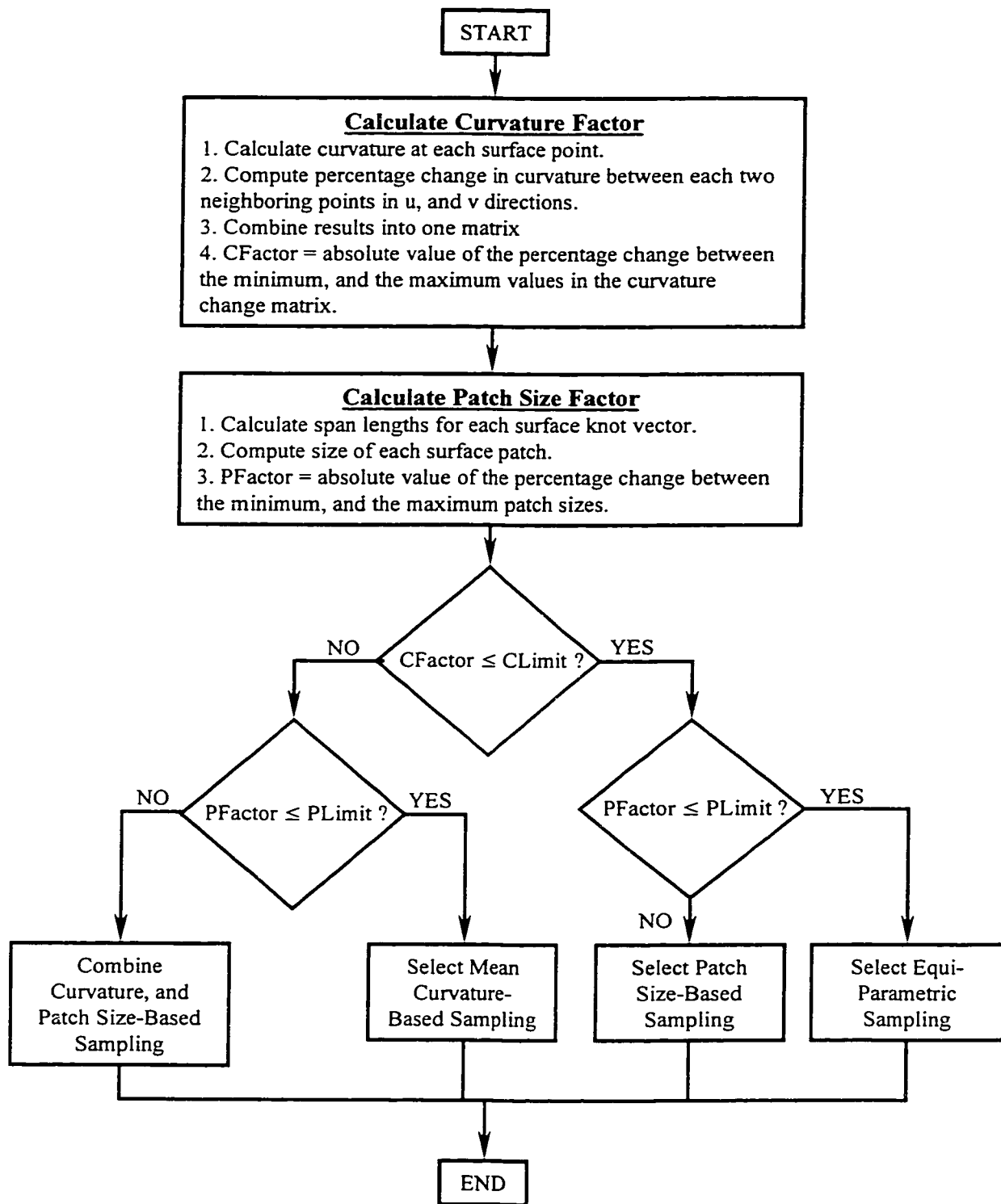
Figure 3.11. Automatic Selection Of Sampling Algorithm Based On Surface

Complexity Check
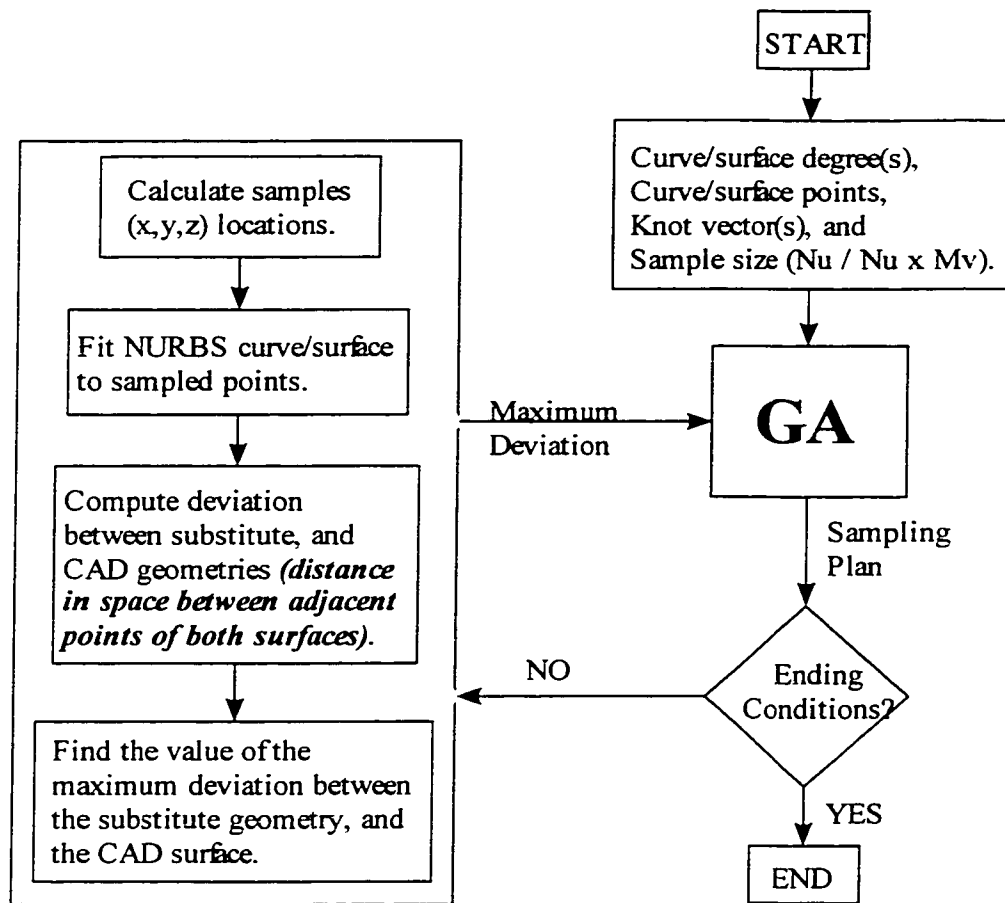
Figure 3.12. Flow Diagram Of The GA's Optimization Of NURBS Surface Sampling

3.8. Discussion

This chapter introduced the different algorithms developed for the sampling of free form surfaces, as well as the implementation of Genetic Algorithms (GAs) in the sampling problem. The sampling algorithms operate in the parametric space to calculate the coordinates of a sample grid whose size is predetermined.

Equi-parametric sampling equally distributes the sample points in the parametric

space. This method is not sensitive to changes in the surface curvature, or patch size. This method is satisfactory for relatively flat surfaces with limited, or small curvature changes.

Patch size-based sampling distributes the sample points among the surface patches according to their sizes. The larger the patch size, the larger its share of sample points. The drawback of this algorithm is its insensitivity to sharp curvature changes, especially in relatively small patches.

Curvature-based sampling ranks the surface patches according to the value of mean curvature per each surface patch. The drawback of this method is its insensitivity to the variation in surface patch size. The combined surface patch size and mean curvature based sampling algorithm emphasizes both effects of the variation in patch size, and curvature.

An algorithm for automatically selecting the proper sampling algorithm based on the surface features was introduced. It testes the surface complexity in terms of its curvature variation, and patch size variation. The algorithm then decides which sampling algorithm to apply.

A Genetic Algorithms-based method has been developed to offer optimum sampling

of free form surfaces. Extensive simulations and numerical tests have been conducted to test the performance, and reliability of the sampling methodologies presented in this chapter. Surfaces with different levels of complexity have been sampled using each methodology, and the results have been analyzed. Results of simulations, and numerical tests are discussed in the next chapter.
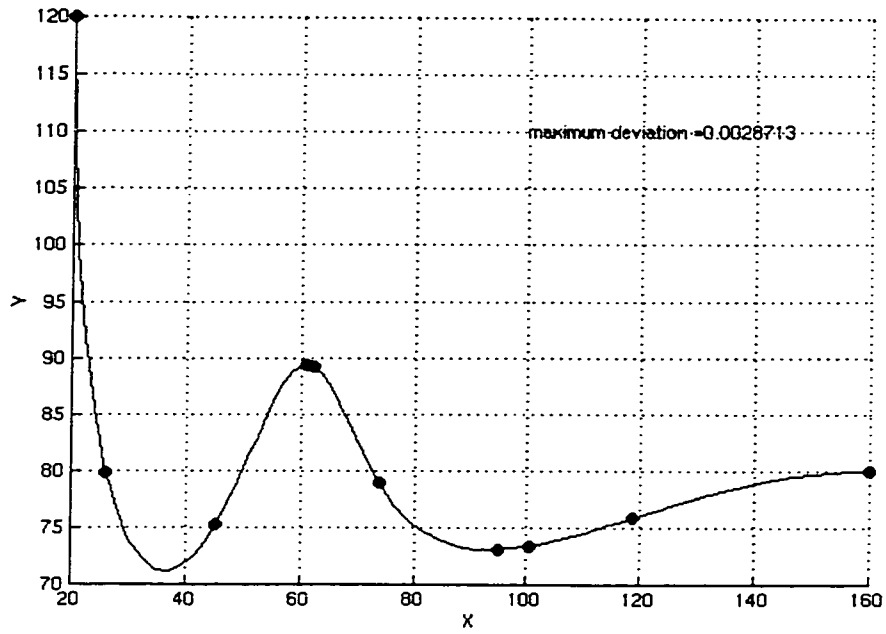
Sample size = 8 points.
Maximum deviation from CAD data = 0.003 mm.
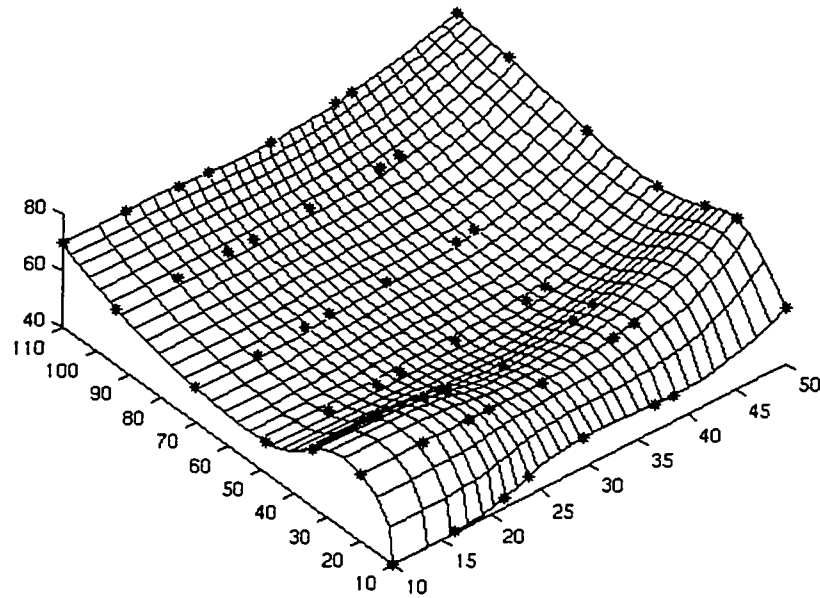
Sample size = $7_{u-direction} \times 8_{v-direction}$.
Maximum deviation from CAD data = 0.005 mm.

Figure 3.13: NURBS Curve, And Surface Sampled Using GA-Based Optimum

Sampling

CHAPTER FOUR

SAMPLING ALGORITHMS SIMULATION

This chapter describes the simulation study performed on the sampling algorithms introduced in the previous chapter. It discusses the method used for the simulation study, and the results of applying the sampling algorithms. The Genetic Algorithms (GAs) based sampling method is also demonstrated.

To emphasize the significance of the algorithms developed throughout the course of this thesis, the result from the different algorithms are compared to the results of sampling using the uniform sampling pattern introduced by in Yau and Menq (1992).

The application of the sampling algorithms to selected cases is also introduced. CAD models of different products were selected, and the sampling algorithms were applied to them. This is demonstrated in this chapter.

4.1. Methodology

This section discusses the methods applied to simulate the sampling algorithms

presented in chapter 3. The surfaces used in the simulation study, and the simulation of sources of error are discussed.

4.1.1. Simulation process outline

The simulation of sampling algorithms took place through the construction of sample NURBS surfaces, simulating the form error error due to manufacturing, applying the sampling algorithm to the surface (with form error added), simulating the measurement error component, fitting NURBS surface to the sampled points. The fitted surface simulates the substitute geometry, which is compared to the CAD surface (surface without form error added). The deviation between the CAD surface, and substitute geometry is computed for each sampling algorithm. Figure 4.1 illustrates the flowchart of the simulation of the sampling algorithms. For each surface, the sample size is fixed, in order to compare the behavior of the sampling algorithms at identical conditions. Throughout the simulation process, the weights assigned for each of the curvature based, and the patch size based sampling in the hybrid sampling algorithm were set to (CWeight : PWeight = 25:75). These values of CWeight, and PWeight are fixed for all the calculations involving the use of hybrid sampling in this simulation study.

4.1.2. Surface complexity

Throughout the context of this chapter, terms such as curvature change, patch size

factor, and complexity level are used to describe the complexity of a NURBS surface, and to distinguish between different NURBS surfaces. Two factors are used to address a NURBS surface complexity level, i.e., the change in the surface patch size, and the surface curvature change.

4.1.2.1. Surface curvature change matrix

The first measure of a NURBS surface complexity is the change in its curvature. In order to have an idea of how sharp surface curvatures can change, the NURBS surface curvature analysis is carried out, as per section 3.6.1 of the previous chapter. The higher the values of elements of the curvature change matrix, the higher the sharpness of the surface curvature change. A primary indicator to the surface complexity is the surface curvature change factor (CFactor). The higher the value of CFactor, the higher the tendency of the surface to contain sharp curvature changes, and vise versa.

4.1.2.2. Surface patch size change

A NURBS surface patch is bounded by the intersection of four isoparametric lines, and its size is calculated as described in section 3.6.2, in the previous chapter. The patch size factor (PFactor) is used as the second measure of a surface complexity. The higher the value of PFactor, the higher the probability of the surface to contain small patches. A Small surface patch might be ignored, or given an insufficient share of sample points.

This might lead to high fitting error, especially if sharp curvature changes exist in those small patches. Therefore, PFactor is used as a second measure for surface complexity.

For each of the NURBS surfaces used in this simulation study, the values of CFactor, and PFactor are indicated, as well as the U, and V knot vectors, and the surface degrees in both the u, and v directions. in order to compare the result of the different sampling algorithms, the size of the sample points grid is fixed for each surfaces, while the sampling algorithms are changed.

4.1.3. Surfaces used for simulation study

To simulate the sampling algorithms and test their behavior, NURBS surfaces with different complexity levels were designed. The surface complexity consists of two main factors, i.e., the curvature variations, and patch size variation. In this simulation study, six different NURBS surfaces are used. The simplest of the six surfaces tends to be flat. The complexity of the surfaces increases to reach its highest level in the sixth surface. Figure 4.2 illustrates the different surfaces used for testing the sampling scenarios. The surface degrees in both the u, and v parametric spaces are set to 3. It was necessary to use surfaces of the 3rd degree to maintain C^2 continuity, which is essential to compute the first and second derivatives for the surface points.

4.1.4. Form error simulation

The simulation of the form error takes place through the construction of a NURBS surface that represents the form error distribution (error pattern), and imposing it on the CAD surface. Since the error pattern is a NURBS surface, it is possible to impose different distributions of the form error. In this study, the form error distribution is based on the curvature change in the CAD surface. The surface zones where the curvature change is sharp receive a larger share of the form error. The form error is then distributed on the surface according to the curvature change matrix. Figure 4.3 illustrates the methodology used to simulate the form error. In this simulation study, the form error values simulated vary between 0 mm, and 0.008 mm, and their values vary in proportion to the change in surface curvature.

The CAD surface is constructed. A series of calculations are then performed to compute the curvature of the surface at each point in the surface grid, as well as the curvature change matrices. These calculations are discussed in section 3.6.1, and equations 3.14 through 3.23. The result of equation 3.23 is the curvature change matrix CCM^1 , whose elements' values range between a minimum value of 0.0, and a maximum value of 1.0. The elements of this matrix are, finally, multiplied by the maximum value of the expected form error (0.008 mm. In this study). The result of this final operation is a curvature-based form error distribution matrix (FEDM).

$$FEDM(i,j)=CCM^1(i,j)\times0.008\,mm.$$
$$i\in\{1,2,..,K\},\quad j\in\{1,2,..,L\}$$

(4.1)

The values obtained from equation 4.1 are then added to the surface nominal data. The result is the manufactured surface data.
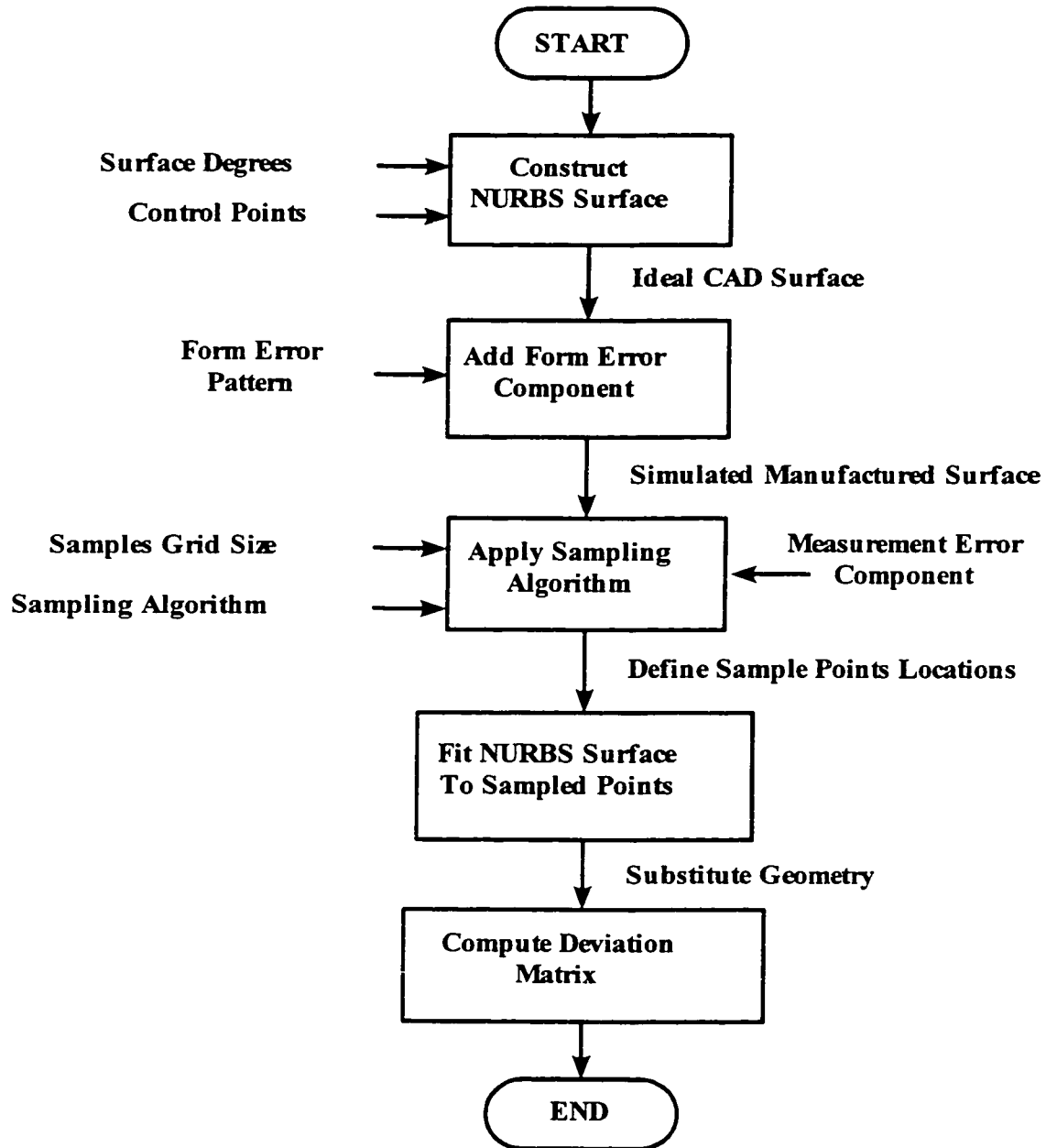
Figure 4.1. Flowchart Of Inspection Sampling Simulation Process

Surface A

Surface B

Surface C

Surface D
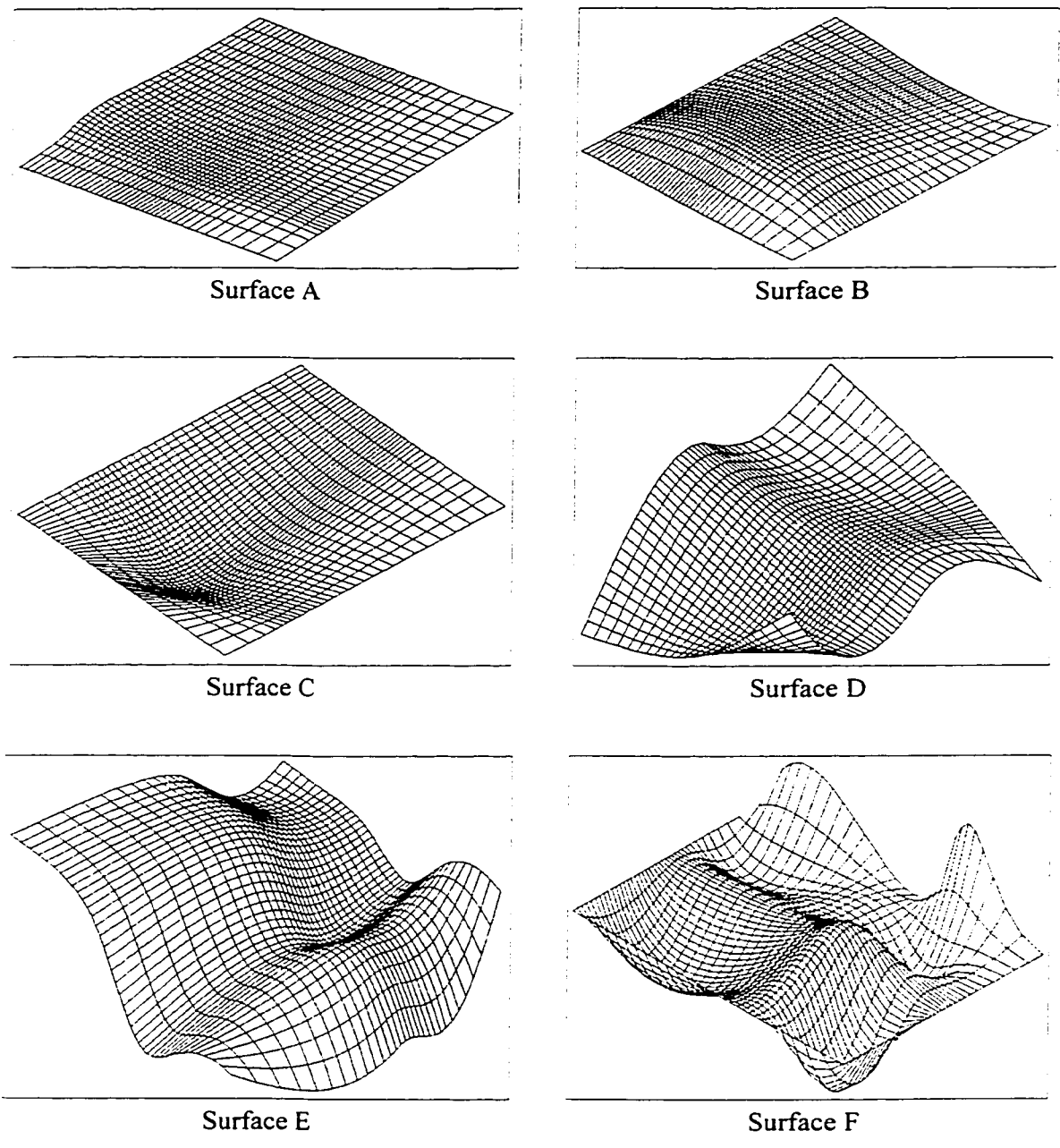
Surface E

Surface F

Figure 4.2. Surfaces Used For Sampling Algorithms Simulation

92

```
        ┌─────────┐
        │  START  │
        └─────────┘
             │
             ▼
    ┌──────────────────┐
    │  Construct CAD   │
    │  NURBS Surface   │
    └──────────────────┘
             │
             ▼
    ┌──────────────────┐
    │  Compute Surface │
    │ Curvatures Matrix│
    └──────────────────┘
             │
             ▼
    ┌──────────────────────┐
    │  Compute Curvature   │
    │  Change Matrices in  │
    │ Both u, and v directions│
    └──────────────────────┘
             │
             ▼
    ┌──────────────────┐
    │ Combine Curvature│
    │  Change Matrices │
    └──────────────────┘
             │
             ▼
    ┌──────────────────┐
    │Distribute Form Error│
    │According to Curvature│
    │   Change Matrix  │
    └──────────────────┘
             │
             ▼
        ┌─────────┐
        │   END   │
        └─────────┘
```
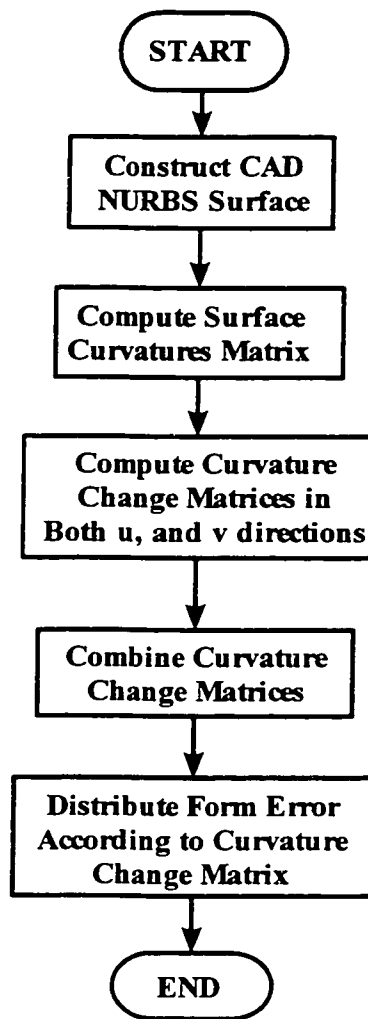
Figure 4.3. Flowchart Of Form Error Simulation

4.1.5. Measurement error simulation

In addition to the form error, the CMM measurement error is also simulated. A random measurement error of ± 0.004 mm. is used to simulate the CMM measurement error. This figure is obtained from the calibration chart of the DEA Mistral CMM, IMS Centre.

93

The sampling algorithm is run on the CAD surface, and the resulting locations of the sample points are, then, shifted to their true locations on the deformed surface, and the CMM random measurement error component is added. Simulating the CMM measurement error component takes place through adding a random value δ_m to each of the computed coordinates for each sample point. The value of δ_m is computed to randomly lie between -0.004 mm., and +0.004 mm, as per the calibration chart of the DEA Mistral CMM (IMS Centre, University of Windsor). The selection of the random measurement error is based on a 6σ uniform distribution, with -0.004 mm. located at -3σ, and +0.004 mm. at $+3\sigma$, respectively.

The sampled points, having the measurement error added, feature two sources of error, the form error, and the measurement error. Let $P_{i,j}=(X_{i,j},Y_{i,j},Z_{i,j})$ be an arbitrary sample point, $P_{i,j}\in S$, where S is the sample points grid as calculated by the sampling algorithm.

$$S = \begin{bmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,N} \\ P_{2,1} & P_{2,1} & \cdots & P_{2,N} \\ \cdots & \cdots & \cdots & \cdots \\ P_{M,1} & P_{M,2} & \cdots & P_{M,N} \end{bmatrix}$$

(4.2)

$i \in \{1, M\}, \quad j \in \{1, N\}$

where:

i = integer index marking the points sampled in the u-direction.

j = integer index marking the points sampled in the v-direction.

M = size of sample points' grid in u-direction.

N = size of sample points' grid in v-direction.

The actual location of that point on the surface due to the form error is

$$P^d_{i,j} = (X^d_{i,j}, Y^d_{i,j}, Z^d_{i,j}) \quad ,$$

where:

$$X^d_{i,j} = X_{i,j} + \delta X_{i,j}, \quad Y^d_{i,j} = Y_{i,j} + \delta Y_{i,j}, \quad Z^d_{i,j} = Z_{i,j} + \delta Z_{i,j} \qquad (4.3)$$

The random CMM measurement error is, then, added to $P^d_{i,j}$. The final location of

$P_{i,j}$ is:

$$P^m_{i,j} = (X^m_{i,j}, Y^m_{i,j}, Z^m_{i,j}) \qquad (4.4)$$

where:

$$X^m_{i,j} = X_{i,j} + \delta X_{i,j} + \delta_m, \quad Y^m_{i,j} = Y_{i,j} + \delta Y_{i,j} + \delta_m, \quad Z^m_{i,j} = Z_{i,j} + \delta Z_{i,j} + \delta_m \qquad (4.5)$$

The measurement points $P^m_{i,j}$ are fitted to a NURBS surface that represents the

substitute geometry. The deviation between the substitute geometry, and the CAD surface

is, then, calculated. The value of the maximum deviation resulting from the application of

each algorithm is compared against the similar value that results from the application of

Genetic Algorithms (GA's) to the sampling problem.

4.2. Results Of Sampling Algorithms Simulation

This section demonstrates the results of the application of the sampling algorithms to

the NURBS surfaces described in the previous section. For each surface, the different

sampling algorithms, including the uniform sampling pattern introduced by Yau and

Menq (1992) are applied, and compared. The different sampling strategies are compared in terms of the maximum deviation between the substitute geometry and the CAD surface.

4.2.1. Application of sampling algorithms to surface A

Surface A (Fig. 4.2) is a simple surface, whose features tend to be flat. There is no relatively large change in the surface curvature values. The values of the knot vectors, surface degrees, CFactor, and PFactor of surface A are given in table 4.1. The sampling algorithms were applied to distribute a 6×6 grid of points on the surface. The results are illustrated in figure 4.4. Each sampling plan is associated with the maximum deviation between the substitute geometry, and the CAD geometry resulting from the application of that plan. Since surface A is a simple surface, with no significant changes in curvature, The results of applying the different sampling algorithms do not vary significantly.

Table 4.1. Features Of Surface A

No. Of Control Points	U-direction	5
	V-direction	6
Knot Vectors	U	[0, 0, 0, 0, 0.3, 1.0, 1.0, 1.0, 1.0]
	V	[0, 0, 0, 0, 0.34, 0.62, 1.0, 1.0, 1.0, 1.0]
Surface Degrees	U-direction	p = 3
	V-direction	q = 3
Complexity Factors	CFactor	0.03
	PFactor	8.0

Equi-parametric Sampling: 12 μm

Patch Size Based Sampling: 10μm

Curvature Based Sampling: 8 μm

Hybrid Sampling: 9 μm

Uniform Sampling: 8 μm

Genetic Algorithms: 8 μm
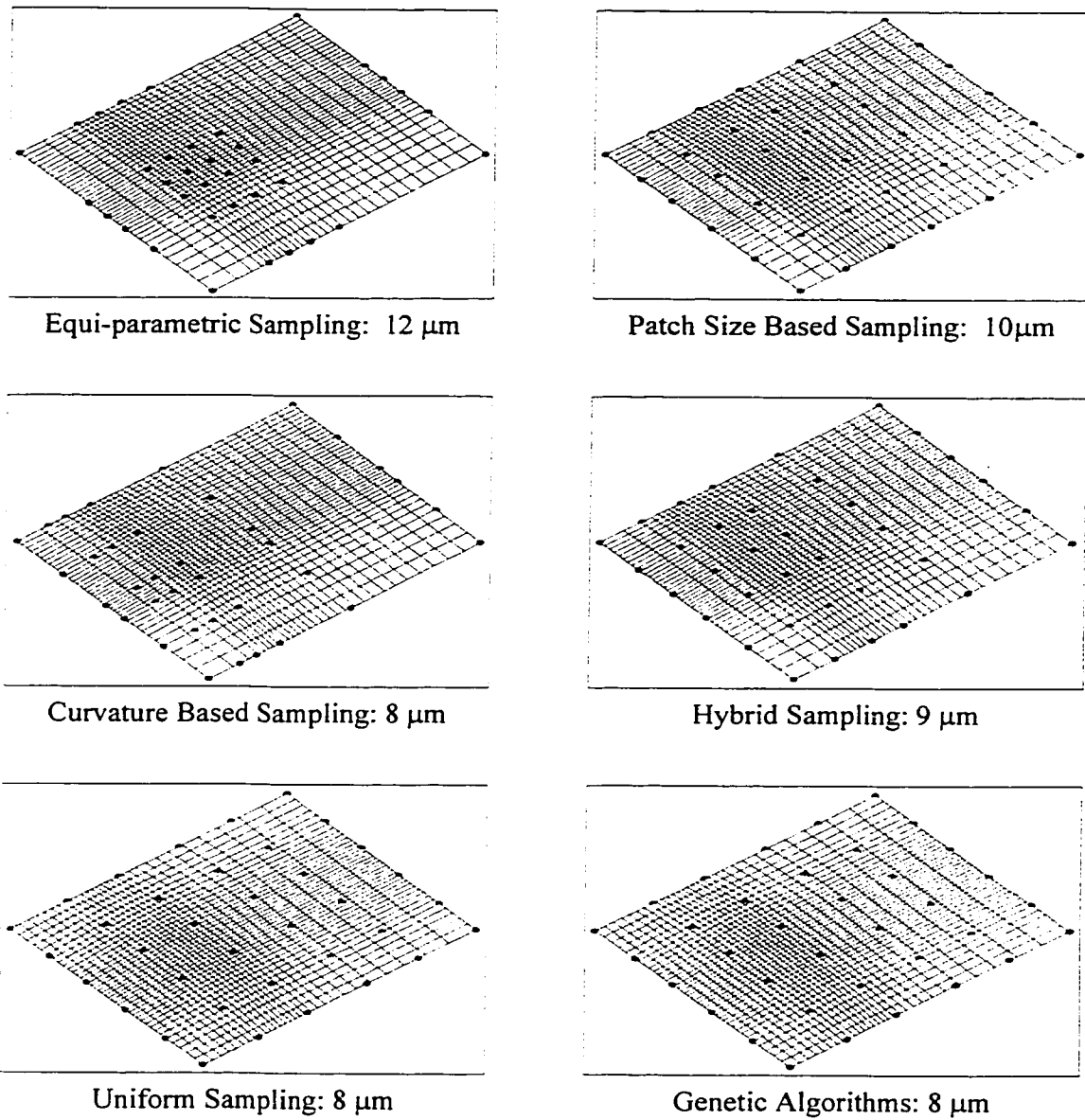
Figure 4.4. Results Of The Application Of Sampling Algorithms To Surface A

97

4.2.2. Application of sampling algorithms to surface B

More complexity has been added to surface B (Fig. 4.2). Moving the control points, a simple convexity was added to the surface. The change in curvature remains insignificant in surface B. However, the size of the surface patches varies significantly. This variation is due to the surface knot vectors values. The values of the features of surface B are given in table 4.2.

Table 4.2. Features Of Surface B

No. Of Control Points	U-direction	5
	V-direction	6
Knot Vectors	U	[0, 0, 0, 0, 0.17, 1.0, 1.0, 1.0, 1.0]
	V	[0, 0, 0, 0, 0.2, 0.4, 1.0, 1.0, 1.0, 1.0]
Surface Degrees	U-direction	p = 3
	V-direction	q = 3
Complexity Factors	CFactor	0.04
	PFactor	14.0

Results of the application of the sampling algorithms to sample a grid of 6×6 sample points are illustrated in figure 4.5.

The results of sampling surface B show significance in the effect of the surface patch size on the results of the sampling algorithms. Equi-parametric sampling algorithm lead to a sampling plan where the majority of the sample points are concentrated in a certain area of the surface. This is due to the non-uniform nature of the knot vectors.

98

Equi-parametric sampling: 16 μm

Patch Size Based Sampling: 9 μm

Curvature Based Sampling: 15 μm

Hybrid Sampling: 13 μm

Uniform Sampling: 11 μm

Genetic Algorithms Sampling: 8 μm
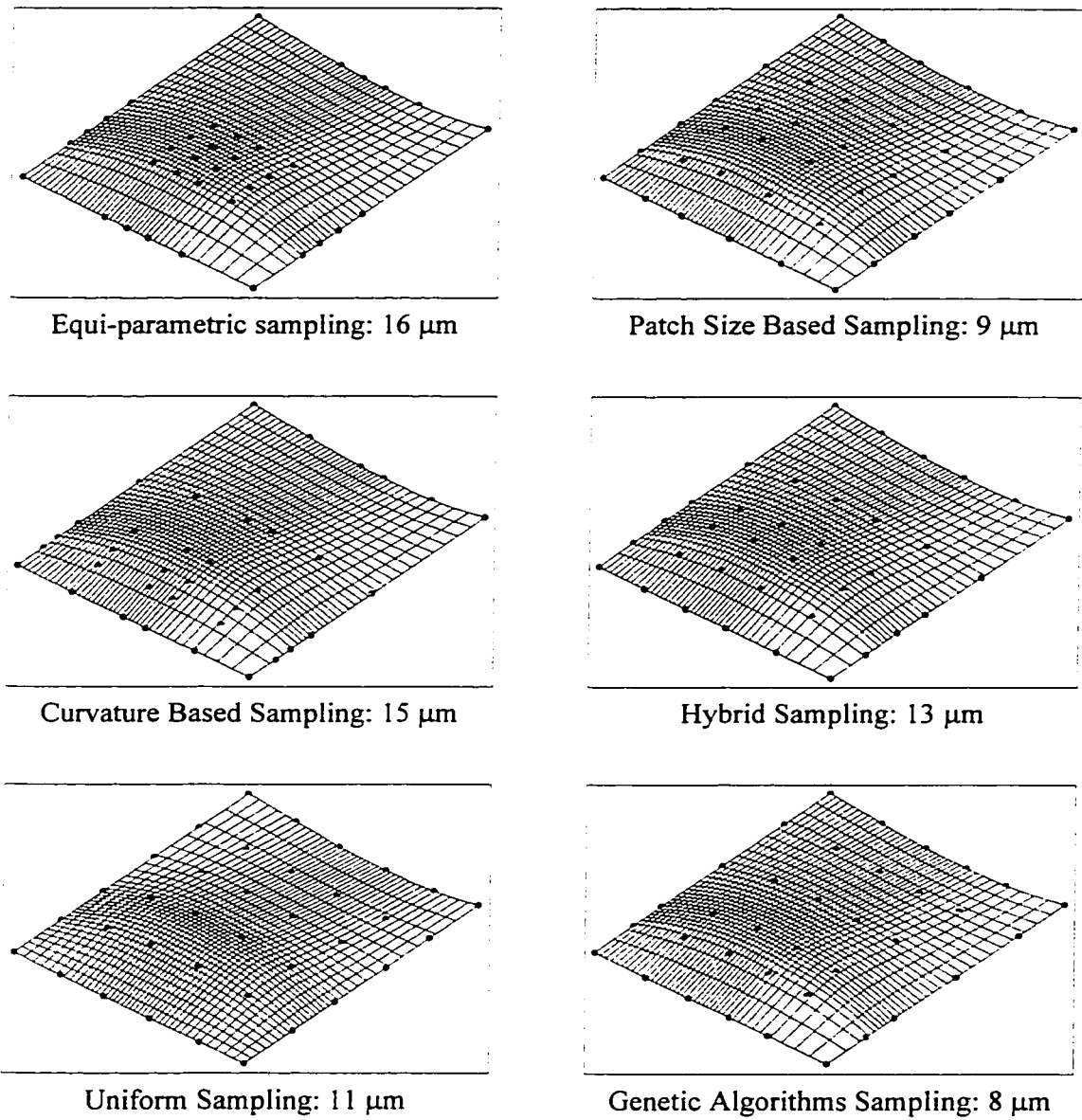
Figure 4.5. Results Of The Application Of Sampling Algorithms To Surface B

Patch size based sampling produced a sampling plan that emphasized the surface significant feature (patch size variation) and, therefore, lead to a smaller maximum deviation between the substitute geometry, and the CAD surface.

Curvature based sampling algorithm concentrated the majority of the sample points in the surface zone where the change in the curvature value is large. However, since surface B tends to be flat in most of its zones, concentrating the sample points where the curvature change is large did not reduce the sampling plan accuracy significantly.

The hybrid sampling approach combined the latter two plans, thus emphasizing the effects of both the patch size change, and the curvature change respectively. As illustrated in figure 4.5, applying the hybrid approach pulled two rows of sample points towards the surface zone where the curvature changes are higher. This increased the maximum deviation between the substitute geometry, and the CAD data.

The uniform sampling pattern uniformly covered surface B, and produced a plan that is close to the patch size based plan in its accuracy. This is due to the insignificant curvature variation of surface B.

Genetic Algorithms (GA's) optimization of the sampling process produced a sampling plan with the minimum value of the maximum deviation between the substitute geometry and the CAD surface. However, this plan is not of significant difference from the plans produced by the other heuristic methods.

4.2.3. Application of sampling algorithms to surface C

Surface C (Fig. 4.2) is similar to surface B in that the curvature values do not change significantly throughout the surface. The only difference between surfaces C, and B is in the direction of the curvature change in both surfaces. Surface C is concave, where surface B is convex. The construction features of surface C are listen in table 4.3.

Table 4.3. Features Of Surface C

No. Of Control Points	U-direction	5
	V-direction	6
Knot Vectors	U	[0, 0, 0, 0, 0.17, 1.0, 1.0, 1.0, 1.0]
	V	[0, 0, 0, 0, 0.19, 0.397, 1.0, 1.0, 1.0, 1.0]
Surface Degrees	U-direction	p = 3
	V-direction	q = 3
Complexity Factors	CFactor	0.05
	PFactor	13.0

The same sample grid size (6×6) is applied in this case too. The sampling algorithms were applied to surface C, and their results are illustrated in figure 4.6.

The similarity between surface C, and surface B (in terms of complexity) lead to similar behavior of the sampling algorithms when applied to both surfaces (as can be inferred from figure 4.6).

Patch size based sampling remains the second to GA's in producing an efficient

sampling plan. Nevertheless, the optimization of the sampling plan in both surfaces did not significantly change the magnitude of the maximum deviation between the nominal surface, and the substitute geometry (maximum deviation was reduced by 10%-11% in both cases).

Equi-parametric Sampling: 16 μm

Patch Size Based Sampling: 10 μm

Curvature Based Sampling: 12 μm

Hybrid Sampling: 11 μm

Uniform Sampling: 10 μm

Genetic Algorithms Sampling: 9 μm

Figure 4.6. Results Of The Application Of Sampling Algorithms To Surface C

4.2.4. Application of the sampling algorithms to surface D

Surface D (Fig. 4.2) is more complex than the latter two surfaces. This higher level of complexity results from the values of the knot vectors, and curvature changes, which is reflected in the values of CFactor, and PFactor for surface D. Table 4.4 shows the values of the features of surface D.

Table 4.4. Features Of Surface D

No. Of Control Points	U-direction	5
	V-direction	6
Knot Vectors	U	[0, 0, 0, 0, 0.09, 1.0, 1.0, 1.0, 1.0]
	V	[0, 0, 0, 0, 0.1643, 0.3512, 1.0, 1.0, 1.0, 1.0]
Surface Degrees	U-direction	p = 3
	V-direction	q = 3
Complexity Factors	CFactor	0.06
	PFactor	18.0

The curvature values vary between different surface zones. However, there is no sharp curvature changes in surface D. Results of applying the sampling algorithms to sample a 6×6 sample grid are demonstrated in figure 4.7.

The results demonstrated in figure 4.7 illustrate the decline in the ability of the heuristics to produce a reliable sampling plan. Equi-parametric sampling does not have any sensitivity to the surface complexity and, therefore, produced an inaccurate result. Patch size based sampling distributed the sample points so that the bigger patches

receive larger shares of sample points. Curvature based sampling concentrated the majority of the sample points in the surface zone with the highest curvature values. Uniform sampling uniformly distributed the sample points on the surface regardless of its complexity. Hybrid sampling, in this case, produced results that are better than all other heuristics. This is due to the fact that both surface complexity features, i.e., patch size variation, and curvature variation were taken into account. Genetic Algorithms based sampling produced the most accurate plan. However, the difference between hybrid sampling, and Genetic Algorithms sampling is not significant. This is due to the smoothness of the surface curvature changes (i.e., no sudden, sharp change in surface curvature). Increasing the complexity of the surface features may dramatically change the sampling results. The next two subsections demonstrate the application of sampling algorithms to more complex surfaces.

Equi-parametric Sampling: 25 μm

Patch Size Based Sampling: 19 μm

Curvature Based Sampling: 32 μm

Hybrid Sampling: 14 μm

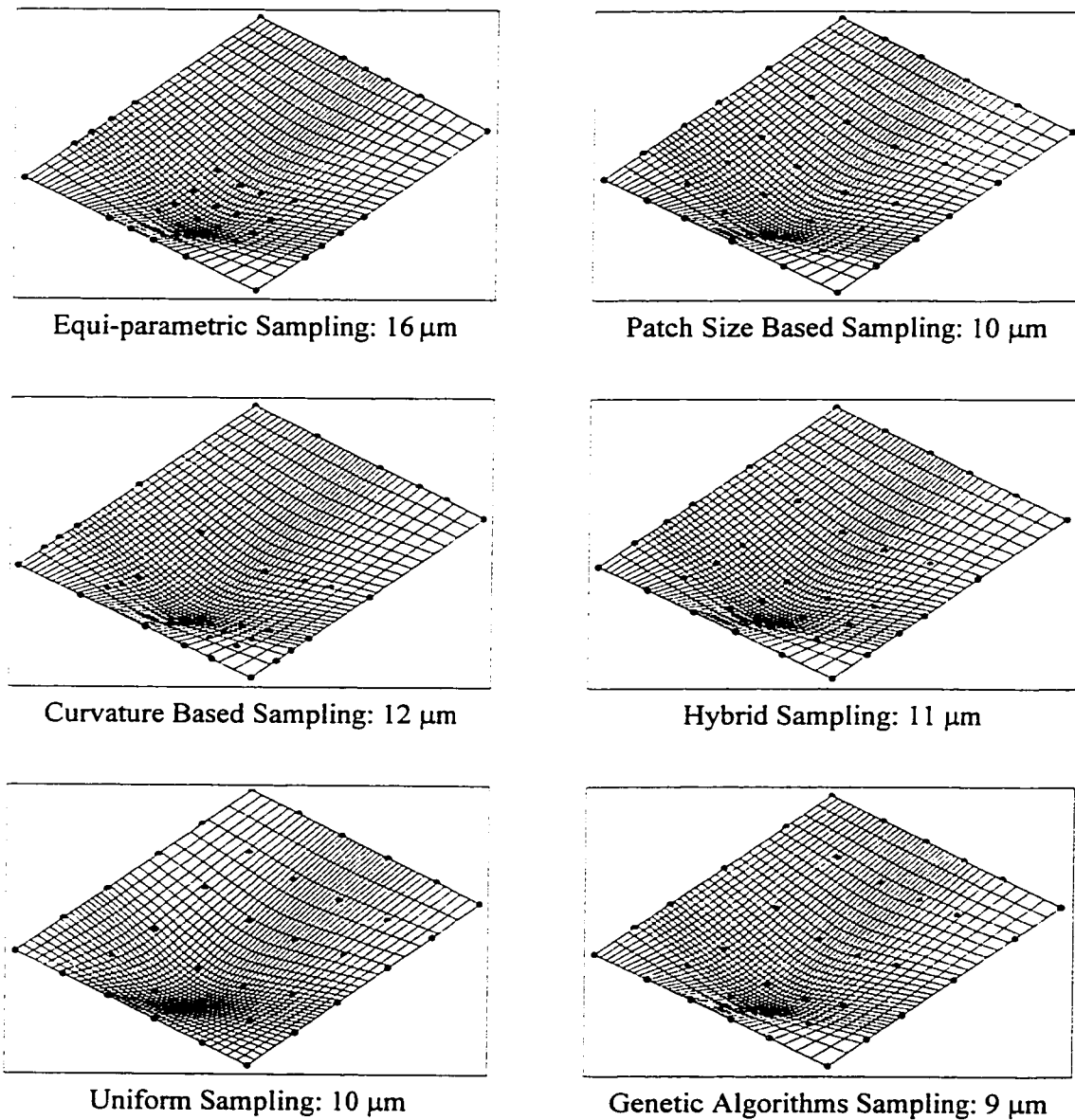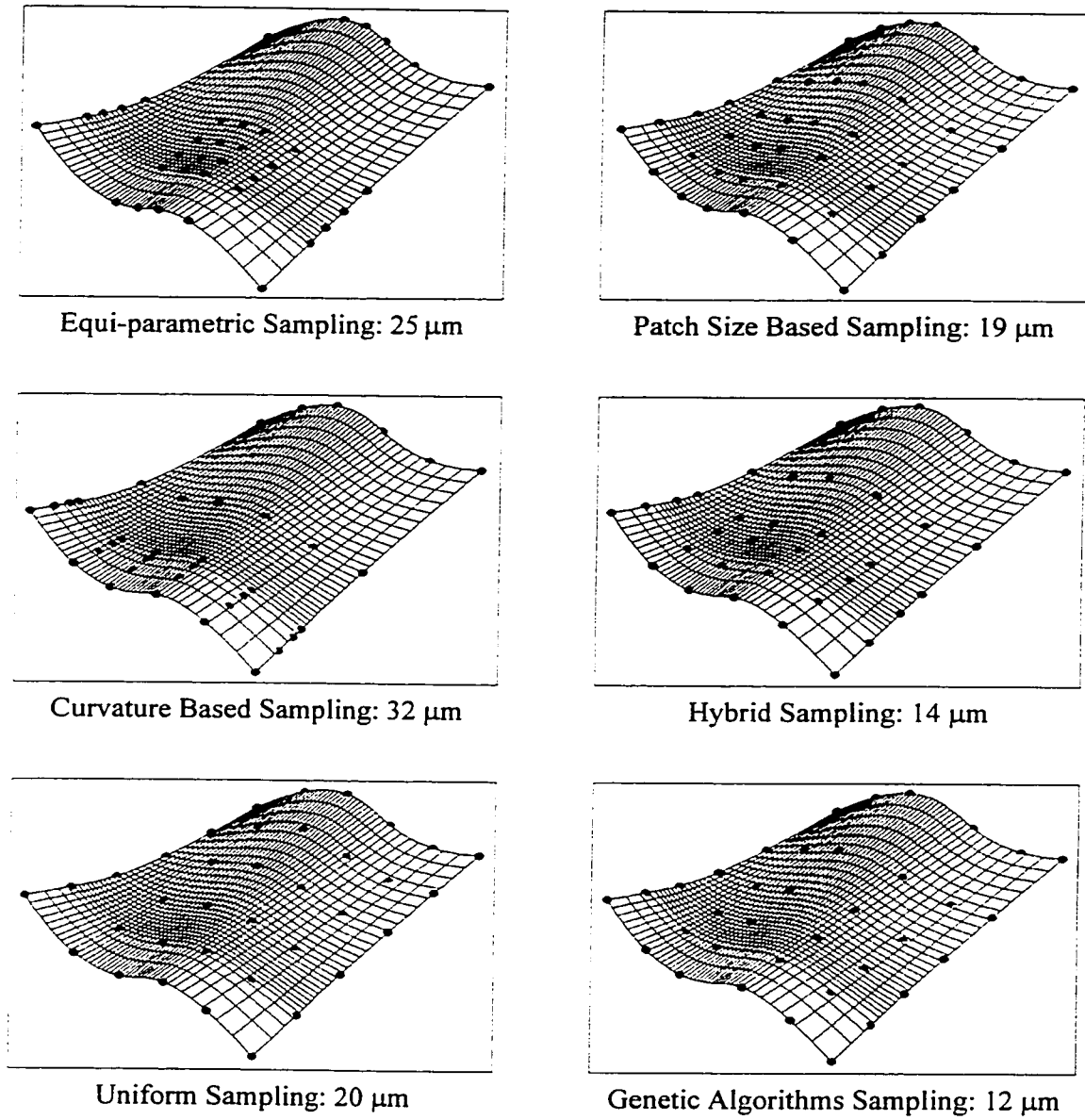Uniform Sampling: 20 μm

Genetic Algorithms Sampling: 12 μm

Figure 4.7. Results Of The Application Of Sampling Algorithms To Surface D

4.2.5. Application of sampling algorithms to surface E

Surface E (Fig. 4.2) is a more complex surface. Its patches vary in their sizes, and mean curvature values as well. The surface includes a number of zones where the surface curvature sharply changes. The surface features are summarized in table 4.5.

The most significant complexity feature of surface E is the variation in surface patches' mean curvature values. This has an effect on the sampling algorithms applied to the surface. A larger sample size was needed in case of surface E due to its higher complexity than surfaces A through D. Sampling algorithms were applied to sample a grid of 6×7 sample points from the surface. The results of the application of the sampling algorithms to surface E are illustrated in figure 4.8.

Table 4.5. Features Of Surface E

No. Of Control Points	U-direction	5
	V-direction	6
Knot Vectors	U	[0, 0, 0, 0, 0.2246, 1.0, 1.0, 1.0, 1.0]
	V	[0, 0, 0, 0, 0.1403, 0.3981, 1.0, 1.0, 1.0, 1.0]
Surface Degrees	U-direction	p = 3
	V-direction	q = 3
Complexity Factors	CFactor	2.3
	PFactor	14.0

As can be concluded from figure 5.8, the accuracy of most of the sampling algorithms, except that of the curvature based sampling algorithm, declined. The reason is the sharp

106

curvature changes that the surface has. Curvature based sampling came next to Genetic

Algorithms based optimization in providing a reliable solution to the sampling problem.
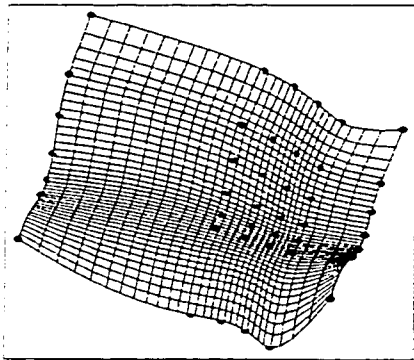
Equi-parametric Sampling: 35 μm   Patch Size Based Sampling: 25 μm
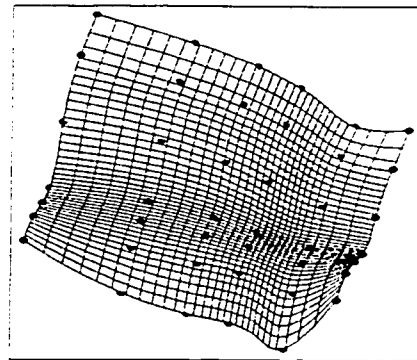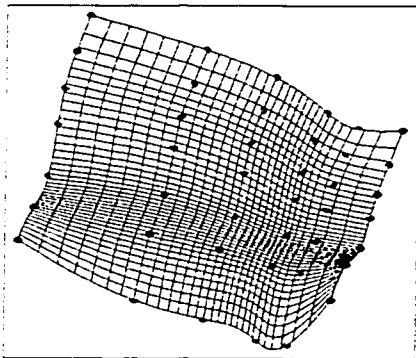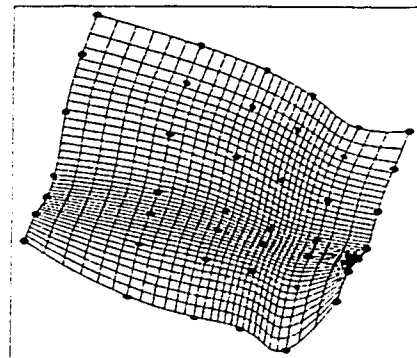
Curvature Based Sampling: 15 μm   Hybrid Sampling: 19 μm

Uniform Sampling: 29 μm   Genetic Algorithms Sampling: 12 μm

Figure 4.8. Results Of The Application Of Sampling Algorithms To Surface E

Hybrid sampling came next to curvature based sampling. The difference between hybrid sampling, and curvature based sampling is not large, but the gap is wider between hybrid sampling, and the other sampling algorithms, especially equi-parametric sampling.

Up to this point, heuristics produced satisfactory results. When the surface complexity increases, such that the variation of surface patch sizes increase, and the surface curvature values in change within smaller patches, the heuristics may fail to find a solution to the sampling problem. An example to such case will be demonstrated in the next subsection.

4.2.6. Application of sampling algorithms to surface F

Surface F (Fig. 4.2) features more complexity than the former five surfaces. The number of control points are larger in surface F and, therefore, the knot vectors are more complex. This is due to the dependency of the number of knots on the number of control points. The surface curvature values add more source of complexity, for they sharply change at different surface zones. The features of surface F are listed in table 4.6.

Due to its higher complexity, surface F was sampled using a larger sample size. A sample grid of 8×8 points was sampled using the sampling algorithms described in the previous chapter. The results of the sampling processes are illustrated in figure 4.9.

Table 4.5. Features Of Surface F

No. Of Control Points	U-direction	6
	V-direction	6
Knot Vectors	U	[0, 0, 0, 0, 0.1704, 0.4018, 1.0, 1.0, 1.0, 1.0]
	V	[0, 0, 0, 0, 0.1942, 0.3568, 1.0, 1.0, 1.0, 1.0]
Surface Degrees	U-direction	p = 3
	V-direction	q = 3
Complexity Factors	CFactor	9.0
	PFactor	14.0

The results illustrated in figure 4.9 show how the effect of surface complexity can dramatically change the results of the sampling algorithms. In the previous five cases, the maximum deviation resulting from optimum sampling was better than the highest accurate heuristic by (0%-20%), whereas in case of surface F, GA's surpassed the highest accurate heuristic (Patch size based sampling) by 36%. It is intuitive that the higher the surface complexity, the higher this ratio is. In such a complex shape like surface F, Genetic Algorithms based sampling succeeded to produce a reliable sampling plan. Each of the other methods focused on one surface feature, and ignored the others. The only heuristic methodology that is able to produce an accurate plan is the hybrid approach, through choosing the correct weights for each of the patch size, and mean curvature based methodologies. However, this is a tedious iterative operation that involves an enormous amount of calculations. This is due to the need of trial and error of different values for the weights of the patch size, and the patch mean curvature methods. Instead, Genetic

Algorithms based sampling provides such a reliable sampling plan.
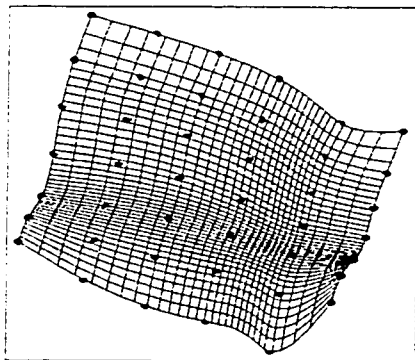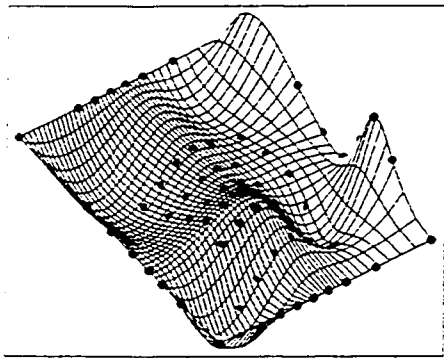

Equi-parametric Sampling: 48 μm.
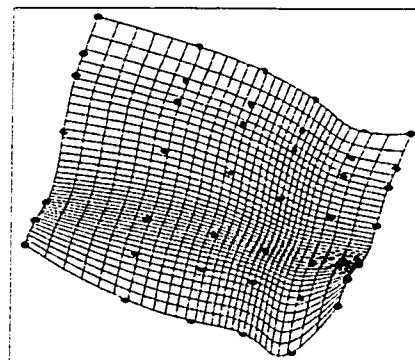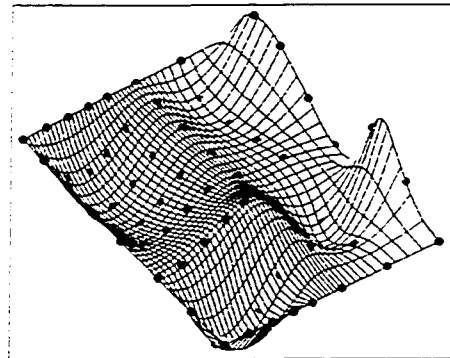
Patch Size Based Sampling: 25 μm.
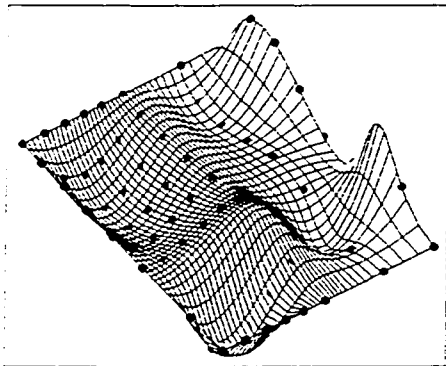
Curvature Based Sampling: 40 μm.
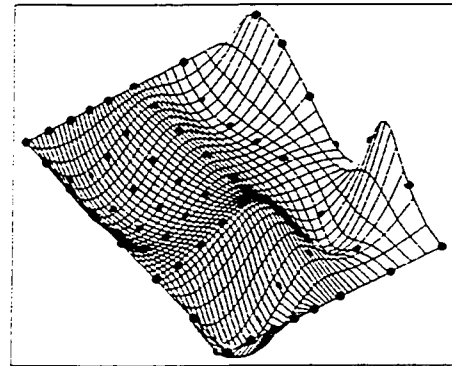
Hybrid Sampling: 33 μm.
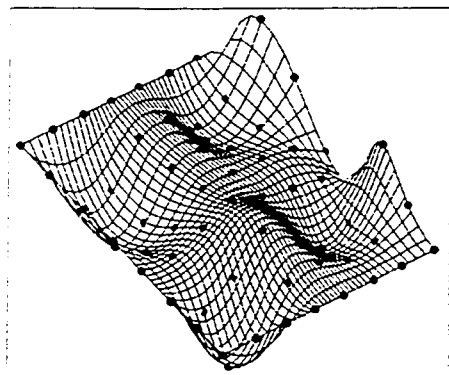
Uniform Sampling: 30 μm.
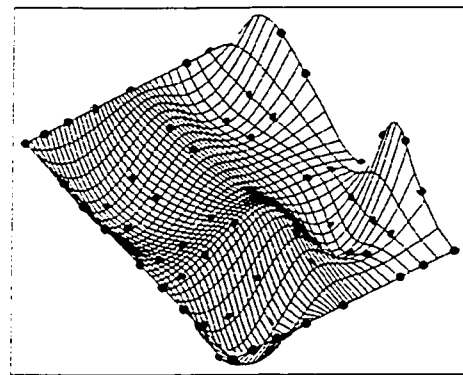
Genetic Algorithms Sampling: 16 μm.

Figure 4.9. Results Of The Application Of Sampling Algorithms To Surface F

4.3. Analysis Of Simulation Results

The simulation process is carried out on a Pentium III - 733 MHz CPU. The time (in seconds) spent by the algorithms to compute a sampling plan for each of surfaces A through F was coded. Table 4.7 displays the results of the simulation study.

Table 4.7. Results Of Simulation Study

	Surface A		Surface B		Surface C		Surface D		Surface E		Surface F	
	Dev.	Time	Dev.	Time	Dev.	Time	Dev.	Time	Dev.	Time	Dev.	Time
Equi-parametric	12	10	16	11	16	12	25	13	35	14	48	15
Patch Size Based	10	15	9	17	10	18	19	19	25	21	25	22
Curvature Based	8	20	15	22	12	24	32	26	15	28	40	30
Hybrid Sampling	9	40	13	44	11	48	14	51	19	55	33	60
Uniform Sampling	8	9	11	10	10	11	20	12	29	12	30	13
Genetic Algorithms	8	6000	8	6180	9	6365	12	6556	12	6753	15	6956

The results of the simulation study was coded in a graphical format for simplicity. The behavior of each methodology is compared according to the surface complexity. The comparison criterion used is the algorithm accuracy, expressed in the formula:

$$A_{i,j} = \frac{Dev_{GA,j}}{Dev_{i,j}}$$ \hfill (4.16)

where:

$A_{i,j}$ = Accuracy of the ith algorithm when applied to the jth surface.

$Dev_{GA,j}$ = Maximum deviation between substitute geometry, and CAD surface when Genetic Algorithms sampling is applied to sample the jth surface.

$Dev_{i,j}$ = Maximum deviation between substitute geometry, and CAD surface when the ith algorithm is applied to sample the jth surface.

111
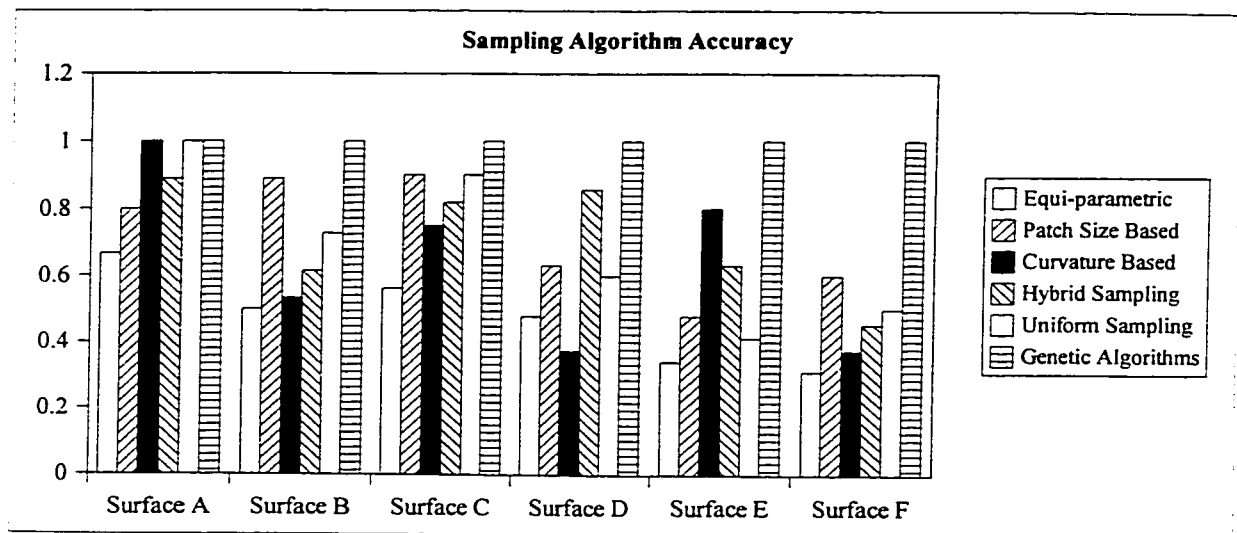
Figure 4.10. Comparison Of Sampling Algorithms Applied In Simulation Study

The value $A_{i,j}$ was chosen as a measure for the sampling algorithms accuracy, for it presents how the sampling algorithms behave when applied to surfaces of different complexity levels. Figure 4.10 illustrates the graphical representation of the results of the simulation study presented in this chapter. The trend of the various graphs in figure 4.10 show that the results obtained from the sampling algorithm vary according to the surface complexity. For simple surfaces, the results of the different sampling algorithms, including GA's based sampling do not seem to have significant differences. As the complexity of the surface increases, the variation in sampling results increases.

The surface complexity features are patch sizes, and patch mean curvatures, respectively. When both complexity features are not significant (as in surface A),

sampling algorithms produce efficient plans.

In cases where patch sizes are a key issue, patch size based sampling proved to be a reliable methodology. Examples on such case are surfaces B, and C. In this case, other sampling algorithms did not perform as efficiently as patch size based sampling. This is due to the focus of each algorithm on a certain surface feature, i.e., curvature, and parameterization. Uniform sampling came next to patch size based sampling. This is because both surfaces B, and C do not feature sharp curvature changes. The attempt to combine patch size sampling with curvature based sampling using the hybrid sampling algorithm produced a less accurate plan. This is due to introduction of the effect of patch mean curvature to the sampling plan.

In cases where sharp curvature changes are encountered, as in surface E, curvature based sampling produced the most accurate plan among all the heuristic methods. The complex shape of surface E made the other sampling algorithms, excluding Genetic Algorithms, fail to produce reliable sampling plans. This is due to their insensitivity to the surface complex features. Hybrid sampling, nevertheless, could have been utilized more efficiently, having the weights given to each of its methodologies chosen properly. This process, however, involves a tedious amount of trial-and-error selections of the weights. This is obvious in case of surface D, where the surface features involve higher

complexity, in terms of its non uniform knot vectors, and curvatures change.

When the surface complexity is high, the task of sampling becomes very difficult, and using any heuristic method may not lead to a satisfactory solution. In such cases, Genetic Algorithms prove to be superior to the heuristic methodologies. An example to such cases is surface F. However, a trade off to the high accuracy of Genetic Algorithms based optimum sampling is the high computation time, as can be inferred from table 4.1. But, since the sampling process is usually carried out off line, this cost can be justified. Besides, faster algorithms for GA's based sampling can be implemented.

4.4. Case Studies

This section presents the application of the sampling algorithms to selected CAD models for various products featuring free form surfaces. Flowchart of the approach used to apply sampling algorithms to the case studies is presented, as well as the results of the sampling algorithms. Results of running the sampling algorithms on selected surfaces of the the case studies are also presented.

4.4.1. Case studies pre processing

Case studies were collected from different sources, and put subject to the sampling algorithms. The case studies were imported in IGES format. Since the sampling algorithms are not provided with the capability to read IGES data, it was necessary to

114

translate the model files to a format that easily provide the data needed to build the NURBS surfaces within the sampling algorithms. This data is the control points' locations, the surface degrees, and the surface knot vectors. To achieve this goal, the IGES files were first imported to Rhinoceros 2.0 CAD software (Robert McNeel and Associates, Inc.), the surfaces were manipulated, and exported in the Alias Wave Front (OBJ) format. The latter is a simple text format, from which the necessary data to construct the NURBS surface were obtained.

Manipulating the CAD model consists of subdividing the CAD model into a set of NURBS surfaces to simplify the calculations performed by the sampling algorithms, performing operations such as stitching, knot insertion, and knot removal, on the model surfaces to keep the continuity of the CAD model.

Figure 4.11 presents the steps followed to perform sampling on the selected case studies. Figures 4.12, and 4.13 illustrate samples of the IGES surfaces, and Wave Front OBJ (Alias|Wavefront, Inc.) formats respectively.

4.4.2. Selected CAD models

Four different CAD models, with varying complexities, were chosen to be used as case studies. NURBS surfaces were extracted from each model, and used as examples to

demonstrate the application of the sampling algorithms described in this thesis. For each surface, the sample points grid size was fixed, and the sampling algorithms have been applied. The results shown in this section represent the outcome of applying the sampling algorithms to different Figure 4.14 illustrates the four CAD models selected for this study, and the NURBS surfaces extracted from each model.

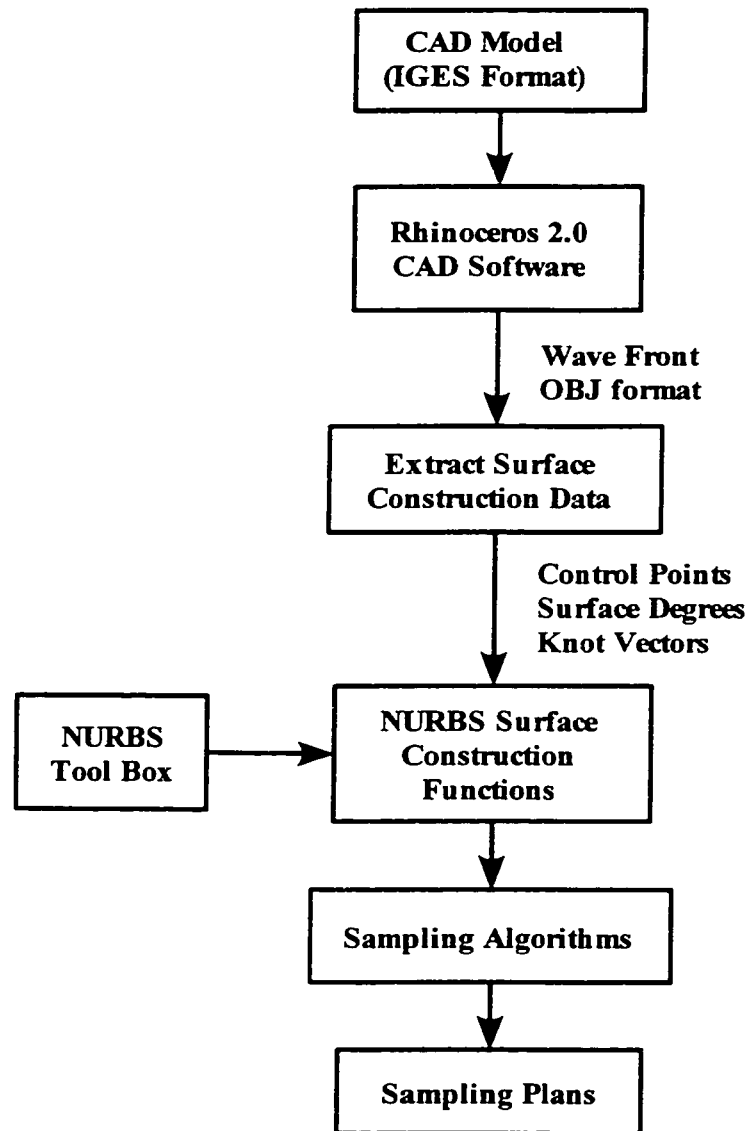Figure 4.11. Pre Processing Of IGES Data

```
WRAP                                              S    1
B-spline surfaces.                                S    2
Number of Patches: 61                             S    3
,,18Hgeomagic Shape 2.0,                               G    1
22HE:\Web_files\torso.igs,                             G    2
45HRaindrop Geomagic, Inc. – geomagic Shape 2.0,            G    3
13H<unspecified>,32,38,6,38,15,13H<unspecified>,       G    4
1,1,2HIN,,0.016,                                       G    5
15H20000202.142451,                                    G    6
1e-015,0,                                              G    7
13H<unspecified>,13H<unspecified>,10,0;                G    8
    128    1        0    0    0    0    0    0     D    1
    128    0    0   309    0            Patch    1      D    2
    128  310        0    0    0    0    0    0          D    3
    128    0    0   309    0            Patch    2      D    4
    128    0    0   309    0            Patch    3      D    6
    128  928        0    0    0    0    0    0          D    7
    128    0    0   309    0            Patch    4      D    8
    128 1237        0    0    0    0    0    0          D    9
    128    0    0   309    0            Patch    5      D   10
```

Figure 4.12. Example Of IGES Format

```
v -153.506 20.7655 -102.798
v -164.128 25.7309 -86.6543
v -178.374 38.712 -55.1234
v -183.415 57.2185 -22.6252
v -183.794 69.7631 -7.3137
v -138.737 18.6483 -97.8665
v -149.567 22.8403 -81.6223
v -164.009 36.2375 -48.7789
v -168.368 57.5008 -16.7906
v -167.916 71.4379 -2.23239
v -109.258 16.3522 -83.9579
v -120.11 18.8748 -69.318
v -134.765 31.2744 -37.3374
v -136.074 53.532 -7.20769
v -132.227 66.971 5.73106
v -84.1006 18.2288 -64.0051
v -92.3231 18.7937 -53.6208
v -104.723 26.4547 -28.2547
v -104.721 41.2365 -3.81712
v -101.621 47.7304 6.08739
v -72.7229 20.2069 -52.6015
v -78.5932 19.8058 -44.613
v -88.418 23.9196 -24.0913
cstype bspline
deg 3 3
surf 0 2 0 2 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
parm u 0 0 0 0 1 2 2 2 2
parm v 0 0 0 0 1 2 2 2 2
```

Figure 4.13. Example Of Wave Front OBJ Format

Cellular phone CAD model

Fan CAD model

Camera CAD model

Spray gun CAD model
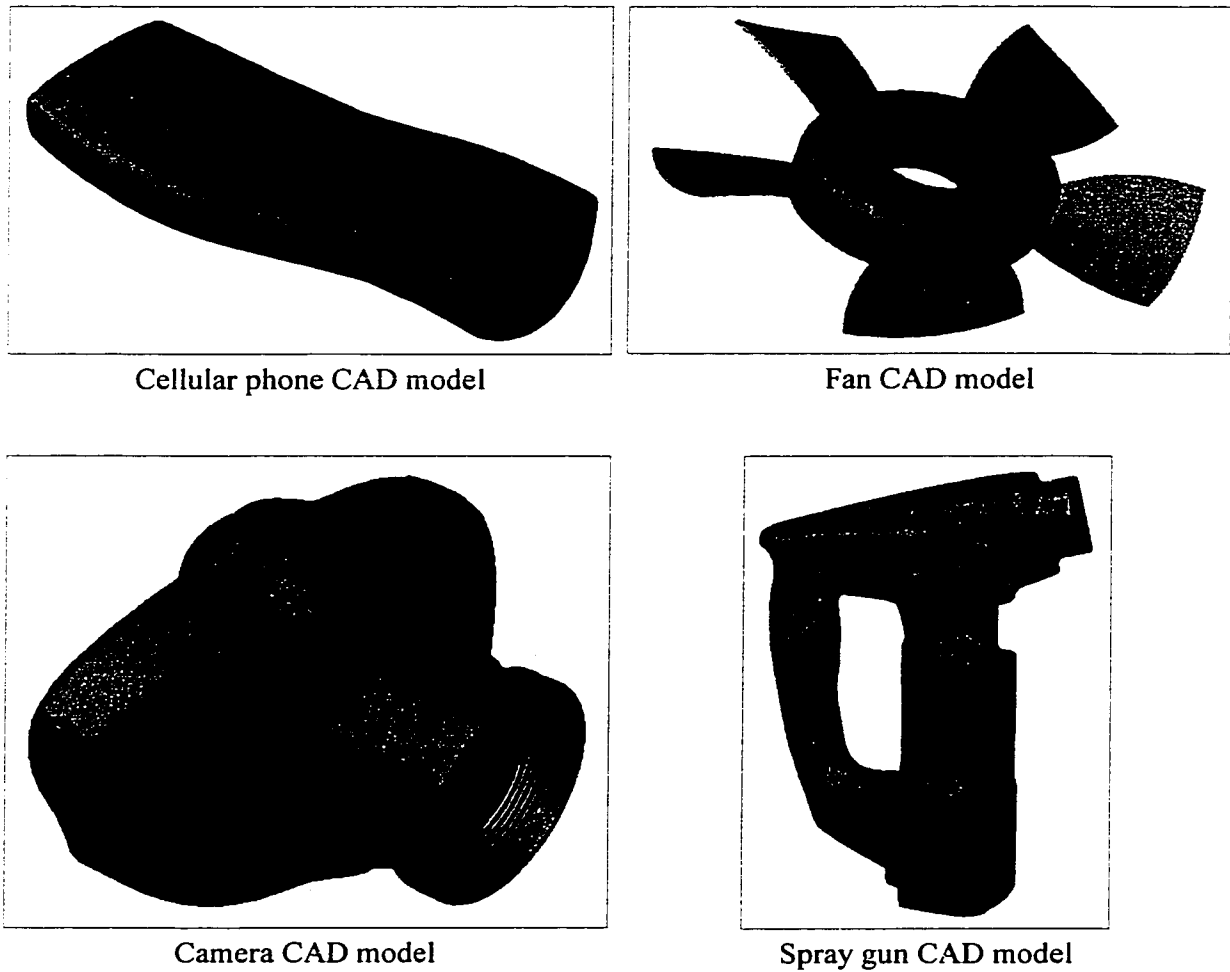
Figure 4.14. CAD Models Used For Case Studies

Figure 4.15 illustrates the surfaces extracted from the CAD models, as described in figure 4.11. The sampling algorithms have been applied to the surfaces extracted from the CAD models illustrated in figure 4.14. The deviation between the CAD surface, and the substitute geometry has been computed per each surface. Examples of the sampling of selected surfaces are illustrated in table 4.2.
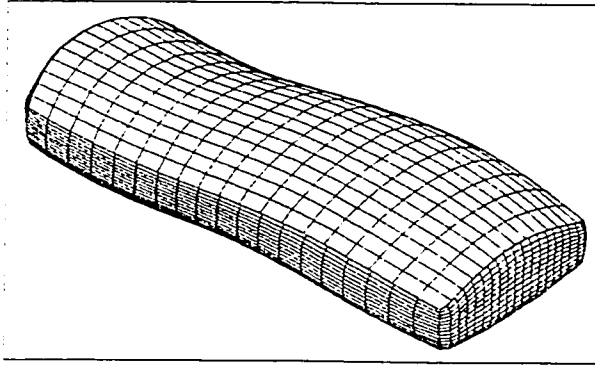
Cellular phone surfaces
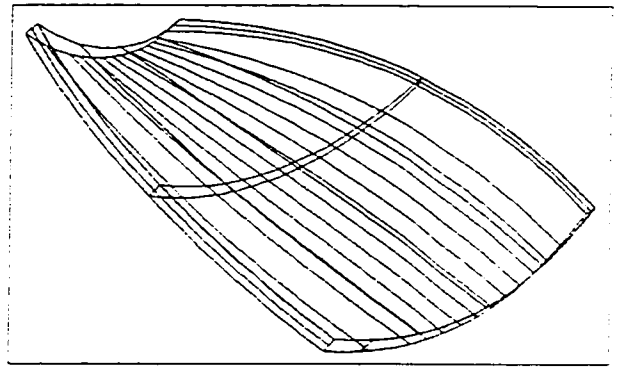
Fan blades surfaces
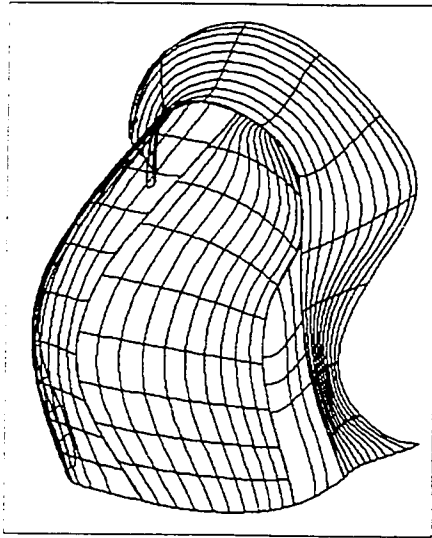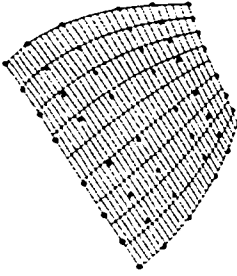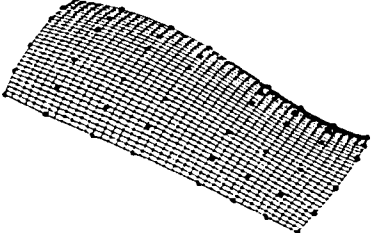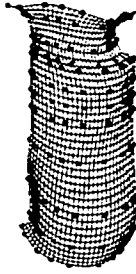
Camera surfaces
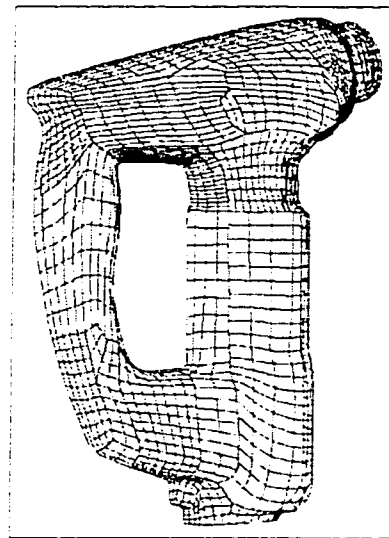
Spray gun surfaces

Figure 4.15. Surfaces Extracted From CAD Models

119

Table 4.8. Results Of Sampling Of Selected Surfaces

Surface	Sampling Plan	Method	Sample Size (Nu X Mv)	Maximum Deviation
Fan Blade		Hybrid Sampling. CWeight:PWeight = 25:75	8 x 7	0.011 mm
Phone Surface		Patch Size Based Sampling	8 x 7	0.015 mm
Camera Surface		Patch Size Based Sampling	10 x 13	0.020 mm
Spray Gun Surface A		Patch Size Based Sampling	8 x 6	0.017 mm
Spray Gun Surface B		Hybrid Sampling. CWeight:PWeight = 25:75	9 x 12	0.016 mm

4.5 Discussion

This chapter presented simulations of the sampling algorithms introduced in the last chapter. The sampling algorithms were compared to the uniform sampling algorithm reported in the literature (Yau, and Menq, 1992).

Results show that the equi-parametric sampling technique includes large approximation and cannot provide reliable sampling plans, especially in the case of the illustrated surface which includes patches with different sizes. A larger sample size should be used to reduce the maximum deviation to a value that is close to the random measurement error component.

A patch size based sampling for this surface proved to be capable of producing better sampling plans, especially when patch size variation is the main factor participating in the surface complexity. However, higher accuracy makes it necessary to use a larger sample size as well.

Mean Gaussian curvature based sampling was found to be very efficient in cases of surfaces with sharp curvature changes. However, if the sharp curvature change occurred within smaller surface patches, this methodology may produce sampling plans in which larger patches are not given sufficient share of the sample points, resulting in a larger

deviation between the fitted surface and the geometric model.

Combining the last two methodologies produces better, and more accurate sampling plans. The manner in which this methodology is implemented allows the user to assign a relative weight to each of the combined two methodologies. For instance, if the effect of relative differences of surface patches was more significant, a higher weight is assigned to the patch size based sampling. The same applies if relatively sharp curvature changes are involved. The drawback of this methodology is that it is highly dependent on the user's judgment and, therefor, is most likely to produce insufficient plans, especially where complex surfaces are involved.

The genetic algorithms (GA's) based optimization provided the most accurate plan amongst the five sampling methodologies discussed in this paper. Although it involves a lot of calculations and CPU time, this is normally done off line. If the surface to be inspected is complex, especially when high throughput involved, it is of great importance to produce a reliable and efficient sampling plan that optimally locates the sample points. In this case, GA's based sampling provides a very efficient tool.

Uniform sampling proved to efficiently produce reliable sampling plans when the surfaces are smooth, and without sharp curvature changes. As the complexity increases,

uniform sampling algorithm fails to produce an accurate sampling plan, unless the sample size was significantly increased. This is a major drawback, for it increases the inspection process time and, consequently, the inspection cost.

The integration of the sampling algorithms simulated in this chapter leads to a module for sampling of free form surfaces that may be a key component in a computer-aided tactile inspection planning system. Furthermore, there is a need for a utility to automatically select the sampling algorithm that best suites the surface being sampled. These issues are discussed in chapter 5.

To further demonstrate the capability of the sampling algorithms introduced in this chapter, they were applied to more case studies. Examples of the case studies, their pre processing, and the results of applying sampling algorithms to them were demonstrated in section 4.4.

CHAPTER FIVE

COMPUTER AIDED SAMPLING OF FREE FORM SURFACES

This chapter presents a computer-aided system for the sampling (CASampler) of the points used for inspecting free form surfaces using CMMs. CASampler was developed to remedy the shortcomings encountered in sampling free form surfaces for inspection planning. Such shortcomings include the lack of offering alternative sampling scenarios, the lack of the optimality criterion, and the dependency on the user's skills to select an appropriate sampling algorithm.

This chapter discusses the details of CASampler. Flowcharts, sample screen shots, and a step-by-step example are also included.

5.1. Overview Of CASampler

CASampler integrates different sampling algorithms, including optimum sampling, (ElKott, D., ElMaraghy, H., and Nassef, A., 1999) into a computer system with graphical user interface. It offers the user the flexibility of choosing between different sampling

algorithms, optimizing the sampling plan, or having the system automatically perform the sampling task. The sampling algorithms are, namely, equi-parametric sampling, surface patch size-based sampling, surface patch mean Gaussian curvature-based sampling, and a hybrid approach of the last two methods. Optimal sampling plans are achieved using Genetic Algorithms. In the automatic mode, the system performs a series of calculations to determine the degree of surface complexity and, based on the results, decides which sampling algorithm to apply. If the system decides to use the hybrid sampling algorithm, it computes an appropriate weight to each of the surface patch size and mean Gaussian curvature algorithms. The system then outputs the sampling points' locations and inspection probe approach vectors, the substitute geometry data (i.e. surface fitted to the sampled data), and the expected deviation of the substitute geometry from the surface CAD data. Within CASampler, a CMM random measurement error component, is simulated. Figure 5.1 shows the CASampler flow chart.

The sampling algorithms shown in figure 5.1, including the automatic sampling automatic selection of sampling algorithms, were discussed in details in chapter 3.

5.2. Structure Of CASampler

The functions developed throughout the course of this study were divided into seven main groups which are used as building blocks of CASampler. These building blocks are,

namely, the graphical user interface (GUI) group, NURBS tool box, Genetic Algorithms (GA's) tool box, sampling algorithms group, the automatic sampling algorithm selection group, the input/output (I/O) group, and the data visualization group. Figure 5.2 demonstrates the correlation between these groups, followed by a description of each.
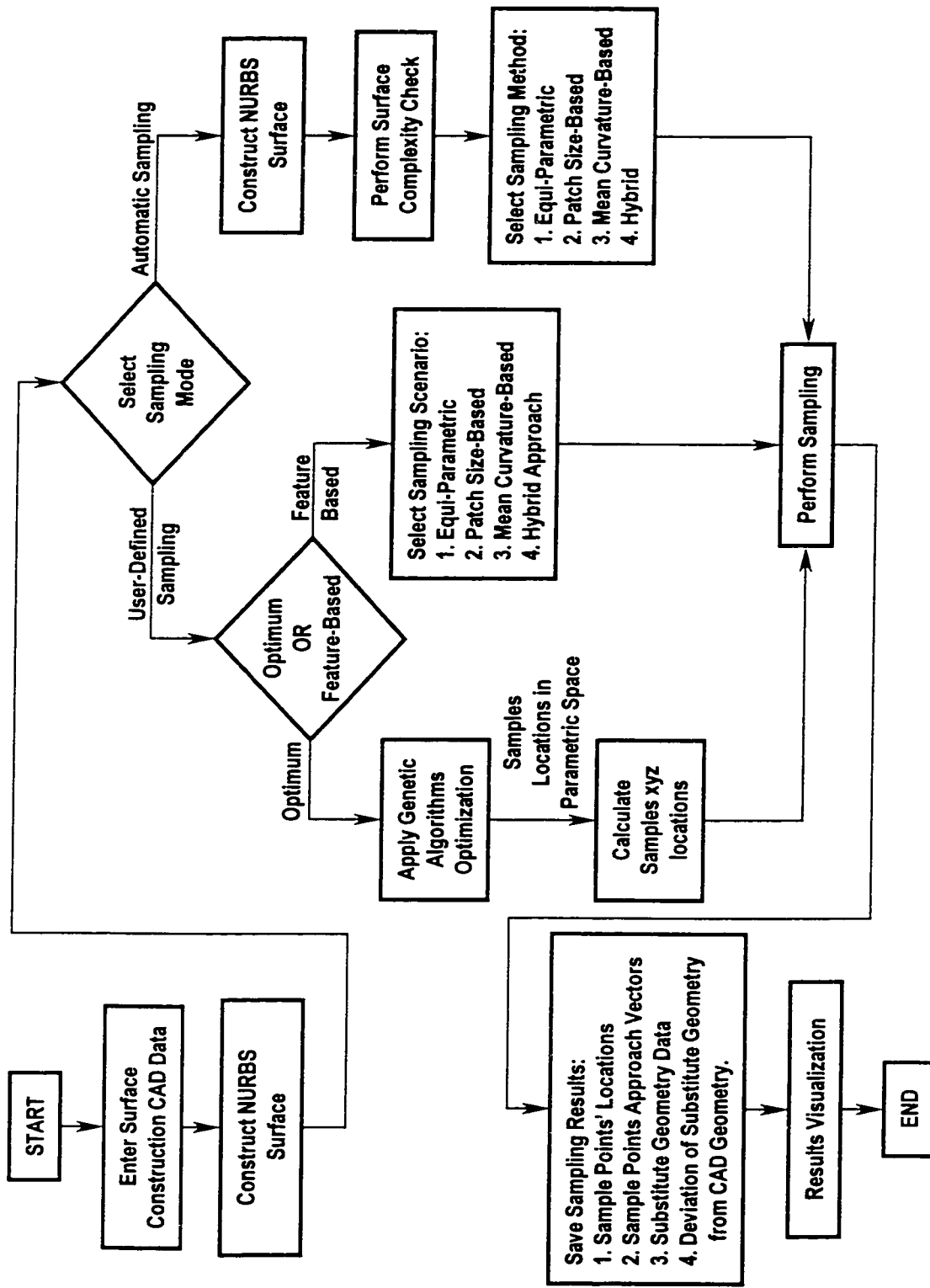
Figure 5.1. Flowchart of CASampler

127

MATLAB 5.3 (Math Works, Inc.) was used to implement the functions used within

CASampler, as well as the graphical user interface (GUI) functions.
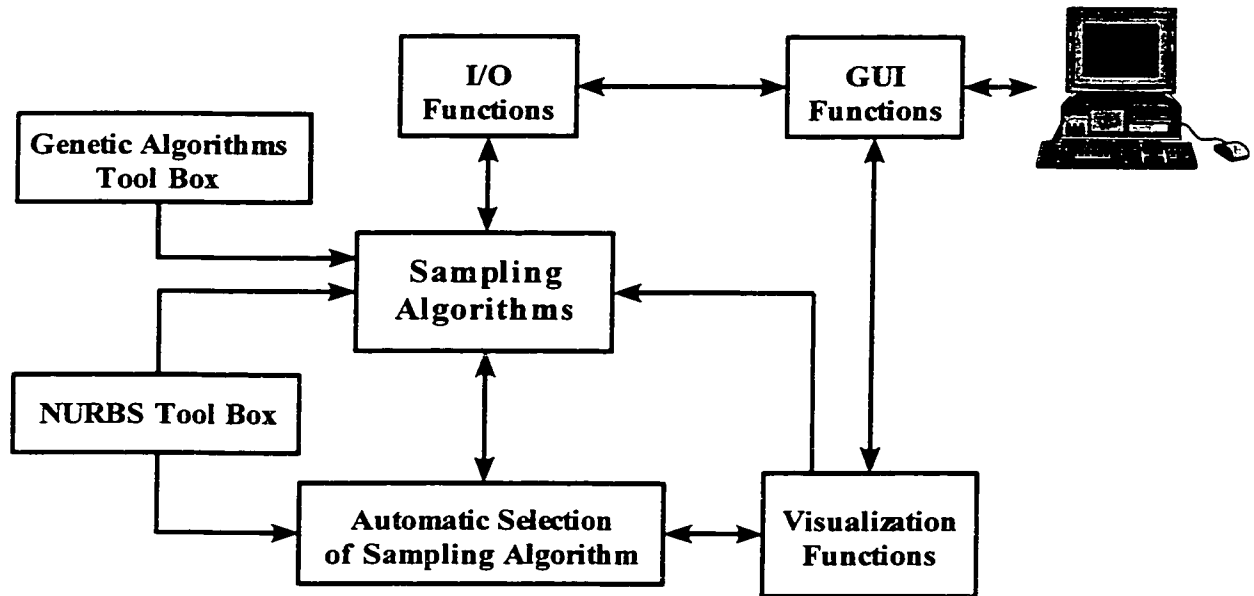
Figure 5.2. Structure Of CASampler

5.2.1. NURBS tool box

This group of functions includes all the functions needed to construct a NURBS

surface. It was built by Dr. Ashraf Nassef, IMS Centre, University of Windsor, and

extended throughout the course of this study. The NURBS algorithms used within the

NURBS tool box are based on Piegle, L. and Tiller, W. (1997), and Mortenson, M.

(1985). The NURBS tool box functions are categorized according to their functionality.

This is illustrated in table 5.1.

Table 5.1. NURBS Tool Box Functions

Category	Functionality
Control Polyhedron Constructors	Read points' 3D coordinates and construct curve/surface control polygon/polyhedron out of them.
Knot Vectors Constructors	1. Non Uniform: calculate knot vectors, given the locations of control points, and curve/surface degree(s). 2. Uniform: calculates a uniform knot vector, given the number of control points, and the surface degrees.
Basis Functions Calculators	Calculate the curve/surface rational, and B-Spline basis functions.
B-Spline/NURBS Curve and Surface Constructors	Construct B-Spline/NURBS curves/surfaces, given their control points, degrees, knot vectors, and display resolution (grid size).
Curve and Surface Fitting	Fit a B-Spline/NURBS curve/surface to a row/grid of points, given the number of control points, knot vectors, and degrees.
Curve/Surface Derivatives, Curvature, and Normal Vectors	Compute partial derivatives, curvature, and normal vectors of a curve/surface at any point.
Span/Patch Width/Size Calculators	Compute curve/surface span/patch width/size based on knot vectors.

5.2.2. Genetic algorithms tool box

The Genetic Algorithms (GA's) tool box was used as the core of the algorithm for optimum sampling of free form curves/surfaces (ElKott, D., ElMaraghy, H., and Nassef, A., 1999). It includes functions to perform the operations of solution population evaluation, selection, crossover and mutation. The tool box is designed to work with a user-specified objective function. In case of sampling of free form curves/surfaces, the objective function computes the maximum deviation between the substitute geometry,and the CAD surface. The optimality criterion is then set to minimize the value returned by

129

the objective function.

5.2.3. Input/output functions

This group of functions manages the I/O to and from the system. These functions are

divided into console I/O, and file I/O. The first receives user's input through the GUI, and

outputs data such as the program status, and error message. The latter function group

reads/write data files, creates run-time temporary files, and writes program results to files.

5.2.4. Graphical user interface (GUI) functions

This group of functions is responsible for the interaction between the user and the

system. It was designed to provide a straightforward user-friendly interface to

CASampler. Functions of this group are categorized as in table 5.2.

Table 5.2. GUI Functions

Category	Functionality
Data Output	Display the program status, and error messages, and prompts user to input data.
Data Input	Receive user to input data regarding the free form surface construction, sampling parameters, and sampling algorithms selections.
Function Calls	Call different system functions, such as NURBS construction, visualization, and sampling algorithms.

5.2.5. Visualization functions

This group of functions displays the program results. Upon user input, the

visualization group opens output data files, retrieves data, and displays graphical output.

Examples of system output are CAD surface, control points, sampling plan, substitute

geometry, and the deviation between substitute geometry, and CAD data.

5.2.6. Sampling functions group

This group represents the core of the program. Functions of this group perform the

sampling operations. The functions of this group are categorized according to the

functionality of each group, i.e., according to the sampling algorithms. The categories of

the sampling functions group are illustrated in table 5.3.

Table 5.3. Sampling Functions Group

Category	Functionality
Equi-Parametric Sampling	Read surface data, sample size, and error sources. Functions equally distribute samples on the surface in the parametric space.
Patch Size Based Sampling	Read surface data, sample size, and error sources. Functions perform sampling based on the patch sizes.
Patch Mean Curvature Based Sampling	Read surface data, sample size, and error sources. Functions compute curvature values at surface points, and perform sampling based on the surface patch mean curvature value.
Hybrid Sampling	Read surface data, sample size, and error sources. Functions combine the former two plans using weights assigned to each methodology.
GA's Based Sampling	Read surface data, sample size, and error sources. Functions call the GA's tool box to optimize the sampling plan using the objective function provided by the system user.
Miscellaneous Functions	Functions used to compute the deviation between two surfaces, the distance between two points, data preprocessing, and form error simulation.

5.2.7. Automatic selection of sampling algorithms functions group

This group of functions performs the automatic selection of sampling algorithm, as described in section 3.6. This group includes functions for surface complexity check, and assignment of sampling algorithms according to the results of the surface complexity check.

5.3. Operation Of CASampler

The program starts with the dialogue shown in figure 5.3 to input the surface construction data, i.e., control points, degrees, and display resolution. The user selects the surface degree, display resolution, and inputs the name of the data file in which the control points are stored.

The user is then prompted to choose either the automatic or the user-defined sampling modes, as illustrated in figure 5.4. If the user-defined sampling mode is selected, the user is given two options, i.e., surface feature-based sampling, and optimal sampling (figure 5.5). In surface features-based sampling, the user can choose from four sampling scenarios based on the NURBS surface properties. The choice of the surface feature-based sampling scenario is accompanied by the selection of the sample grid size. If the user selects the combined sampling approach, he/she is prompted to specify the weight to be given to each of the algorithms to be combined. A default weight of 1 is

132

given to each of the surface patch size and surface mean Gaussian curvature-based sampling scenarios. The dialogue for that step is shown in figure 5.6.

If the user selects the optimization option, the system performs Genetic Algorithms (GA's) optimization to determine the optimum locations of the sample grid points.

Figure 5.3. Surface Data Entry Form
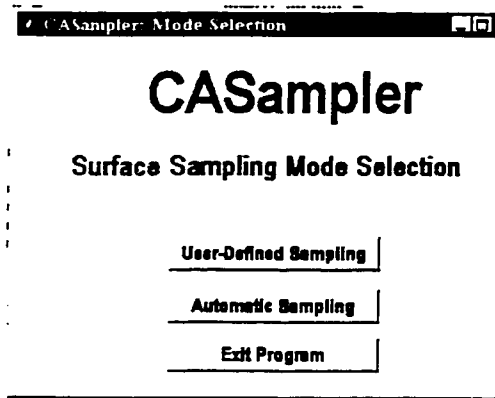
Figure 5.4. Sampling Mode Selection
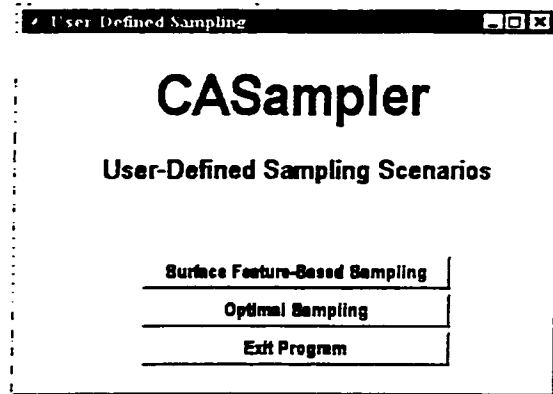
Figure 5.5. User Defined Sampling
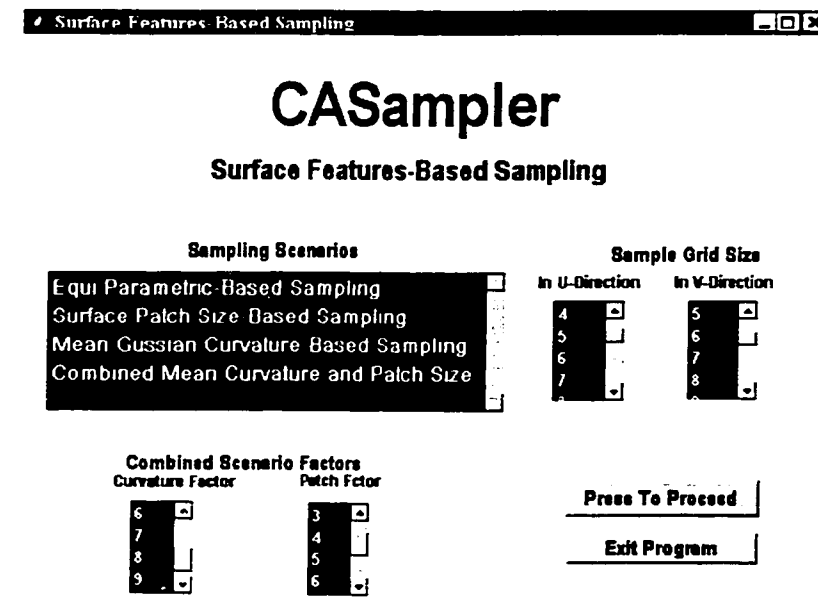
Figure 5.6. Surface Feature-Based Sampling

The automatic mode of CASampler performs the process of blending the surface patch size and surface mean Gaussian curvature-based sampling scenarios without user interference. While CASampler is performing the sampling task, a program status screen

134

displays a summary of the process variables, as well as the program status (Fig. 5.7).
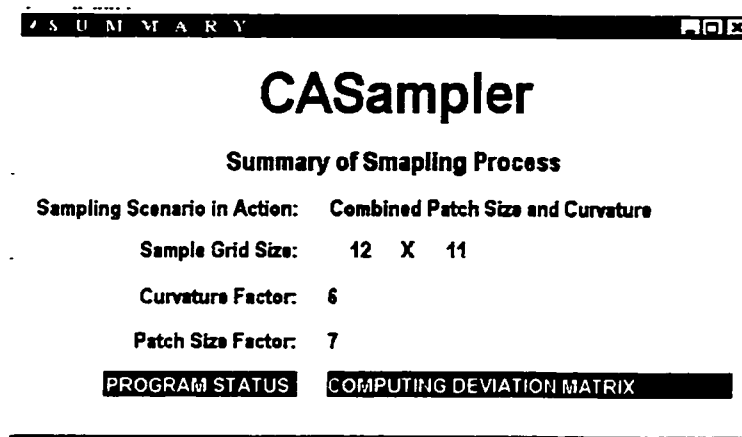
Figure 5.7 Program Status Summary Screen

After the sampling process is carried out by CASampler, the program constructs

the substitute geometry (the NURBS surface that fits the sample points), and computes

the deviation of the substitute geometry from the CAD model (original NURBS

surface). The program then stores all the calculated data for future retrieval. The results

visualization dialogue, shown in figure 5.8, can access the data files produced by the

program throughout the sampling process. It allows the user to view the surface CAD

data, the sample points distribution (Fig. 5.9), the substitute geometry, and the deviation

distribution (Fig. 5.10).
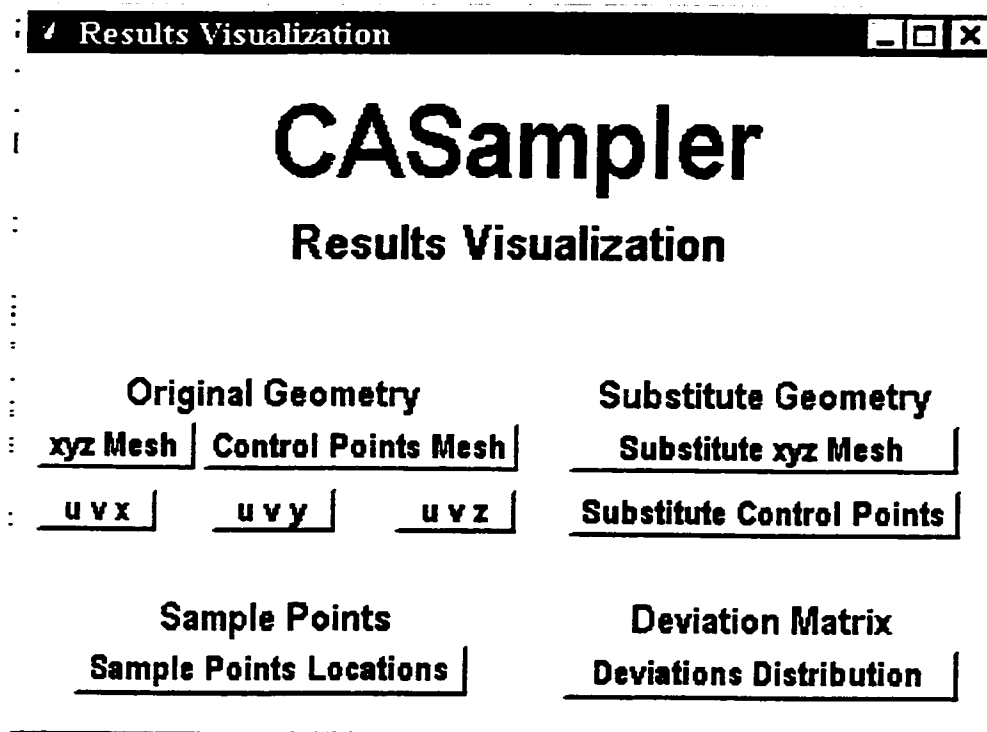
Figure 5.8. Results Visualization Screen
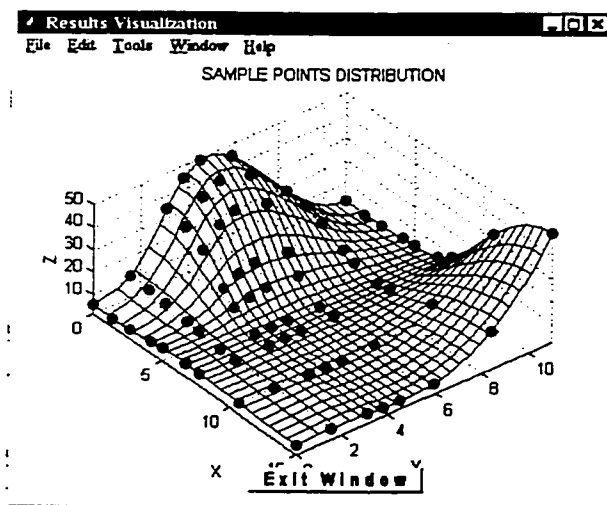
Figure 5.9. Samples Distribution
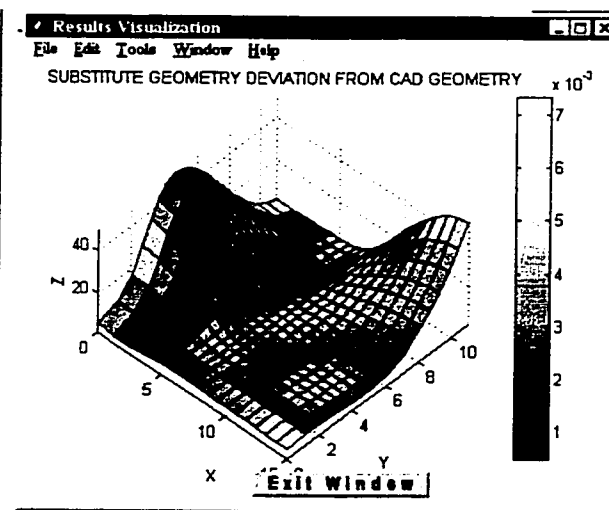
Figure 5.10. Deviation Distribution

136

5.4. Discussion

An interactive system for the Computer-Aided Sampling of free form surfaces has been presented in this chapter. The CASampler includes several innovative features; it offers alternative solutions to the sampling problem, provides several optimality criteria, as well as automatic sampling option. CASampler is a candidate for integration with Computer-Aided Tactile Inspection Planning systems , such as CATIP (Limaiem, A., and ElMaraghy, H. A.,1999).

In order to integrate CASampler with CATIP, it needs to be migrated from MATLAB to C++. Furthermore, new features need to be added (e.g., the ability to read solid model files instead of plain ASCII data). This latter feature can be achieved by implementing a geometric modeling kernel (e.g. ACIS of Spatial Technologies Inc.).

CHAPTER SIX

CONCLUSION

This chapter is divided into three sections, i.e., contributions, summary of research, discussions and conclusions, and future research.

6.1. Contributions

The work reported in this thesis makes the following contributions to the field of CMM inspection planning:

1. Four methodologies for the sampling of free form surfaces, based on their NURBS characteristics were developed.

2. A Genetic Algorithms (GA's) based method for the optimum sampling of free form surfaces was introduced.

3. A utility for the automatic selection of free form surface sampling algorithm, based on the surface complexity check was developed, and integrated with the developed sampling methodologies.

4. An integrated system for the sampling of free form surfaces was developed,

implemented using MATLAB 5.3 (Math Works Inc.). This system is a candidate for integration with computer-aided tactile inspection planning, i.e., CATIP (Limaiem, A., and ElMaraghy, H., 1999).

6.2. Summary Of Research

The focus of this research was the sampling for CMM inspection planning of free form surfaces. The main concern was the allocation of sample points on the free form surface, so that the surface fitted to the sampled points is as close as possible to the CAD data.

Four heuristic algorithms were developed to sample points from free form surfaces, based on the surface NURBS features. These are equi-parametric sampling, surface patch size based sampling, surface patch mean curvature based sampling, and a hybrid approach of the last two algorithms.

For better utilization of the heuristic sampling algorithms, the selection of the sampling algorithm was automated through the development of an algorithm that performs a series of complexity checks on the surface being sampled. According to the results of the surface complexity check, this algorithm selects which sampling algorithm to apply to the surface being tested.

6.3. Discussion And Conclusions

Results of the simulation study indicate that the equi-parametric sampling technique includes large approximation and cannot provide reliable sampling plans, especially in the case of surfaces which includes patches with different sizes. A larger sample size should be used to reduce the maximum deviation to a value that is close to the random measurement error component.

A patch size based sampling proved to be capable of producing a better sampling plan when the variation in surface patch sizes is the main contributor to the surface complexity. However, higher accuracy makes it necessary to use a larger sample size, especially in the case of surfaces with variations in curvature values.

Patch mean Gaussian curvature based sampling was found to be very efficient in cases of surfaces with sharp curvature changes. However, if the sharp curvature change occurred within smaller surface patches, this methodology may produce sampling plans in which larger patches are not given sufficient share of the sample points, resulting in a larger deviation between the fitted surface and the geometric model.

Combining the last two scenarios produces better, and more accurate sampling plans. The manner in which this methodology is implemented allows the user to assign a

A methodology for Genetic Algorithms (GA's) based optimum sampling of free from surfaces was also presented in this work. The goal of that algorithm is to provide an optimum solution to the sampling problem when the heuristic methods fail to provide a satisfactory solution. This is when the complexity of the surface being sampled is high.

The sampling algorithms developed in this research were implemented in MATLAB 5.3 (Math Works Inc.), tested, and exposed to extensive simulation studies. The simulation of sampling algorithms involved the following steps:

1. The construction of an ideal geometry (surface CAD model).

2. The simulation of the manufacturing process form error.

3. The application of the sampling algorithm, in which the CMM random measurement error is simulated.

4. Fitting a NURBS surface to the sampled points. This surface is referred to as the substitute geometry.

5. Computing the maximum deviation between the CAD surface, and the substitute geometry. This maximum deviation is used as a measure of the sampling algorithm efficiency, i.e., the lower the deviation between the CAD surface, and the substitute geometry, the higher the more accurate the sampling algorithm.

relative weight to each of the combined two methodologies. For instance, if the surface patch sizes were significantly different, a higher weight is assigned to the patch size-based sampling. The same applies if relatively sharp curvature changes are involved.

The Genetic Algorithms (GA's) based optimization provided the most accurate plan amongst the four sampling methodologies presented in this research. Although it involves extensive calculations and CPU time, this is normally done off line. If the surface to be inspected is complex, especially when high throughput is considered, it is of great importance to produce a reliable and efficient sampling plan that optimally allocates the sample points. In this case, GA's based sampling provides a very efficient tool, especially as the sampling is performed off-line.

The sampling algorithms presented in this research were compared to the uniform sampling pattern presented by Yao, and Menq (1992). The developed sampling algorithms proved to perform better than the uniform sampling pattern, when the surface being sampled is one with complex features, especially changes in the values of surface curvature.

Sampling has been proven to be a major factor that affects the accuracy of the CMM measurement of free form surfaces. Changing the sampling plan changes the distribution,

142

and magnitude of the deviation of the substitute geometry from the nominal geometry. This contributes to the measurement uncertainty. Therefore, it is of great importance to implement a sampling plan that best represents the geometric feature being inspected. The sampling plan should reflect the degree of complexity of the geometric feature being inspected. When the surface complexity reaches a high level, it becomes crucial to find an optimum solution to the sampling problem.

6.4. Future Research

There are a number of research, and development issues related to the sampling of free form inspection planning that need further investigation.

6.4.1. Research issues

1. The development of algorithms to sample points in a random order, rather than a grid of points.

2. Further exploration of methodologies for optimum sampling of free form surfaces, and the conduction of studies to compare the performance of GA's versus other optimization methods.

3. The development of a methodology to determine the optimal sample size for

CMM inspection planning of free form surfaces, and integrate this methodology with the samples' distribution algorithms.

6.4.2. Development issues

1. Migrating the algorithms developed in this research from MATLAB to C++ programming language. This is an essential step to integrate a sampling module with CATIP (Limaiem, A., and ElMaraghy, H., 1999).

2. The development of a utility to allow the system user to interactively allocate sample points in areas where he/she desires to place more emphasis.

3. The integration of the developed algorithms with a system for computer-aided tactile inspection planning.

4. Enhancing the system to deal with solid models, instead of only ASCII data. This step can be achieved through the use of commercial geometric modeling kernels (e.g. ACIS of Spatial Technology Inc.).

REFERENCES

Caskey, G., Hocken, R., Palanvelu, D., Raja, J., Wilson, R., Chen, K., and Yang, J., "Sampling Techniques for Coordinate Measuring Machines", Proceedings of NSF Design and Manufacturing Systems Conference, Atlanta, GA, 1992, pp. 779-786.

Cho, M. W., and Kim, K., "New Inspection Planning Strategy for Sculptured Surfaces Using Coordinate Measuring Machine", International Journal of Production Research, Vol. 33, No. 2, pp. 427-444, 1995.

Chong, Edwin K. P., and Zak, Stanislaw H., "An Introduction to Optimization", John Wiley and Sons Inc., 1996.

Dowling, M. M., Griffin, P. M., Tsui, K. L., and Zhou, C., "Statistical Issues in Geometric Feature Inspection Using Coordinate Measuring Machines", Technometrics, Vol. 39, No. 1, pp. 3-17, 1997.

Edgeworth, R., and Wilhelm, R. G., "Adaptive Sampling for Coordinate Metrology", Precision Engineering, No. 23, 1999, pp. 144-154.

ElKott, Diaa F., ElMaraghy, Hoda A., and ElMaraghy, Waguih H., "Computer-Aided Free Form Surfaces Sampling System", The 2000 Pacific Conference on Manufacturing, September 6-8, 2000, Southfield, Michigan, Vol.2, pp. 830-835.

ElKott, Diaa F., ElMaraghy, Hoda A., and Nassef, Ashraf O., "Optimal Sampling of NURBS Curves", Forum on Dimensional Metrology, November 23-25, 1998, Quebec City, Quebec, Canada.

ElKott, Diaa F., ElMaraghy, Hoda A., and Nassef, Ashraf O., "Sampling for Free Form Surfaces Inspection Planning", CD-Rom Proceedings of The 1999 ASME Design Engineering Technical Conference, September 12-15, 1999, Las Vegas, Nevada.

ElMaraghy, H. A., and ElMaraghy, W. H., "Computer Aided Inspection Planning (CAIP)", Manufacturing Research and Technology 20, Advances in Feature-Based Manufacturing, Elsevier Science, B. V., pp. 363-395, 1994.

ElMaraghy, H. A., and Gu, P. H., "Expert System for Inspection Planning", Annals of the CIRP, Vol. 36/1/1987, pp. 85-89, 1987.

ElMaraghy, Hoda A., "Evolution and Future Perspective of CAPP", Annals of the CIRP, Vol. 42, No. 2, pp. 1-13, 1993.

Fan, K. C., and Leu, M., C., "Intelligent Planning of CAD-Directed Inspection for Coordinate Measuring Machines", Computer Integrated Manufacturing Systems, Vol. 11, No. 1-2, pp. 43-51, 1998.

Fang, K. T., Wang, S. G., and Wei, G., "A Stratified Sampling Model in Spherical Feature Inspection Using Coordinate Measuring Machines", Statistics & Probability Letters, No. 51, 2001, 25-34.

Gen, Mitsuo, and Cheng, Runwei, "Genetic Algorithms and Engineering Design", John Wiley and Sons Inc., 1997.

Hocken, Robert J., Raja, J., and Uppliappan, B., "Sampling Issues in Coordinate Metrology", Manufacturing Review, Vol. 6, No. 4, pp. 282-294, December 1993.

Karr, Charles L., and Freeman, Michael L., "Industrial Applications of Genetic Algorithms", CRC Press, Boca Raton, FL, 1999.

146

Kim, D. H., and Ozsoy, T. M., "New Sampling Strategies for Form Evaluation of Free Form Surfaces", CD-Rom Proceedings of The 1999 ASME Design Engineering Technical Conference, September 12-15, 1999, Las Vegas, Nevada.

Kim, W. S., and Raman, S., "On The Selection of Flatness Measurement Points in Coordinate Measuring Machine Inspection", International Journal of Machine Tools and Manufacture, No. 40, 2000, pp. 427-443.

Kreici, James V., "Application Software", Coordinate Measuring Machines and Systems, Edited by John A. Bosch, pp. 39-74, Marcel Dekker Inc., 1995.

Kweon, Soonki, and Medeiros, D. J., "Part Orientations for CMM Inspection Using Dimensional Visibility Maps", Computer-Aided Design, Vol. 30, No. 9, pp. 741-749, 1998.

Lee, G. L., Mou, J., "Design The Sampling Strategy for Dimensional Measurement of Geometric Features Using Coordinate Measuring Machine", Japan/USA Symposium on Flexible Automation, ASME 1996, Vol. 2, pp. 1193-1200.

Limaiem, A., and ElMaraghy, H. A., "Integrated Accessibility Analysis and Measurement Operations Sequencing for CMMs", Journal of Manufacturing Systems, Vol. 19, No. 2, 2000, pp. 83-93.

Limaiem, Anis, and ElMaraghy, Hoda A., "Automatic Planning for Coordinate Measuring Machines"; Proceedings of the 1997 IEEE International Symposium on Assembly and Task Planning; Marina del Rey, CA; August 1997; pp. 243-248.

Limaiem, Anis, and ElMaraghy, Hoda A., "CATIP: A Computer-Aided Tactile Inspection Planning System", International Journal of Production Research, Vol. 37, No. 2, pp 447-465, 1999.

Machireddy, R. K., Hari, Y., and Hocken, R., "Sampling Techniques for Measurement of a Plane Surface Using a Coordinate Measurement Machine", National Design Engineering Conference, Chicago, IL, ASME, 1993.

Menq, C. H,, Yau, H. T., Lai, G. Y., and Miller, R. A., "Statistical Evaluation of Form Tolerances Using Discrete Measurement Data"; 1990 ASME Winter Annual Meeting; ASME; PED; No. 47; pp. 135-149.

Menq, C. H., Yau, H. T., and Wong, C. L., "AN Intelligent Planning Environment for Automated Dimensional Inspection Using Coordinate Measuring Machines", Transactions of the ASME, Journal of Engineering for Industry, Vol. 114, pp. 222-230, 1992.

Mortenson, M., "Geometric Modeling", John Wiley and Sons, 1985.

Ni, Jun, and Waldele, Franz, "Coordinate Measuring Machines ", Coordinate Measuring Machines and Systems, Edited by John A. Bosch, pp. 39-74, Marcel Dekker Inc., 1995.

Orady, E., Chen, Y., Li, S., and El-Baghdady, A. A., "A Fuzzy Decision-Making System for CMM Measurements in Quality Control", The 2000 Pacific Conference on Manufacturing, September 6-8, 2000, Southfield, Michigan, Vol.2, pp. 939-944.

Pahk, H. J., Jung, M. Y., Hwang, S. W., Kim, Y. H., Hong, Y. S., and Kim, S. G., "Integrated Precision Inspection System for the Manufacturing of Moulds Having CAD Defined Features", The International Journal of Advanced Manufacturing Technology (1995), Vol. 10, pp. 198-207, 1995.

Philips, S. D., Borchardt, B., and Cskey, G., "Measurement Uncertainty Considerations for Coordinate Measuring Machines", NISTIR 5170, U.S. Department of

Commerce, Technology Administration, National Institute of Standards and Technology, 1993.

Philips, Steven D., "Performance Evaluation", Coordinate Measuring Machines and Systems, Edited by John A. Bosch, pp. 39-74, Marcel Dekker Inc., 1995.

Piegle, Les, and Tiller, Wayne, "The NURBS Book", Springer-Verlag, 1997.

Reklaitis, G. V., Ravindran, A., and Ragsdell, K. M., "Engineering Optimization-Methods and Applications", John Wiley and Sons Inc., 1983.

Slocum, Alexander H., "Precision Machine Design", Prentice-Hall Inc., 1992.

Spyridi. A. J., and Requicha, A. A. G., "Automatic Programming of Coordinate Measuring Machines", Proceedings of The IEEE International Conference on Robotics and Automation, San Diego, CA, pp. 1107-1112, 1994.

Uppliappan, B., Raja, J., Hocken, R. J., and Chen, K., "Sampling Methods and Substuitute Geometry Algorithms for Measuring Cylinders in Coordinate Measuring Machine", Transactions of NAMRI/SME, Vol. XXV, pp. 353-358, 1997.

Vafaeesefat, A., and ElMaraghy, H. A., "Automated Accessibility Analysis and Measurement Clustering for CMMs", International Journal of Production Research, Vol. 38, No. 10, 2000, pp. 2215-2231.

Wong, C. L., Menq, C. H., and Bailey, R., "Computer-Integrated Dimensional Inspection of Manufactured Objects Having Sculptured Surfaces", Advanced Manufacturing Engineering, Vol. 5, pp. 37-44, January 1991.

Woo, T. C., and Liang, R., "Optimal Sampling for Coordinate Measurement: Its

Definition and Algorithm", Quality Through Engineering Design, Edited by Kuo, W., pp. 333-346, Elsevier Science Publishers B. V., 1993.

Yau, H. T., and Menq, C. H., "An Automated Dimensional Inspection Environment for Manufactured Parts Using Coordinate Measuring Machines", International Journal of Production Research, Vol. 30, No. 7, pp. 1517-1536, 1992.

Zeid, Ibrahim, "CAD/CAM Theory and Practice", McGraw-Hill Inc., 1991.

Zhang, Y. F., Nee, A. Y. C., Fuh, J. Y. H., Neo, K. S., and Loy, H. K., "A Neural Network approach to Determining Optimal Inspection Sampling Size for CMM", Computer Integrated Manufacturing Systems, Vol. 9, No. 3, pp. 161-169, 1996.

Zieman, C. W., and Medeiros, D. J., "Automating Probe Selection and Part Setup for Inspection on a Coordinate Measuring Machine", International Journal of Production Research, Vol. 11, No. 5, pp 448-460, 1998.

APPENDIX A

CODE FOR SAMPLING ALGORITHMS

A.1. Nomenclature

– M = Sample size in the u-direction.

– N = Sample size in the v-direction.

– P = 3D matrix defining the surface control points.

– U = U knot vector.

– V = V knot vector.

– p = Surface degee in u-direction.

– q = Surface degree in v-direction.

– su = The values of the u-parameter coordinate of the sample points.

– sv = The values of the v-parameter coordinate of the sample points.

– Su = Matrix of the u-coordinates of the sample points.

– Sv = Matrix of the v-coordinates of the sample points.

– Sx = Matrix of x-coordinate of sampling points.

– Sy = Matrix of y-coordinate of sampling points.

– Sz = Matrix of z-coordinate of sampling points.

151

- Ucurvatures = Surface curvature in the u-direction.

- Vcurvatures = Surface curvature in the v-direction.

- Ucurvature_p = Matrix of surface curvatures on patch "p" in u-direction.

- Vcurvature_p = Matrix of surface curvatures on patch "p" in v-direction.

- Patch_Mean_Curv_U = Mean curvature of a patch in u-direction.

- Patch_Mean_Curv_V = Mean curvature of a patch in v-direction.

- Curv_Weight = weight of the patch mean curvature-based sampling *(to use in the hybrid sampling method)*.

- Patch_Weight = weight of the patch size-based sampling *(to use in the hybrid sampling method)*.

- Su_Patch = u-parameter locations of the sample points, obtained from patch size-based sampling.

- Sv_Patch = v-parameter locations of the sample points, obtained from patch size-based sampling.

- Su_Curvature = u-parameter locations of the sample points, obtained from patch mean curvature-based sampling.

- Sv_Curvature = v-parameter locations of the sample points, obtained from patch mean curvature-based sampling.

A.2. Equi-parametric Sampling Algorithm

1. Input: M, N, P, U, V, p, q;

2. U_Interval = [MAX(U) − MIN(U)]/(M-1);

3. V_interval = [MAX(V) − MIN(V)]/(N-1);

4. su(1) = MIN(U);

   su(M) = MAX(U);

   For i ∈ {2,....,M-1}

   su(i) = su(I-1) + U_Interval;

   End

5. sv(1) = MIN(V);

   sv(N) = MAX(V);

   For j ∈ {2,....,N-1}

   sv(i) = sv(j-1) + V_Interval;

   End

6. *For k ∈ {1,...,M}*

   For l ∈ {1,...,N}

   Sv(k,l) = sv(l);

   End

   End

7. *For l ∈ {1,...,N}*

   For k ∈ {1,...,M}

   Su(k,l) = su(k);

   End

   End

8. Use Su, and Sv to determine Sx, Sy, Sz.

A.3. Surface Patch Size Based Sampling Algorithm

1. Input: M, N, P, U, V, p, q;

153

2. Compute U_Spans. U_Spans = {Usp1, Usp2, ..., Uspm};

3. Compute V_Spans. V_Spans = {Vsp1, Vsp2, ..., Vspn};

4. su(1) = MIN(U);

    su(M) = MAX(U);

    For k ∈ {2,....,M-1}

        U_Span_Weight(k) = Usp(k)/SUM(U_Span);

        U_Span_Share(k) = U_Span_Weight(k) x M;

        Equally distribute points within span;

    End

    Get su;

5. sv(1) = MIN(V);

    sv(N) = MAX(V);

    For l ∈ {2,....,N-1}

        V_Span_Weight(l) = Vsp(l)/SUM(V_Span);

        V_Span_Share(l) = V_Span_Weight(l) x N;

        Equally distribute points within span;

    End

    Get sv;

6. *For i ∈ {1,...,M}*

        For j ∈ {1,...,N}

            Sv(i,j) = sv(j);

        End

    End

7. *For j ∈ {1,...,N}*

        For i ∈ {1,...,M}

            Su(i,j) = su(i);

        End

154

End

8. Use Su, and Sv to determine Sx, Sy, Sz.

A.4. Patch Mean Curvature-based Sampling

1. Input: M, N, P, U, V, p, q;

2. Compute U_Spans. U_Spans = {Usp1, Usp2, ..., Uspm};

3. Compute V_Spans. V_Spans = {Vsp1, Vsp2, ..., Vspn};

4. su(1) = MIN(U);

   su(M) = MAX(U);

5. sv(1) = MIN(V);

   sv(N) = MAX(V);

6. SUM_Ucurvatures = 0;

   SUM_Vcurvatures = 0;

   For i ∈ {1,...,m}

         For j ∈ {1,...,n}

               Patch(i,j) = [Usp(i) x Vsp(j)];

               While p ∈ {Patch(i,j)}   Do

                   Patch_Mean_Curv_U(i,j) = MEAN(UCurvature_p);

                   Patch_Mean_Curv_V(i,j) = MEAN(VCurvature_p);

               End

               SUM_Ucurvatures = SUM_UCurvatures+Patch_Mean_Curv_U(i,j);

               SUM_Vcurvatures = SUM_VCurvatures+Patch_Mean_Curv_V(i,j);

         End

   End

7. *For i ∈ {1,...,m}*

         For j ∈ {1,...,n}

155

Patch_UWeight(i,j)= Patch_Mean_Curv_U(i,j)/SUM_Ucurvatures;

Patch_VWeight(i,j)= Patch_Mean_Curv_V(i,j)/SUM_Vcurvatures;

End

End

8. Equally distribute sample points within each patch.

9. Get Su, Sv.

10. Use Su, and Sv to determine Sx, Sy, Sz.

A.5. Combined Surface Patch Size, And Patch Mean Curvature Based Sampling

1. Input: M, N, P, U, V, p, q, Curv_Weight, Patch_Weight;

2. Perform Patch Size Based sampling;

3. Get Su_Patch, and Sv_Patch;

4. Perform Patch Mean Curvature Based sampling;

5. Get Su_Curvature, and Sv_Curvature;

6. W_Su_Curvature = Su_Curvature x Curv_Weight;

   W_Sv_Curvature = Sv_Curvature x Curv_Weight;

7. W_Su_Patch = Su_ Patch x Patch _Weight;

   W_Sv_ Patch = Sv_ Patch x Patch _Weight;

8. Su = (W_Su_Curvature+ W_Su_Patch)/(Patch_Weight+ Curv_Weight);

9. Sv = (W_Sv_Curvature+ W_Sv_Patch)/(Patch_Weight+ Curv_Weight);

10. Use Su, and Sv to determine Sx, Sy, Sz.

APPENDIX B

USER GUIDE FOR COMPUTER-AIDED FREE FORM SAMPLING SYSTEM (CASampler)

As described in chapter five, CASampler is composed of a collection of MATLAB functions, and tool boxes. This appendix explains how to setup, and use the program. Also, explained are the formats of the various input data files used by CASampler.

B.1. Program Setup

B.1.1. File locations

To set up CASampler, it is important to define the locations of its various functions, and tool boxes to be accessible by MATLAB. Figure B.1 illustrates a typical example for the directory structure of CASampler files. The contents of every directory are explained in table B.1.

Table B.1. Contents Of CASampler Directories

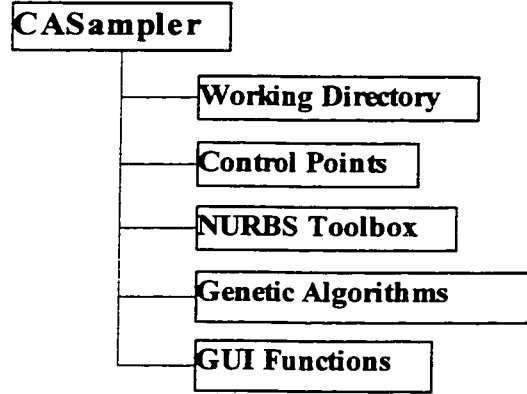| Category	Functionality
Working directory	Main work directory. Output files are saved here.
Control Points	Control points data files are stored here.
NURBS Toolbox	Copy NURBS toolbox files here.
Genetic Algorithms Toolbox	Copy Genetic Algorithms toolbox files here.
GUI Functions	Copy all the GUI functions here.

Figure B.1. Structure Of CASampler Installation Directory

B.1.2. Construction of control points data files

All the data files containing the control points of surfaces are stored in the "control points" directory. The control points of a particular surface are stored in data file named "Surface_Name.m". The control points are stored in a 3D array format. The surface control point is expressed in 4 coordinates, i.e., its Cartesian coordinates, and its weight.

Let $P_{i,j}$ be a control point that lies in the intersection of the ith layer of control points in the u-direction, an the jth layer of control points in the v-direction. $P_{i,j}$ is expressed as:

$$P_{i,j} = \begin{bmatrix} x_{i,j}, & y_{i,j}, & z_{i,j}, & w_{i,j} \end{bmatrix}^T \tag{B.1}$$

Let P be a matrix of control points for a particular surface. P is expressed as:

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & \cdots & P_{1,R} \\ P_{2,1} & P_{2,2} & P_{2,3} & \cdots & P_{2,R} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ P_{S,1} & P_{S,2} & P_{S,3} & \cdots & P_{S,R} \end{bmatrix} \tag{B.2}$$

158

where:

R = Number of control point layers in the u-direction.

S = Number of control point layers in v-direction.

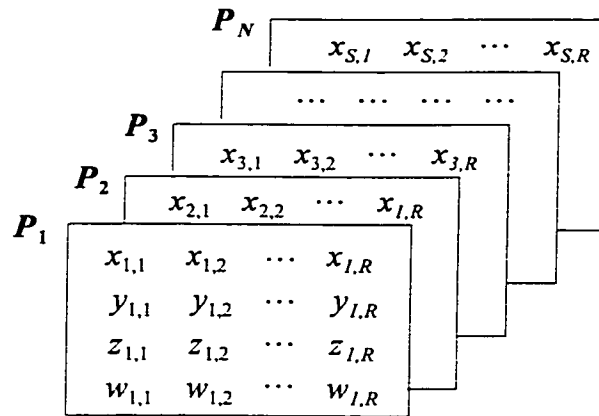Figure B.2. Illustrates the structure of the different layers of P .

Figure B.2. Layers Of P Control Points 3D Matrix

Figure B.3 illustrates an example of a 3D matrix of control points, whose dimensions

are $5_{u\,direction}$ $\times$ $5_{v\,direction}$ $\times$ $4_{x,y,z,w}$.

```
P(:,:,1) = [    0    0    0    0    0;
                0    2    5    8    11; %     First layer of control points
                6    6  100    2    2;
                1    1    1    1    1]; % ← Weights of control points

P(:,:,2) = [    2    2    2    2    2;
                0    2    5    8    11; %     Second layer of control points
                4    6  100    4    1;
                1    1    1    1    1]; % ← Weights of control points

P(:,:,3) = [    5    5    5    5    5;
                0    2    5    8    11; %     Third layer of control points
                8    8    6    6    4;
                1    1    1    1    1]; % ← Weights of control points

P(:,:,4) = [    8    8    8    8    8;
                0    2    5    8    11; %     Fourth layer of control points
                7    5    6    5    1;
                1    1    1    1    1]; % ← Weights of control points

P(:,:,5) = [   12   12   12   12   12;
                0    2    5    8    11; %     Fifth layer of control points
                6    7    4    3    50;
                1    1    1    1    1]; % ← Weights of control points
```
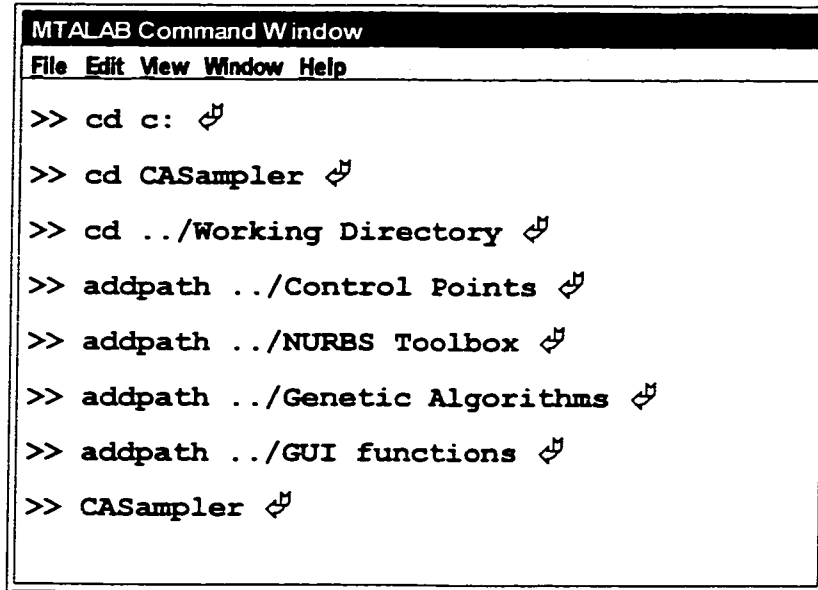
Figure B.3. MATLAB Format Of **P** Control Points 3D Matrix

B.2. Running The Program

Running the program takes place through the following steps:

1. Run MATLAB 5.3 (Math Works Inc.).

2. Add the directories illustrated in fig. B.1 to the program search path.

3. Run "CASampler" file from the command prompt.

Figure B.4. Illustrates steps 1 through 3.

```
┌─────────────────────────────────────────────┐
│ MTALAB Command Window                         │
├─────────────────────────────────────────────┤
│ File  Edit  View  Window  Help                │
├─────────────────────────────────────────────┤
│ >> cd c:                                      │
│                                               │
│ >> cd CASampler                               │
│                                               │
│ >> cd ../Working Directory                    │
│                                               │
│ >> addpath ../Control Points                  │
│                                               │
│ >> addpath ../NURBS Toolbox                   │
│                                               │
│ >> addpath ../Genetic Algorithms             │
│                                               │
│ >> addpath ../GUI functions                   │
│                                               │
│ >> CASampler                                  │
│                                               │
└─────────────────────────────────────────────┘
```

Figure B.4. Running CASampler On MATLAB Command Prompt

Once the last command "CASampler" is input, the program starts, and the user is prompted to input the data for constructing the surface, and perform the sampling. The program outputs the sampling plans. The interaction with CASampler is explained in chapter 5.

VITA AUCTORIS

Diaa ElKott was born in 1970 in Mansoura, Egypt. He graduated from Mansoura Secondary School in 1987. From there he went to Mansoura University where he obtained a B.Sc. In Industrial Production Engineering in 1992. After graduation, he worked as a project engineer in machine tool manufacture. He is currently a candidate for the Master of Applied Science degree in Industrial and Manufacturing Systems Engineering at the University of Windsor, and hopes to graduate in Winter 2001.