Development of a Methodology for Reconditioning Complex Precision-Machined Components Using a Robotic Deburring Workcell

BRUCE J ROUSSEL

A Thesis

in

The Department

of

Mechanical Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montréal, Quebec, Canada

July 1996

ISBN 0-612-18432-3

Canada

ABSTRACT

Development of a Methodology for Reconditioning Complex Precision-Machined Components Using a Robotic Deburring Workcell

BRUCE J ROUSSEL

The methodology of this thesis constitutes techniques for digitizing, feature recognition, and determination of desired reconditioned component geometry for use in corrective machining toolpath generation Digitizing strategies consisting of both off-line and on-line path generation are developed and implemented for a robotic deburring workcell The workcell operating software (see Ayyadevara [13]) is modified to incorporate methodologies developed in this thesis and to carry out experimental verification

A fuzzy logic-based algorithm for feature recognition is presented which can identify lumps, pitting and the like on component surfaces Features are recognized from analysis of the digitized database representing a component surface The present recognition algorithm does not rely on comparison with original design data, but rather mimics on digital computer the innate ability in human intelligence for understanding the *concept* of a flaw Such a capability is important in the refurbishment of complex components such as gas turbine impeller blades which undergo slight warping deformations during service

A surface-fitting scheme is presented for replacement of rejected grid points in a digitized database, yielding a Maximum Likelihood Eventual Surface (MLES), representing desired reconditioned component geometry This scheme performs surface fitting by determining an optimal set of bivariate polynomial coefficients for surface models

approximating local trends in nonrejected component geometry. Sensitivity analysis is carried out for determination of the optimization scheme evaluation point.

Experimental verification of the present techniques for digitizing, recognition and MLES generation are presented for three test surfaces. Extraneous features are placed on these surfaces, which are digitized and afterwards subjected to feature recognition and MLES generation. Average predicted maximum corrective machining error obtained in these case studies is 0.185 grid units, a result indicating potential for successful implementation of these techniques in industrial applications.

TO MY PARENTS,

without your unwavering support this would not have been possible

ACKNOWLEDGEMENTS

I would like to extend the sincerest gratitude and appreciation to my thesis supervisors, Drs R M H Cheng and R Rajagopalan, for their time, effort, insightful guidance, and moral and financial support throughout the course of this research I wish to thank Mr Thai Luong, of Pratt & Whitney Canada, and an adjunct associate professor in this Department, who was instrumental to the conceptualization of the robotic deburring project at Concordia I am also indebted to Dr Chung Wu and Mr Joe Sinclair of Pratt & Whitney Canada Special thanks are due to Mr Venkatesh Ayvadevara and Dr Mostafa Mehrabi for helping to get me started on the robotic workcell, and to Mr Gilles Huard for his technical and moral support I wish to thank Mr Karun Thanjavur for his important contribution of machining the test surfaces used in this work, and Dr Mark Temple-Raston for his explanations of surface modelling Thanks and gratitude are also due to the resourceful staff in the Mechanical Engineering Department's machine shop, for their prompt and cheerful cooperation in producing parts based on my sketches and for repairing things and loaning me tools I would also like to thank all the other faculty, staff and students at the Centre for Industrial Control from whom I have learned so much This work was supported by NSERC Strategic Grant STRO134360 awarded to Drs R M H Cheng and R Rajagopalan Thanks are also due to the Hammond Foundation (Michigan USA), for providing US$ 45,000 towards the purchase of the robotic workcell Finally, I would like to thank my future wife, Ms Hong Chen, whose support and faith in me has been a continual source of strength

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

α	Diagonal direction in digitizing grid (radians) / Wrist joint displacement on Yamaha Zeta-1 deburring robot (degrees)
β	Diagonal direction in digitizing grid (radians) / Wrist joint displacement on Yamaha Zeta-1 deburring robot (degrees)
Δh_c	Digitizing Probe Clearance (mm)
Δh_i	Digitizing Probe Interference (mm)
Δx	Horizontal distance between adjacent digitizing points in the x-direction (mm)
Δy	Horizontal distance between adjacent digitizing points in the y-direction (mm)
ϵ_θ	Vertical angle tolerance for regional segmentation (radians)
$\theta_{\alpha\ i,j}$	Angle subtended in the vertical plane between line segments connecting points adjacent to grid point (i, j) in the α-direction on a grid of digitized elevations (radians)
$\theta_{\beta\ i,j}$	Angle subtended in the vertical plane between line segments connecting points adjacent to grid point (i, j) in the β-direction on a grid of digitized elevations (radians)
$\theta_{x\ i,j}$	Angle subtended in the vertical plane between line segments connecting points adjacent to grid point (i, j) in the x-direction on a grid of digitized elevations (radians)
$\theta_{y\ i,j}$	Angle subtended in the vertical plane between line segments connecting points adjacent to grid point (i, j) in the y-direction on a grid of digitized

	elevations (radians)
$(\mu_{Ha})_R$	High input membership function corresponding to area for region R
$(\mu_{Hr})_R$	High input membership function corresponding to roughness for region R
$(\mu_{La})_R$	Low input membership function corresponding to area for region R
$(\mu_{Lr})_R$	Low input membership function corresponding to roughness for region R
$(\mu_{out})_R$	Output membership function vector
$(\mu_{out,i})_R$	Output membership function for region R corresponding to rule i
A	A point in the workspace of the deburring workcell Workcell cutter axis orientation
a_{pq}	Bivariate polynomial function coefficient
B	A point in the workspace of the deburring workcell / Workcell cutter axis orientation
F_V	Bivariate polynomial function identifier
H	Identifier representing High input membership functions
I_{err}	Error index for grid point replacement computations
$(I_r)_{i,j}$	Roughness index corresponding to grid point (i, j)
L	Identifier representing Low input membership functions
N	Outward normal vector
N	Number of rules in fuzzy rule base / Order of bivariate polynomial surface

	function
N_{imf}	Number of input membership functions
N_{in}	Number of input variables
N_{out}	Number of output variables
N_p	Number of grid points within the current reference disk
$(N_p)_R$	Number of points in region R
$\boldsymbol{n}$	Outward unit normal vector
N_x	Digitizing grid size in x-direction
N_y	Digitizing grid size in y-direction
$\boldsymbol{p}$	Placement vector
p_i	ith component of the placement vector
$p_{i,j}$	Digitized point at grid location (i, j)
R	Regional identifier
r_α	Absolute value of the deviation from π radians of $\theta_{\alpha,i,j}$
r_β	Absolute value of the deviation from π radians of $\theta_{\beta,i,j}$
r_x	Absolute value of the deviation from π radians of $\theta_{x,i,j}$
r_y	Absolute value of the deviation from π radians of $\theta_{y,i,j}$
S	Special character denoting sample-on-request for off-line digitizing path generation
V	Speed of machining point of robotic deburring workcell (mm/s)
w_a	Membership function width corresponding to input membership function $(\lambda_a)_R$

w_r	Membership function width corresponding to input membership function $(x_r)_R$
X	Cartesian x-component of robotic workcell machining point location (mm)
x	Cartesian x-coordinate for bivariate polynomial surface function
$(x_a)_R$	Measure of regional area for region R
x_N	Horizontal displacement of Nth digitizing point (mm)
x_{out}	Crisp output variable from fuzzy classifier
x_p	Cartesian x-coordinate of evaluation point for grid elevation replacement computations
$(x_r)_R$	Regionally-averaged measure of maximum local deviation from flatness for region R
Y	Cartesian y-component of robotic workcell machining point location (mm)
y	Cartesian y-coordinate for bivariate polynomial surface function
y_p	Cartesian y-coordinate of evaluation point for grid elevation replacement computations
Z	Cartesian z-component of robotic workcell machining point location (mm)
z	Surface function elevation
z_{ci}	ith elevation computed using the current set of bivariate polynomial coefficients
z_{di}	ith digitized elevation
z_N	Elevation of Nth point in a digitizing grid (mm)
z_S	Offset added as a safety factor during extrapolation of digitizing elevations

ACRONYMS

CAD	Computer Aided Design
CCD	Charge Coupled Device
CIC	Centre for Industrial Control
CMM	Coordinate Measuring Machine
CNC	Computer Numerical Control
CPU	Central Processing Unit
MLES	Maximum Likelihood Eventual Surface

CHAPTER 1 INTRODUCTION

1.1 Introduction to Automated Inspection

1.1.1 The Need for Automated Inspection

Implementation of quality standards in a mass production environment requires reliable, high-speed verification of conformance to tolerances Such procedures are most easily carried out with automation of inspection, since this eliminates human error and overcomes the restrictive limits of manual productivity Typical human inspection errors include omissions in direct visual or tactile sensing of inspected parts, as well as misinterpretation of results obtained from manually-operated inspection equipment Manual inspection productivity is primarily limited by the rate at which human beings can assimilate and process information necessary to render judgements pertaining to inspected part quality Kalpakjian and McKee [1] describe the conventional approach to automated inspection as consisting of two stages use of computer-controlled instrumentation for component examination, followed by comparison-based analysis referencing nominal design data

More than just cutting costs is at stake in the use of automated inspection, however, since many potential buyers of precision-engineered components are required to ascertain implementation of recognized quality control standards Without automated inspection, it will not be possible for suppliers to fully document the quality of their products Such documentation is the hallmark of engineering standards, and is especially important for components whose failure can result in endangerment of human life Factors which limit implementation of automated inspection procedures include (1) existence of technologies appropriate to the specific task, (2) financing of required R&D or capital equipment

expenditures, (3) managerial / corporate commitments to quality and (4) availability of qualified personnel to carry out the program

Cottman [2] provides a concise overview of one of the most comprehensive industrial quality control standards, namely that maintained by the International Standards Organisation (ISO) Suntag [3] acknowledges that the ISO 9000 series in particular has gained worldwide acceptance The ISO 9001 standard is intended for larger-scale complex manufactured systems or components with significant R&D investment, whose operation involves critical human safety and / or environmental factors ISO 9002 is intended for those less critical engineered products which still require individual operational verification or some form of factory testing prior to delivery ISO 9003 is a standard for implementation of statistical process control for mass-produced products which do not require precision tolerances, and which cannot be individually inspected Industries in which use of ISO standards has become prevalent include aerospace, automotive, electronics, nuclear, and even software development

1.1.2 Methods of Automated Inspection

Automated inspection consists of computer-controlled data acquisition followed by comparison-based analysis providing an assessment of the condition of a component Based on the results of inspection, an automated workcell may afterwards be used to plan and execute required reconditioning steps entirely without human intervention Such an automated workcell typically consists of a computer-controlled robotic manipulator which accurately positions an end effector to perform digitizing or machining The common feature of all

inspection schemes is use of comparison for accept / reject decision-making This comparison is performed between geometric data of the original design and sampled geometry of the actual component, to determine which (rejected) portions of a surface require reconditioning

In the case of inspection for dimensional tolerances and surface finish quality, a digitized geometric database of the component must first be established An automated point-by-point comparison between digitized data and nominal design specifications may then be carried out This will serve to identify all defective regions on the surface Finally, B-splines or other interpolative techniques such as those discussed by Farin [4], and Lancaster and Salkauskas [5] may be used to generate a corrective machining toolpath

In the automotive and aerospace industries, precision automated inspection of dimensional tolerances is often carried out with coordinate measuring machines (CMMs). Tanaka *et al* [6] describe the use of these machines to produce maps of component surfaces suitable for computer-based visualization and analysis Harris *et al* [7] describe how CMMs may be preprogrammed to follow expected surface contours, or to perform automated navigation in real-time over a surface for which no geometric data is available. Technology associated with CMMs will be discussed in more detail in Section 1.4 1

1.2 Introduction to Surface Reconditioning

1.2.1 The Need for Automation in Surface Reconditioning

To assess the need for automation in reconditioning complex precision-machined components, advantages and disadvantages of both manual and automated techniques must be examined. Specifically, the benefits / drawbacks of manual versus automated surface polishing,

deburring, grinding, and light machining are of importance This section outlines such factors, followed by discussion of an industrial example to illustrate special difficulties encountered in surface reconditioning automation This example leads to a summary of the basic requirements for an automated workcell capable of reconditioning complex precision-machined components

The main advantages of manual surface reconditioning for precision components are as follows First, human intelligence - an applied conceptual understanding of the task - allows for minimal supervision Human sensory perception (touch and vision) are naturally utilized by a skilled worker to achieve quality results Finally, there is a wide availability of manual labour capable of performing such simple tasks The main disadvantages of manual surface reconditioning are as follows Manual reconditioning inevitably results in lower dimensional accuracies than those achievable with automation Low capital costs of manual labour are ultimately offset by higher operating costs (salaries, benefits, vacations) and an inherently limited production capacity Finally, other human factors such as stress, fatigue, boredom, motivation and the like all have a potentially adverse impact on both quality and productivity

Automated surface reconditioning is required during overhaul of gas turbine-powered aircraft engines, and other types of turbomachinery and fluid power control surfaces It is of significant economic concern if rotor blades such as those shown in Figure 1 1 must be replaced, since these are costly items In the course of normal operation, these components are subjected to thermal and inertial stresses, abrasion, impacts and corrosion Typically, the rotor blades accumulate localized defects such as cracks, dents, pitting and scratches In addition, burrs resulting from primary machining may be present Wauer [8] describes how

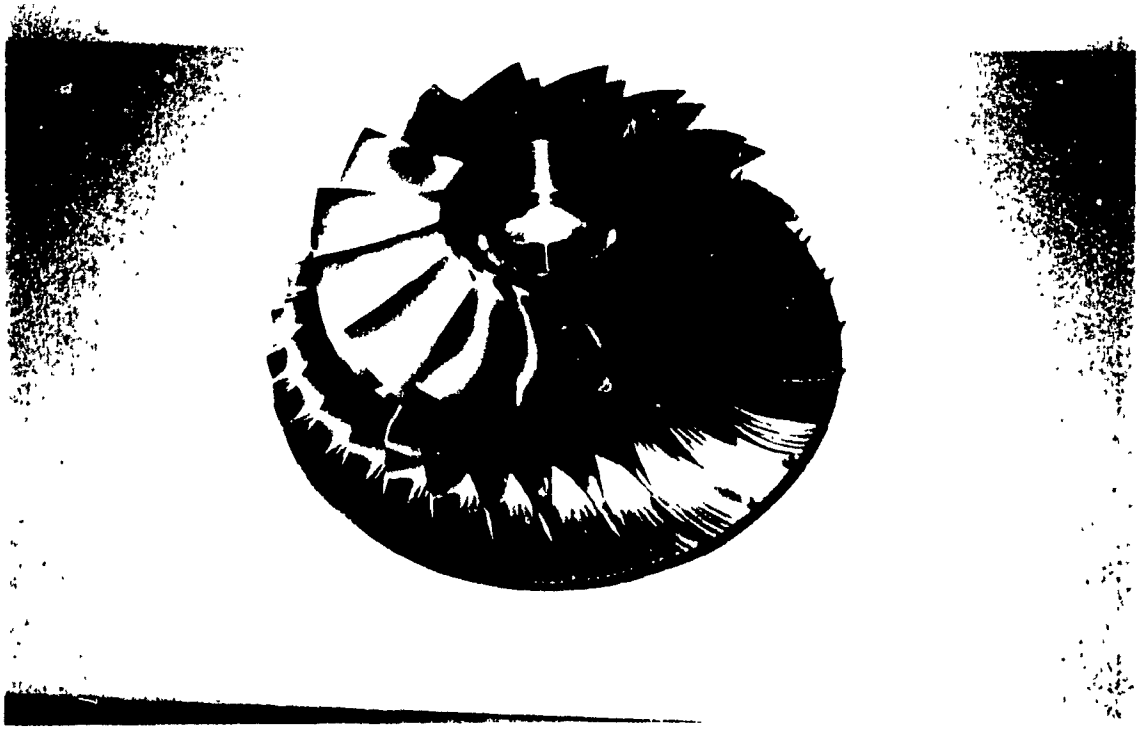


Figure 1.1: Gas Turbine Rotor Blade Unit

such defects constitute a serious problem, since they give rise to the stress concentrations which cause catastrophic failures such as rotor burst. In addition, the blades undergo slight arbitrary warping deformations The effects of warping may be limited to an acceptably small reduction in aerodynamic efficiency, say less than 3%, coupled with slightly increased (but possibly acceptable) rotational imbalance

Automated refurbishment of such rotor blades clearly cannot make use of original design data for comparison-based recognition of defective features, since the blades are distorted One solution is for the workcell to mimic human capabilities in understanding the *concept* of a flaw to perform an autonomous feature recognition With defective features thus identified, the workcell must next compute a desired geometry for the component. This may be done by interpolating local trends in nonrejected points so as to make all rejected features

vanish in a new database representing desired final geometry A corrective machining toolpath may then be generated by comparison of actual and desired surface contours

1.2.2 Machining and Surface Reconditioning

Thyer [9] and Lynch [10] characterize CNC machines as differing from NC machines in that they have a computer as part of the controller unit The advantage of CNC is therefore greater programming flexibility Specifically, program editing can take place at the machine rather than at some remote location which must generate a new copy of the entire program each time a change is made CNC machines may be programmed in teach-and-playback mode using keyboard, mouse or joystick inputs For example, the reader is referred to user's manuals provided by Yamaha Corporation [11] and Yamakazi Mazak Corporation [12], which discuss these programming methods Automatic toolpath generation is also possible offline from a CAD database representing initial (actual) and final (desired) geometries

As an example of online path planning, consider a coordinate measuring machine (CMM) which performs automated digitizing In this mode of operation actual component geometry is *a priori* assumed unknown The system must therefore extrapolate locally-encountered geometric trends so as to compute an expected position for each successive digitized point. The advantage of such a scheme over offline digitizing path generation (using original design data) is in that unsupervised probe navigation becomes possible over regions containing warping deformations or other unexpected features

Surface reconditioning typically consists of light machining cuts for removal of small extraneous features from a surface, and restoration of the original surface finish Cracks,

pitting, and scratches on used components will require grinding down and subsequent buildup with weld prior to machining Heavy initial machining cuts may also be necessary for removal of larger defects It is noteworthy that machining techniques applied to reconditioning of used components can also be applied to the removal of flashes and burrs on new components

1.3 Layout of the Robotic Workcell

1.3.1 Introduction

In this section, an overview of a real-time external controller developed to modify the toolpath of a Yamaha Zeta-1 deburring robot at Concordia University's Centre for Industrial Control (CIC) is presented For further details, the reader is referred to the work by Ayyadevara [13] This robot is of interest in the present work because it has been adapted by the CIC Deburring Group for digitizing surfaces Such experimental data is required to test and demonstrate the research methodology to be presented in this thesis Principal hardware components of the Yamaha deburring robot will first be described, including a separate section discussing digitizing probes used for establishing component geometry Hardware layout of the modified workcell controller is discussed in Section 1 3 4 This controller is an enhanced version of the Yamaha-designed original, incorporating INMOS T-800 transputer technology. For more details, the reader is referred to the documentation supplied by INMOS Corporation [14] Layout of the controller software is described in Section 1.3.5.

1.3.2 Deburring Robot

The Yamaha Zeta-1 deburring robot (Figure 1 2) is a five axis manipulator with three

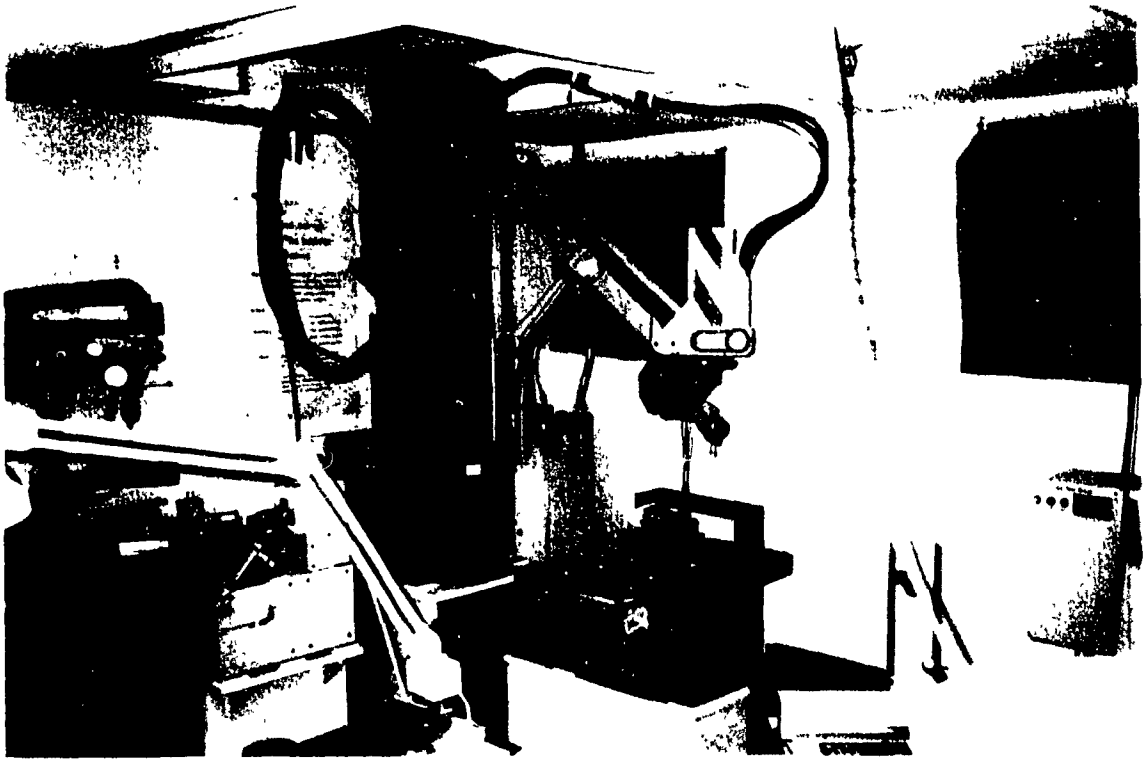


Figure 1.2: Yamaha Zeta-1 Deburring Robot

translational joints (R, θ, Z) analogous to cylindrical coordinate system axes, and two wrist joints (α, β) For more details, the reader is referred to the Yamaha Zeta-1 Deburring Robot User's Manual [11] The lambda-mechanism extending radially from the main vertical column is designed to enhance the structural rigidity of the robot frame The machining point of the robot is defined as the point of intersection of the two wrist joint axes Tool drivers are mounted on the β-drive faceplate, and are designed to position the cutter tip at the machining point Machining point location depends only on (R, θ, Z) displacements, and not on wrist orientation It is therefore possible to use the wrist alone to change tool orientation without any movement of the machining point This robot is reported to have a repeatability of ± 0 1 mm (0 004 in) and can support a maximum moveable load of 30 kg (66 lb) [11]

The robot controller accepts toolpath instructions from a downloaded ASCII script

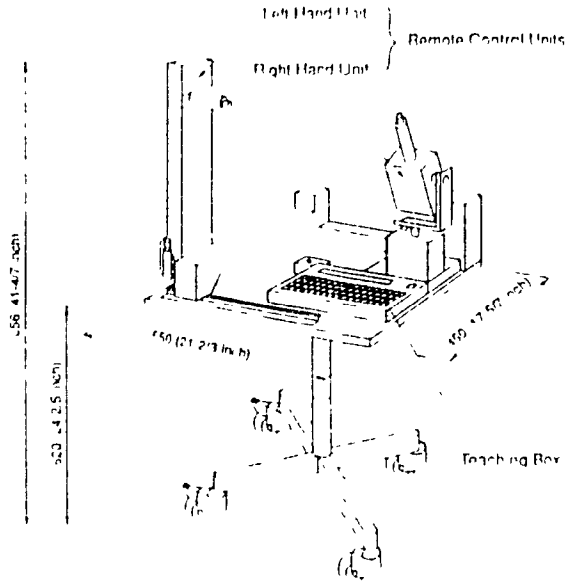


Figure 1.3: Teaching Station for Yamaha Zeta-1 Deburring Robot [11]

file or via a teaching station The teaching station (Figure 1 3) consists of two joysticks and a keypad The joystick labelled "left hand unit" controls machining point position, and the joystick labelled "right hand unit" controls tool orientation The keypad can be used to manually edit machining programs consisting of trajectory coordinates and speeds, and to display current position It can store up to a total of 255 positions and speeds in one or several programs [11] A keypad jog mode is also available as an alternative to the joystick mode

The Yamaha Zeta-1 at Concordia University's Centre for Industrial Control has additional flexibility due to special enhancements of the controller carried out by Rajagopalan *et al* [15] The original Yamaha controller has been interfaced with a PC transputer network to permit real-time toolpath modification. An automated digitizing capability, to be discussed

in more detail in Chapter 2, has been implemented for the robot using this interface

1.3.3 Digitizing Probe

Criteria for probe selection are given in detail by Avvadevara in [13], but will be reviewed here for the sake of completeness

(i) Resolution and Accuracy It is desirable for the probe to be an order of magnitude more accurate than the repeatability of the robot This will ensure that the probe does not make any significant contribution to errors during digitizing

(ii) Range of Measurement The range of the probe should be large enough to accomodate errors due to path execution of the robot

(iii) Accessibility Physical dimensions of the probe determine whether the entire surface of complex components can be digitized.

(iv) Speed of Data Collection Digitizing operations involving large numbers of points will significantly benefit from a faster probe sampling time However, the speed of the robot is also a limiting factor

(v) Sensitivity to Noise. In a typical manufacturing environment, the probe is exposed to electrical noise and vibration in a dust-filled atmosphere Robustness to these conditions is essential for reliability

(vi) Ease of Interfacing. A readily-interfaced probe reduces development time for the deburring workcell.

(vii) Cost The most inexpensive design to satisfy the above criteria is preferred

There are a variety of commercially available devices which fulfil these requirements The robotic workcell has been interfaced with an Ono Sokki EG-225 displacement probe (Figure 1 4) This probe is reported to have a resolution of 1 μm, and an accuracy of 3 μm, with a measuring range of 25 4 mm (the reader is referred to literature by Ono Sokki Ltd [16] for further details) During operation, displacement readings are transmitted at a sampling period of 35 ms A mounting bracket designed at CIC supports the probe as an end effector on the Yamaha robot (Figure 1 5) The undisplaced probe tip is positioned at the machining point so that the probing axis is coincident with the tool axis

As a precaution against probe crashes (in which the probe is accidentally driven into the component causing damage) and eventual damage to the robot structure, a microswitch for emergency stop has been added to the mounting bracket Emergency stop (power cutoff to all axis drive motors) is triggered by the probe stylus reaching within 2 mm of its maximum stroke This feature is particularly useful during the development and verification of digitizing strategies The probe tip is machined from hard resin-based insulating material, to ensure electrical isolation of the probe from the workpiece This reduces the possibility of electrical noise interfering with signal transmission

1.3.4 Workcell Controller

Ayyadevara [13] provides a detailed description of the workcell controller For completeness, a brief overview of this controller is given below Operation of the workcell controller is divided into three parts supervisory level, process level and servo level Servo level control takes place within the Yamaha controller unit, where voltages are generated to drive the DC

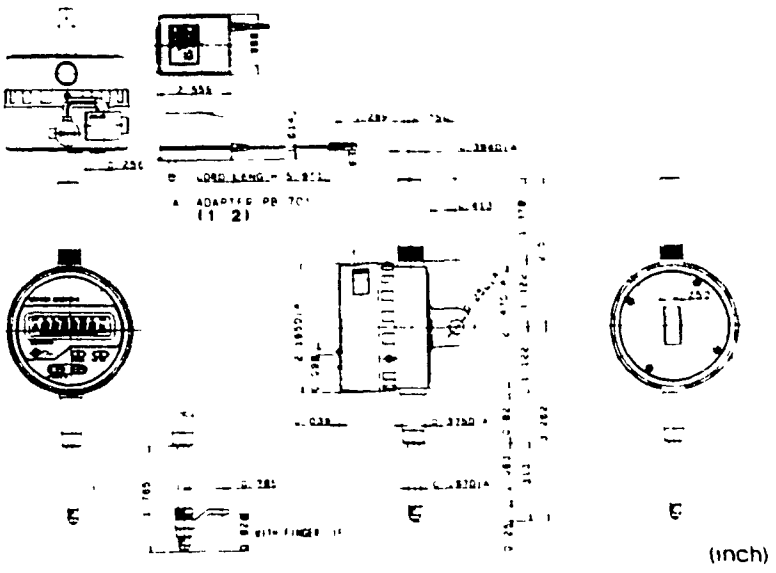


Figure 1.4: Schematic of the Ono Sokki EG-225 Digital Linear Gauge [16]

servomotors for each joint Process level control actions include trajectory tracking command generation and communication with sensors (digital probe and joint optical encoders) Supervisory level control consists of online path planning and interactions with the user

As can be seen in Figure 1 6, the Information Processing Component of the workcell controller consists of a PC-based transputer network The parallel processors used are INMOS T-800 transputers mounted on a TMB008 motherboard and installed in a slot on the IBM PC, which has an INTEL 486-50DX CPU acting as host Ayyadevara [13] has labelled the transputers ROOT, PLANNER, DRIVER and SURFACE ROOT acts as an interface between the host PC and the other transputers Path planning is performed by the transputer labelled PLANNER, and trajectory control functions are carried out using DRIVER In the present work, SURFACE has not been used online Splining and toolpath generation are

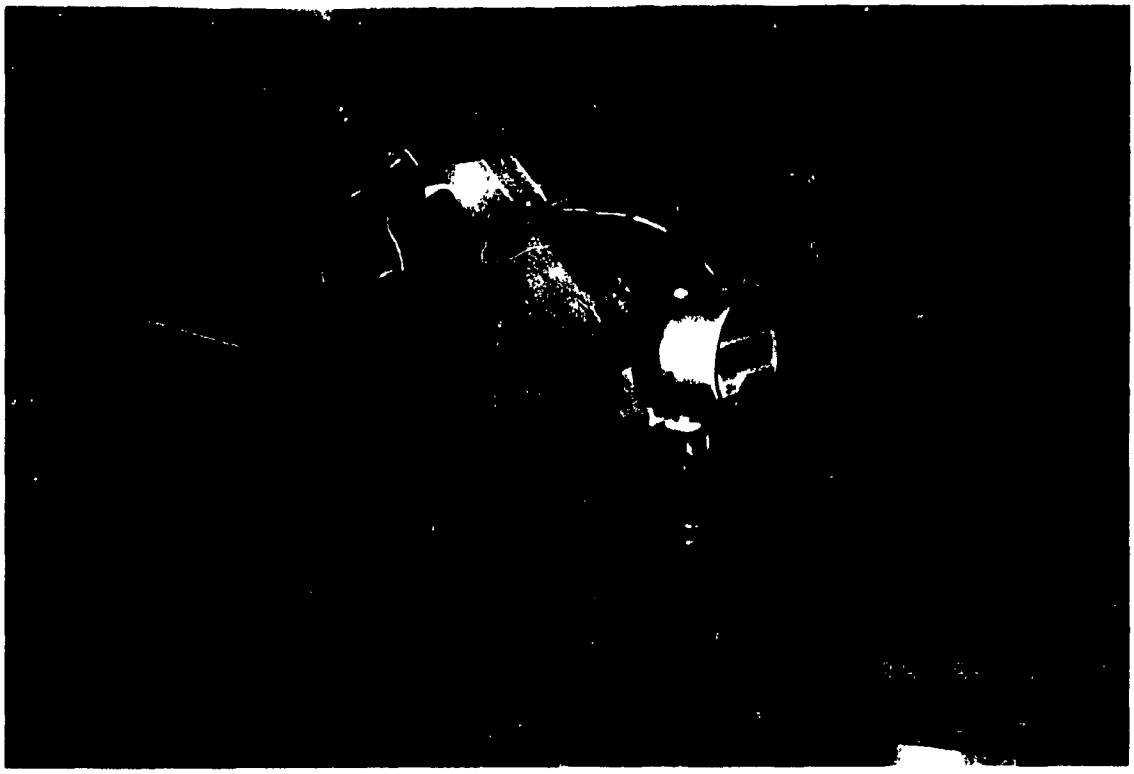


Figure 1.5: Digitizing Probe Mounted on Yamaha Zeta-1 Robot

performed offline using SURFACE. Analysis for feature recognition and surface reconditioning (to be discussed in subsequent chapters) is carried out offline on HOST. A serial-to-parallel interface designed and built by Poon and Huard [17] at Concordia University's Centre for Industrial Control links transputer chips to the data, address or control buses of the low level interface.

1.3.5 Software Layout

All software executing on the transputers has been written in 3L parallel C [18], which is essentially ANSI-standard C [19, 20] augmented with extra libraries to enable parallel processing. Figure 1.7 shows a schematic of the software layout. Tasks are identified in italics and transputer chips are identified in uppercase lettering. A detailed explanation of this

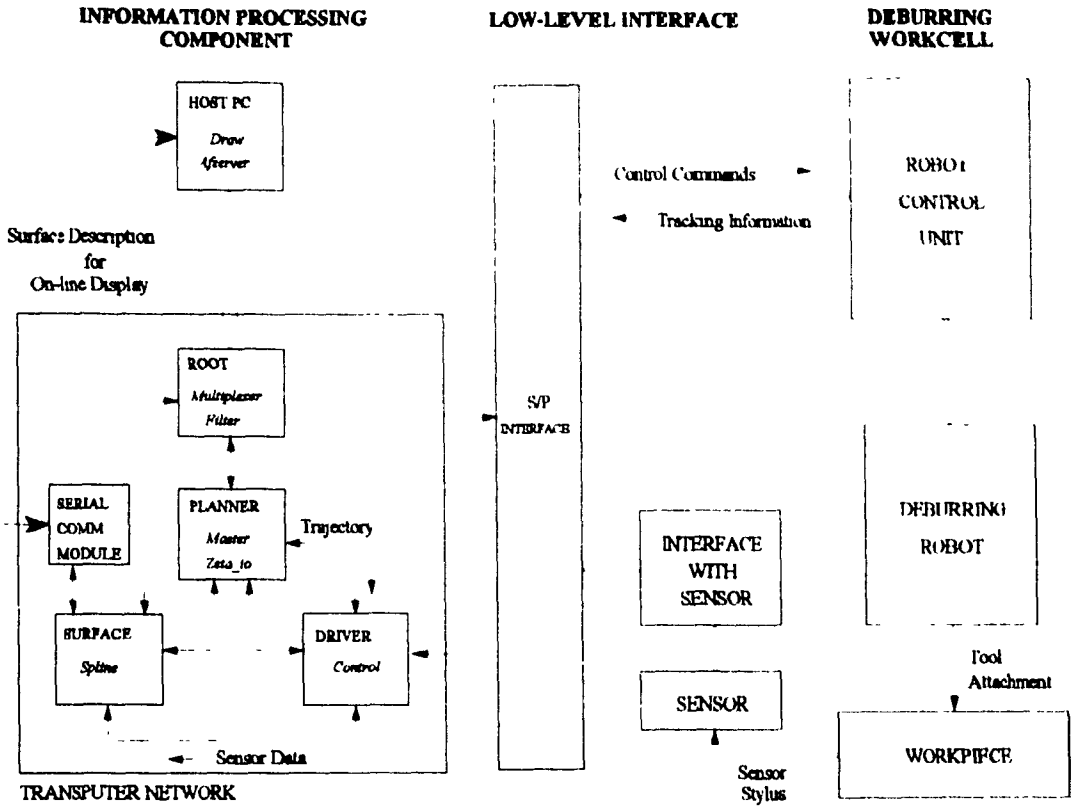


Figure 1.6: Schematic from Ayyadevara [13] of the Yamaha Zeta-1 Robotic Deburring Workcell Controller at Concordia University's Centre for Industrial Control

diagram is given by Ayyadevara [13], but essential features will be briefly outlined here The host PC executes the transputer file server *Afserver*. The task *Master*, executing on PLANNER, responds to choices made by the user in *Afserver* However, *Master* does not directly access workcell hardware, but rather calls *Control* executing in DRIVER which communicates with devices of the workcell

When the workcell is powered up, a homing sequence must first be performed to initialize optical shaft encoders of each joint This homing sequence leaves the robot at a predefined location called the *ready position* The Yamaha controller computes all subsequent

displacements of the robot based on this initialization Following the homing sequence, the Initialization routine must be selected on the main menu of *Afserver* This selection sends a command to *Control* which resets low level interface chips so that the robot is understood to be at the ready position Following homing and initialization, other selections may be made from the main menu of *Afserver*.

The *Change Display Default Setup* option provides a submenu which allows alteration of Automated Digitizing parameters without recompiling These parameters include probing clearance and interference, approach and retract speeds, digitizing grid size and stepsize, and digitizing start position Automated Digitizing initiates a digitizing operation in which component elevations are collected with the robot navigating in real time over the component surface The software uses extrapolation to estimate each successive elevation, and therefore does not require *a priori* knowledge of component geometry

The *Demo Routine* option causes execution of a series of path commands which demonstrate motion capabilities of the robot The *Probe Test* option reads the digital probe every 35 ms for a user-specified interval while the robot is motionless, to allow probe interface verification The *Toolpath Execution* option prompts the user for a script file containing a toolpath given as a sequence of points and corresponding speeds This sequence of points and speeds may also include an optional special character which serves as a digitizing sample-on-request command

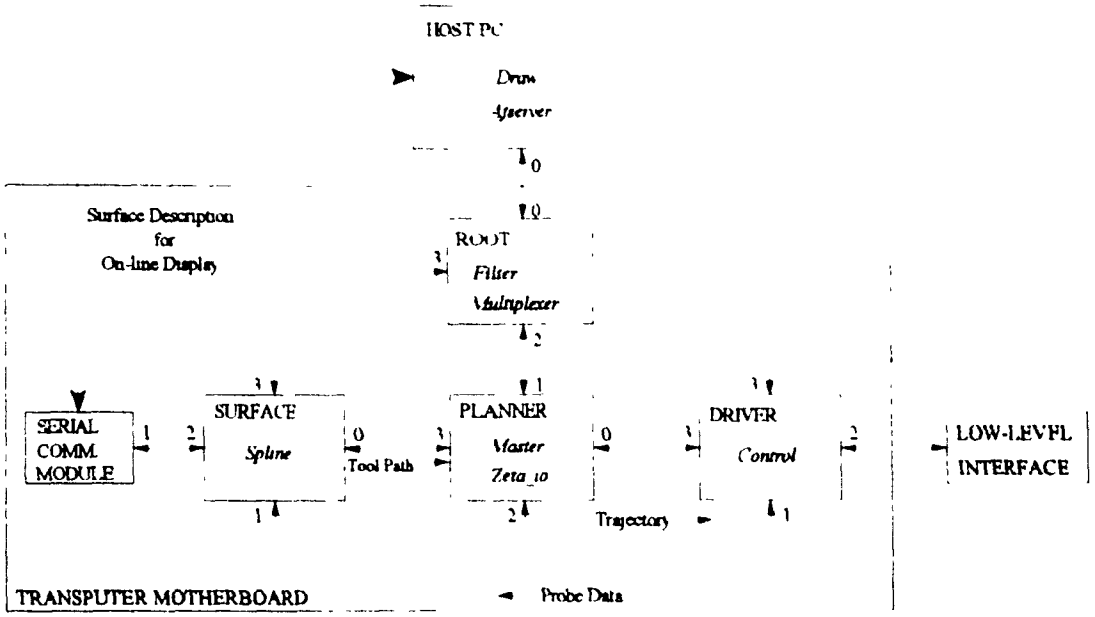


Figure 1.7: Schematic from Ayyadevara [13] of Software Layout for the Workcell Controller

1.4 Literature Review

1.4.1 Digitizing Processes and Automated Inspection

Treywin and Edwards [21] report extensive use of coordinate measuring machines (CMMs) by both aerospace and automotive industries, and point out that developments in CMM technology are progressing along two fronts guidance and sensing Spyridi and Requicha [22] report on the continuing development of more efficient path planning (guidance) strategies for dimensional inspection *American Machinist* [23] recently reported commercial availability of optical non-contact (sensing) probes for use on CMMs. Increased digitizing speed and accuracy are thus the primary goals of this technology

Drawbacks of CMMs include their high capital cost and the necessity of machining toolpath compensation for fixturing errors (arising from removal of digitized components from the CMM and subsequent setup on a machine tool worktable) Cheng *et al* [24] report development of an automated fixturing error compensation method for the Yamaha Zeta-1 robotic deburring workcell In the present thesis, this issue is avoided by development of a digitizing capability for the workcell The above-described fixturing errors are eliminated if the same machine is to be used for both digitizing and machining

Automated inspection with coordinate measuring machines is carried out using a variety of digitizing devices which fall into two main categories contact and non-contact probes Factors affecting selection of these devices include desired speed and accuracy, as well as access restrictions over more intricate component surfaces Contact probes may be regarded as representing the conventional approach to displacement measurement, and are manufactured by several companies, including Ono Sokki Ltd [16] and Mitutoyo Ltd [25] The Ono Sokki model EG-225 digital linear gauge shown in Figures 1 4 and 1 5 is typical of these devices To record a distance, the springloaded stylus must displace itself against a workpiece Resulting measurements are converted into a digital electrical signal transmitted to the computer or other monitoring equipment

Accurate non-contact displacement measurements may also be carried out with devices based on inductive proximity sensing and laser-optical scanning. Cheng *et al.* [24] have demonstrated use of a Kaman [26] inductive proximity sensor to perform precision displacement measurements with an automated deburring workcell This device is compact (length 30 mm, diameter 10 mm) and accurate (repeatability 0 01 mm), but has limited range

(5 0 mm) and can only be used to sense the surfaces of metallic conductors Huang and Zheng [27], Brown and Newton [28], and Zhuang, Zhang, Jiang, Li and Zhang [29] describe laser-optical scanning techniques involving use of a laser to illuminate a grid or set of parallel lines on a surface, while two or more CCD cameras record the scene Recorded images are input to a computer, which performs triangulation computations establishing coordinates of points on the surface

Kalpakjian and McKee [1] have documented the use of automated inspection in manufacturing to perform closed-loop process control for dimensional error minimization Ideally, corrective modifications to manufacturing processes are applied before errors exceed design tolerances, thereby eliminating defective production output For cases where online inspection is not possible, *a posteriori* inspection of all production output may be performed, coupled with corrective action or rejection to maximize yield without compromising quality

1.4.2 Segmentation, Pattern Recognition and Image Processing

Image segmentation is the process whereby regions of pixels are identified as distinct from one another and grouped accordingly Brice and Fennema [30] applied a segmentation technique to 120 × 120 digitized pictures with 16 levels of gray Results served as input to a heuristic decision tree which performed recognition by comparison with several predefined object classes, such as wedges, cubes, wall, floor, etc Tenenbaum and Barrow [31] reported an integration of segmentation and interpretation processes for scene analysis Their technique utilizes input from a knowledge base to make inferences about the interpretation of regions, so that merging of regions is guided in accordance with their possible interpretations

Hoffman and Jain [32] proposed a 3D range image analysis procedure based on segmentation Their approach involves sequential application of segmentation followed by classification.

Chen [33] defines the problem of pattern recognition as that of developing analytical techniques for associating observed data with one or more predefined classes The main approaches to pattern recognition are statistical, fuzzy, neural, or hybrid combinations of these Chen also gives a comprehensive treatment of the statistical approach, including feature extraction, recognition systems, and selected applications Bezdek and Pal [34] have edited an anthology of the more important basic research on fuzzy pattern recognition, starting with a reprint of the original paper on fuzzy sets by Zadeh [35] (this paper is discussed further in Section 1 4 3) Shalkoff [36] has documented pattern recognition using neural networks, and summarizes other approaches as well

Pavlidis [37] defines image processing as dealing with problems in which both input and output are pictures Typical applications include noise removal (filtering), data compression, and contrast or colour enhancement Schalkoff [36] divides image processing into three main areas - enhancement, restoration and analysis There is considerable overlap between image processing analysis and pattern recognition The extensive volume and diversity of image processing research is alluded to by Pratt [38], who also points out dramatic recent increases in industrial applications

1.4.3 Applications of Fuzzy Logic

Zadeh [35] introduced the concepts of fuzzy sets and fuzzy set theoretic operations. Fuzzy logic constitutes a generalization of conventional crisp or nonfuzzy set theory whereby set

membership is no longer restricted to pure inclusion and exclusion With partial or fractional set memberships possible, quantitative representation may be achieved for non-crisp relationships existing between real world variables in a way that reflects human understanding

Lee [39] summarizes fuzzy logic in the context of control theory, with specific treatment of fuzzification, fuzzy reasoning and defuzzification Control systems are a natural application for fuzzy logic, since logic is essentially the process of decision making - an essential characteristic of controllers Methods for automatic tuning of fuzzy controllers are reported by Koivo and Viljamaa [40], Zheng *et al* [41], and many others

Fuzzy logic-based control can be of particular benefit in applications where the system dynamics are poorly understood In such cases, simple fuzzy controller designs may be based on human operator experience using natural language Fuzzy modelling techniques for use in process control are reported by Garcia [42], Chen *et al* [43], and many others Important work in the area of fuzzy system identfication and control has been carried out by Sugeno *et al.* [44, 45]

Fuzzy logic also lends itself naturally to applications in pattern recognition, since decision making is at the core of associative processes Bezdek and Pal [34] give an overview of the extensive body of research which exists in this field Methods of classifier design and feature analysis for use in handwritten character recognition are reported by Gader and Keller [46], Poon *et al.* [47], and many others In summary, it may be stated that fuzzy logic has potential for application in diverse areas where heuristic or qualitative human understanding is translatable to a set of simple operative rules

1.4.4 B-splines, Surface Fitting and Optimization

This section presents an overview of important recent work in the areas of surface modelling, specifically regarding B-splines, surface fitting and related optimization techniques Farin [4] defines a B-spline as any sequence of piecewise continuous polynomial functions These functions must pass through a sequence of knot points which they are meant to interpolate However, an interpolating polynomial surface function cannot be expected or desired to pass *exactly* through all points in an experimentally-sampled elevation database, due to the existence of sampling errors Added complications arise if the data are not distributed in a perfectly uniform rectangular grid, or if an otherwise uniform grid has one or more points missing (that is, unknown) at arbitrary locations

Lancaster and Salkauskas [5] present a variety of surface fitting methods designed for these types of problems In particular, the methods of "weighted least squares" and "interpolating moving least squares" represent a complete class of solution to the problem of interpolating smooth functions to scattered data Hoppe *et al.* [48] propose a general method for automatic reconstruction of surface models from scattered range data based on use of piecewise-smooth subdivided surface representations Fang *et al* [49] report a surface fitting method in which quadric approximating surfaces are generated to filter out measurement errors. Bicubic splines are then computed to model the quadric surfaces. Chen and Medioni [50] report a method for surface approximation based on use of a dynamically inflated balloon model fitted to a triangulated mesh

The aforementioned surface fitting techniques may also be regarded as multivariate optimization schemes from which sets of optimal interpolating polynomial coefficients are

determined Rao [51] defines optimization as the determination of a design vector which extremizes an objective function subject to one or more constraints In the case of surface fitting, the design vector is a set of polynomial coefficients defining a surface function, the objective function is an error index measuring surface fit accuracy, and the constraints are the sampled surface data

Hochbaum and Seshadri [52] report on the development of a practical nonlinear optimization algorithm based on successive piecewise linear approximations of the objective function In this work, convergence results are shown for a variety of objective functions Carter [53] reports on methods for estimation of the desired accuracy for nonlinear optimization schemes, demonstrating that proper accuracy estimation can result in order-of-magnitude savings in computation time Fox [54] reports development of an interactive nonlinear optimization software package designed for teaching senior undergraduate mathematics courses.

1.4.5 Automation of Robotic Deburring and Machining Processes

Gillespie [55] distinguishes robotic deburring from primary machining in several ways Deburring is the removal of burrs from surface edges whereas machining involves removal of the bulk of material from component surfaces The magnitude of deburring cutter forces is significantly less than those encountered in machining Abrasive brushes and small rotary files are used in deburring, whereas machining uses a variety of hardened tooling such as end mills, flank mills and flycutters. Deburring robots are also suitable for surface polishing or other light refinishing operations

Duelen *et al* [56] address problems of design, implementation and performance verification for a hybrid position / force controller on an advanced robotic deburring system The authors report a successful experimental demonstration of this technology for both the servo and motion planning control levels Liu and Asada [57] have developed an adaptive control scheme for deburring robots whose design is based on a set of teaching data taken from human demonstrations In this work, an associative mapping between burr characteristics and human responses is constructed using neural networks, and this mapping is then used as a basis for path planning on subsequent components Smoothness of surface finish is found to be greatly improved by this method Bone and Elbestawi [58] report development of an automated robotic edge deburring system which corrects the deburring path online using force and vision sensors to control chamfer depth This type of system provides an alternative to controllers that require a detailed dynamic model of the deburring process

Ulsoy and Koren [59] review important research contributions for control of machining processes, particularly optimal and adaptive control methods applied to turning, milling, drilling, and grinding Masotti and Bombardi [60] report the development of a prototype system for computer aided process planning This software performs automatic generation of machining toolpath instructions and simulates all operations involved in the production of a part Tang and Chang [61] report the development of a dynamic cutting-simulation system for accurate prediction of cutting forces along programmed toolpaths

Thyer [9] observes that although repeatability is commonly specified, few vendors of robotic machine tools state the accuracy of their products Hayward [62] offers the following

definition of repeatability "the half-range random uncertainty of a typical measurement made by the instrument under specified conditions of operation" By contrast, accuracy is the ability of an instrument to tell the truth, rather than merely stick to the same story Good repeatability is therefore no indication of good accuracy A quantitave definition for accuracy is also given by Hayward which combines repeatability and half-range systematic uncertainty Surprisingly, there is no single universally accepted formula used in defining accuracy

1.5 Scope of the Thesis

1.5.1 Problem Definition

This thesis details work done by the author as a member of a team working on an ongoing strategic research and development project funded by NSERC at Concordia University's Centre for Industrial Control (CIC) This work described in this thesis deals with the following:

- Development of a scheme to perform autonomous recognition of defective features on the surface of machined workpieces
- Development of a technique to perform computational replacement of rejected surface features as a prior step to corrective machining
- Development of an automated digitizing capability for the Yamaha Zeta-I deburring robot
- Development of algorithms and demonstration of the techniques in the preceding steps using digitized data collected by the Yamaha Zeta-I

deburring robot

The fuzzy logic-based scheme for autonomous recognition of defective surface features has been developed for use on the robotic deburring workcell This recognition scheme analyses the digitized database representing surface elevations on a damaged component to identify extraneous features such as beads of weld, pits, scratches, cracks, and the like In the present work, feature rejection is thus performed without comparison to original design geometry This type of analysis is useful for reconditioning surfaces which have acquired slight warping deformations that prevent use of original design data as a basis for comparison For example, aircraft engine gas turbine rotor blade assemblies may possess axial blade tip deformations as large as 2 0 mm (0 079 in)

The work presented in this thesis also reports development of a method for computational replacement of rejected elevations in the digitized database of the damaged component to give a desired geometric model for the reconditioned component This new geometric model provides a basis for corrective machining toolpath generation, and is obtained by use of surface fitting The surface fitting scheme developed for this purpose performs multivariate optimization to interpolate geometric trends in nonrejected digitized elevations. It should be noted that the algorithms developed for feature recognition and surface reconditioning are by no means limited to use on the Yamaha Zeta-1 deburring robot, but may be applied to any suitably-equipped automated machine tool

The Zeta-1 deburring robot as supplied by Yamaha is capable of teach-and-playback mode using joystick or keyboard input, and can also be manually jogged from these input

devices. However, the toolpath cannot be modified in real time Ayyadevara [13] carried out development of an external controller for the Zeta-1 at CIC which is capable of modifying the toolpath in real time The original Yamaha controller has been retained to perform servo-level control, while the transputer network allows for real-time communication of machining or digitizing path instructions The new external real-time controller developed by Ayyadevara [13] can process toolpath data from a script file generated offline to machine a workpiece A digital probe is interfaced with the external controller to perform real-time periodic sampling while the probe is dragged over the surface

The work presented in this thesis reports addition of two more capabilities, namely, sample-on-request for offline digitizing path generation, and automated digitizing Sample-on-request permits use of a four-step digitizing cycle consisting of the following

(1) approach & intersect component surface,

(2) sample-on-request,

(3) retract to a predefined clearance above surface,

(4) index to position probe tip above next digitizing location

Sample-on-request is often preferable to the aforementioned periodic sampling because it does not require the probe tip to be dragged over a component surface Probe tip wear is thus reduced, and recalibration required less frequently Automated digitizing integrates use of the four step sample-on-request digitizing cycle with real-time automated probe navigation over the component surface. This navigation scheme is based on extrapolation of local geometric

trends, from which an expected range is computed for successive elevation *This approach differs from the work of* Ayyadevara [13] *in that a digitizing path need not be prepared offline for each and every component* In the present work, this is done automatically and is independent of the surfaces considered

1.5.2 Layout of the Thesis

Chapter 2 discusses development of two digitizing strategies for the Yamaha Zeta-1 deburring workcell Chapter 3 presents a fuzzy logic-based algorithm for feature recognition. Chapter 4 presents an offline scheme for computational surface reconditioning, intended as a prior step to corrective machining. In Chapter 5, experimental verification of the methods described in Chapters 2, 3 and 4 are provided. Conclusions and recommendations for future work are given in Chapter 6

CHAPTER 2 DIGITIZING STRATEGIES

2.1 Introduction

2.1.1 The Need for Digitizing in Modelling of Component Surfaces

Industrial uses of digitized data representing the geometry of machined components include inspection, reconditioning, and so-called "reverse engineering" Digitizing is also useful in reproduction of accurate geometric models for film animation, commercial art, and the like For inspection or reconditioning purposes, digitizing provides the necessary data to perform computer-aided assessment of the condition of a surface "Reverse engineering" concerns determination of a geometric model for components produced prior to the advent of computer-aided design technologies, or by one's competitors

Digitizing is useful in inspection of new components to verify conformance to design specifications as well as for secondary machining (deburring) path planning Analysis of digitized data is based on point-by-point comparisons between original design and actual digitized geometry Used components may require comparison-based detection of accumulated features such as beads of weld, cracks, scratches, dents, pitting, and the like In addition, assessments of surface roughness and warping may be made for used components Identification of such characteristics is necessary for component rejection, or for generation of a corrective machining toolpath to recondition the component

"Reverse engineering" techniques require determination of a geometric model for an existing component, which may then be reproduced in an automated manufacturing environment, or subjected to computerized performance analysis Digitizing for "reverse

engineering" thus leads to reduced production costs for components originally conceived using older non-computerized technologies In addition, subsequent design changes (if necessary) are greatly facilitated

2.1.2 Methods of Digitizing

This section first describes major steps required in digitizing methods, secondly outlines distinct methods assembled from these steps for use on the CIC deburring workcell, and finally considers their advantages and disadvantages There are four categories of steps described in the following sections, from which four distinct digitizing methods have been assembled in the present work These four categories are as follows

- pre-processing,
- path generation,
- digitizing path types;
- sampling techniques

Pre-processing steps consist of modelling error analysis and coordinate transformations Modelling error analysis is the process whereby a digitizing stepsize recommendation is computed prior to the commencement of digitizing This stepsize is the spacing between neighbouring points in a uniform rectangular grid of sampled elevations Original geometric design data is required as input to this analysis, along with specification of the desired maximum error resulting only from splining over the sampled data, as sampling errors are

attributed to the repeatability of the workcell and digitizing probe Section 2 4 1 describes this process in detail In the present application, the workcell chosen has a repeatability of ± 0 1 mm (0 004 in) and the digitizing probe has a resolution of ± 0 001 mm (40 μin) Transformations are necessary to convert geometric data back and forth between coordinate systems oriented relative to the workpiece (workpiece coordinates) and the robotic workcell (robot coordinates) In practice, components are designed using the former and manufactured using the latter More details on required coordinate transformations are given in Section 2 4 2

Two alternatives exist for digitizing path generation off-line and on-line techniques Off-line path generation refers to preparation of a complete digitizing path prior to the commencement of digitizing This approach requires that an accurate database representing nominal component geometry be available A detailed discussion of off-line path generation is given in Section 2 2 On-line path generation consists of computing extrapolations to predict each successive elevation, based on geometric trends encountered up to that point These elevations are used to generate the digitizing path in real time No *a priori* knowledge of component geometry is required for this approach A detailed discussion of on-line path generation is given in Section 2 3

Two types of digitizing path have been used by the CIC Deburring Group one based on probe "dragging" (the reader is referred to Ayyadevara [13]), and another involving an iterative digitizing motion cycle described in this thesis Probe "dragging" refers to the probe tip being moved so as to trace over the component surface with a slight interference, that is, maintaining contact at all times The iterative digitizing motion cycle used consists of four

steps: approach, dwell / sample-on-request, retract, and index This motion cycle is described in detail in Section 2 3 2

Two sampling techniques have been used by the CIC Deburring Group: periodic sampling (the reader is referred to Ayyadevara [13]), and sample-on-request, developed in this thesis Periodic sampling involves recording probe displacements at a preset frequency (the Ono Sokki model EG-225 digital linear guage used for all digitizing in this thesis transmits new measurements every 35 ms) With the sample-on-request technique, the probe provides data for the workcell only when requested.

The four digitizing methods developed in this thesis and also by Ayyadevara [13] are as follows, in order of increasing sophistication

(i) Off-line path generation with probe "dragging" and periodic sampling (Ayyadevara [13]),

(ii) Off-line path generation with probe "dragging" and sample-on-request,

(iii) Off-line path generation with iterative digitizing motion cycle and sample-on-request,

(iv) On-line path generation with iterative digitizing motion cycle and sample-on-request.

Advantages of method (i) are the simplicity of off-line programming, and the high speed at which surface elevation data may be obtained Disadvantages of method (i) are:

- high probe tip wear due to dragging (creating the need for frequent recalibration),
- possible binding of probe tip against small surface irregularities resulting in damage to probe,
- excessive amounts of data collected,
- difficulty of obtaining a *uniform* grid of digitized points,
- nominal surface geometry must be known

Method (ii) is preferable to method (i) because it allows control over the sampling process, so that only as many points as necessary are collected In addition, elevations may be collected in a uniform grid. Method (iii) also depends on off-line path generation, but sampling events must be explicitly specified in the input file, and the probe tip is not dragged over the component surface. This method has the following advantages

- off-line programming is simple,
- does not require probe tip to be dragged over the surface probe tip wear is thus minimized;
- can collect elevations in a uniform rectangular grid or any other desired pattern, i e , the exact number and location of points to be sampled,
- allows digitizing to proceed faster than with on-line path generation

The common disadvantage of methods (i), (ii) and (iii) is their inability to digitize unknown surfaces. This capability exists only for method (iv), where the prediction of successive

elevations is based on extrapolation This process is described in detail in Section 2 3.2

The discussion of digitizing strategies in this chapter focusses on work done with the Yamaha Zeta-1 automated deburring workcell at Concordia University's Centre for Industrial Control Off-line and on-line digitizing path generation strategies developed for this workcell are described in Sections 2 2 and 2 3 respectively The off-line approach is covered first, since on-line path generation represents a further refinement of this method Modelling error analysis and required coordinate transformations are described in Section 2 4

2.2 Digitizing with Off-Line Path Generation

2.2.1 Definition and Requirements

This section outlines the necessary information required for off-line path generation Off-line path generation consists of determining a digitizing path for the robot-driven probe tip in robot coordinates The undisplaced probe tip is located at the *machining point* of the Yamaha Zeta-1 robotic deburring workcell, and the probe stylus axis is coincident with the *machining axis*, as shown in Figure 2.2. As mentioned in Section 2 1 2, digitizing with off-line path generation may be carried out on the CIC deburring workcell in the following ways:

- off-line path generation with probe "dragging" and periodic sampling;
- off-line path generation with probe "dragging" and sample-on-request;
- off-line path generation with iterative digitizing motion cycle and sample-on-request.

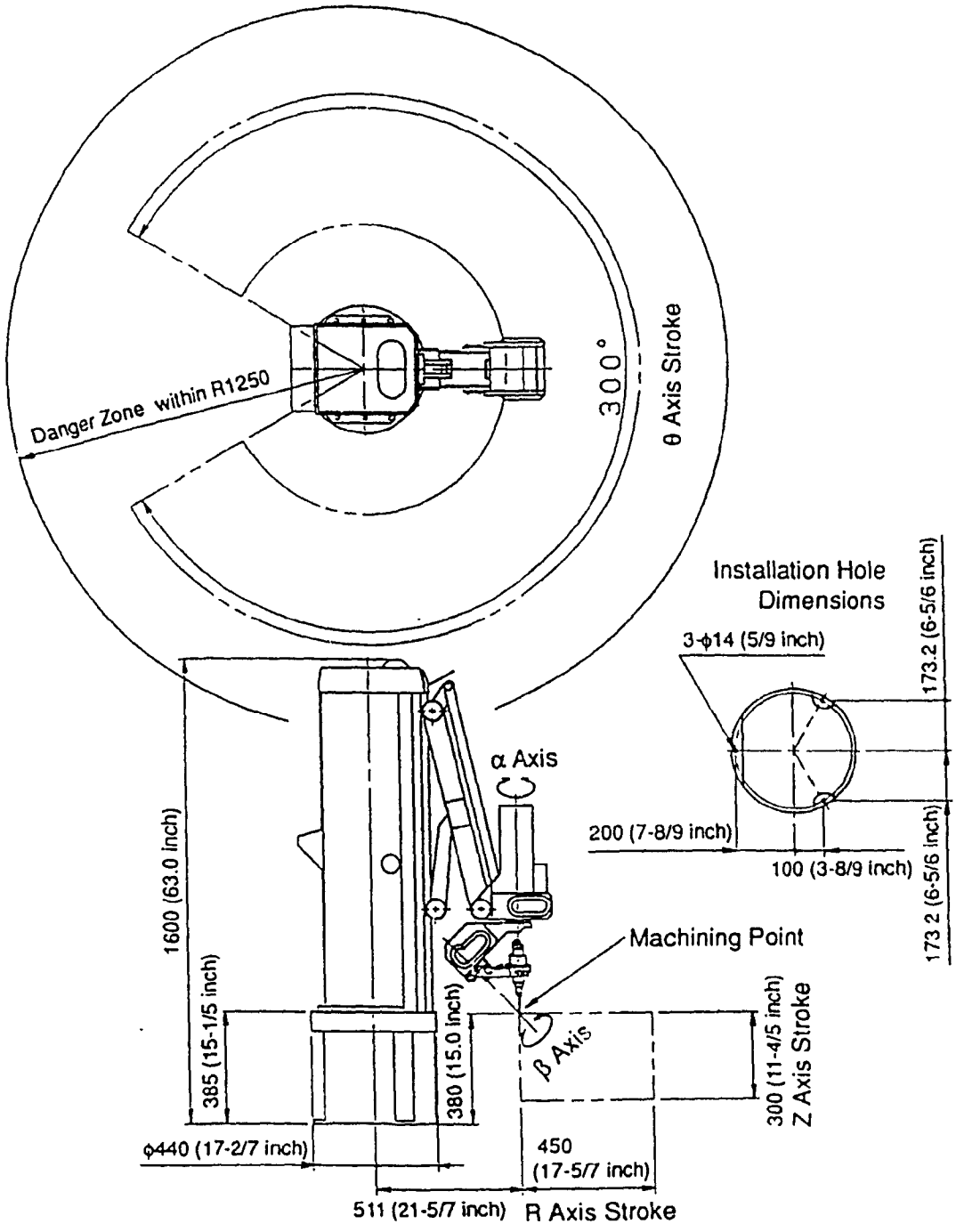


Figure 2.1: Schematic of Yamaha Zeta-1 Deburring Robot [11] (with machining point and tool axis shown)

Primary requirements for these digitizing methods are knowledge of nominal component geometry, and of the coordinate transformation converting this data to robot coordinates Planning of the desired digitizing path is carried out entirely in workpiece coordinates, and afterwards converted to robot coordinates Conversion from workpiece to robot coordinates is followed by generation of required robot joint displacements by the workcell controller. More detail on coordinate transformations is given in Section 2 4 2

2.2.2 Path Generation Algorithm

This section describes the method developed in this thesis for off-line digitizing path generation developed for the Yamaha Zeta-1 robotic deburring workcell A description is first given of special considerations pertaining to initial and final digitizing path segments The iterative digitizing motion cycle is next presented, along with justification (based on safety considerations) for selection of speeds used in each path segment Information necessary to specify path segments is finally summarized, along with details of the sample-on-request process.

Every digitizing path segment consists of the workcell machining point (undisplaced probe tip) moving in a straight line, with the probing axis held vertical By convention, digitizing paths are generated using a common start and finish point, termed the *ready position*. The workcell thus starts from and returns to the ready position at the beginning and end of digitizing runs Ready position coordinates explicitly appear only at the end of a digitizing path, however, since only then does it constitute a destination for the machining point. In the initial path segment, the probe tip moves from ready position to a safe distance

(50 mm (2.0 in) for the Yamaha Zeta-1 deburring workcell) directly above the first digitizing point on the (nominal) component surface Likewise, the final path segment commences from a safe distance (50 mm (2 0 in) for the Yamaha Zeta-1 deburring workcell) above the final digitizing point, and returns the machining point to the ready position Both initial and final path segments are executed at a relatively high speed (40 mm/s (1 6 in/s) for the Yamaha Zeta-1 deburring workcell), since they occur farthest from the workpiece The second digitizing path segment consists of a vertical descent at a relatively moderate speed (10 mm/s (3.9 in/s) for the Yamaha Zeta-1 deburring workcell) to a distance equal to the probe clearance (shown in Figure 2 2) above the surface, after which the first low-speed close approach of the iterative digitizing cycle commences

The iterative digitizing cycle consists of a sequence of sampling events in which the probe stylus is repeatedly guided to approach, intersect, dwell / sample-on-request, retract and index over the component surface Figure 2 2 shows a trajectory profile for several cycles of digitizing motion. User-specified vertical probing interference Δh_i and clearance Δh_c distances allow for approximately constant probing intersection, retraction and indexing heights over the entire surface during a digitizing run During off-line digitizing path generation, probing interference and clearance values are used to shift the nominal surface elevation data. These two parameters are therefore chosen based on the maximum expected variation of actual surface elevations relative to nominal data. Digitizing with off-line path generation is illustrated in the flowchart of Figure 2 3

Based on considerations of safety, a close approach speed (4 mm/s (0 16 in/s) in the present case, as shown in Figure 2 2) is selected, along with indexing and retract speeds (10

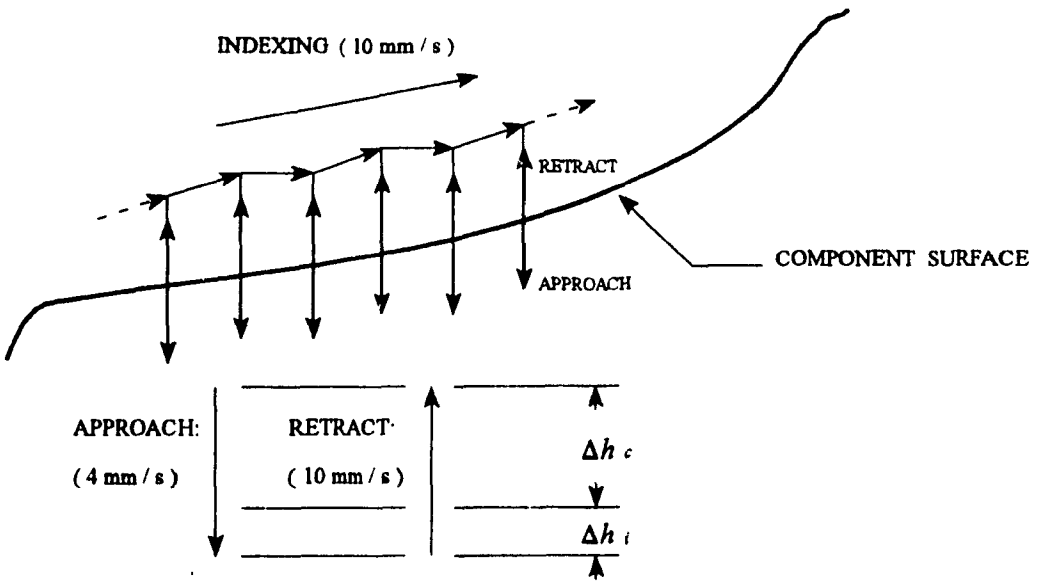


Figure 2.2: Schematic of iterative digitizing motion cycle, showing probe clearance Δh_c, and probe interference Δh_i

mm/s (0 39 in/s) in the present case, also shown in Figure 2 2). A low speed for close approach is selected for two reasons. Firstly, the sample-on-request following close approach will thus occur with minimal transient vibrations in the robot structure. This is a desireable precondition for obtaining accurate measurements Secondly, close approach path segments are the most critical portions of the digitizing cycle, since the robot is at that time moving towards its closest contact with the workpiece. Chances of accidental damage to the robot structure are thus minimized by selection of a low close approach speed

To move the robot from point A to point B, one needs to specify seven parameters, namely X, Y, Z, α, β, V, S. These parameters are defined as follows:

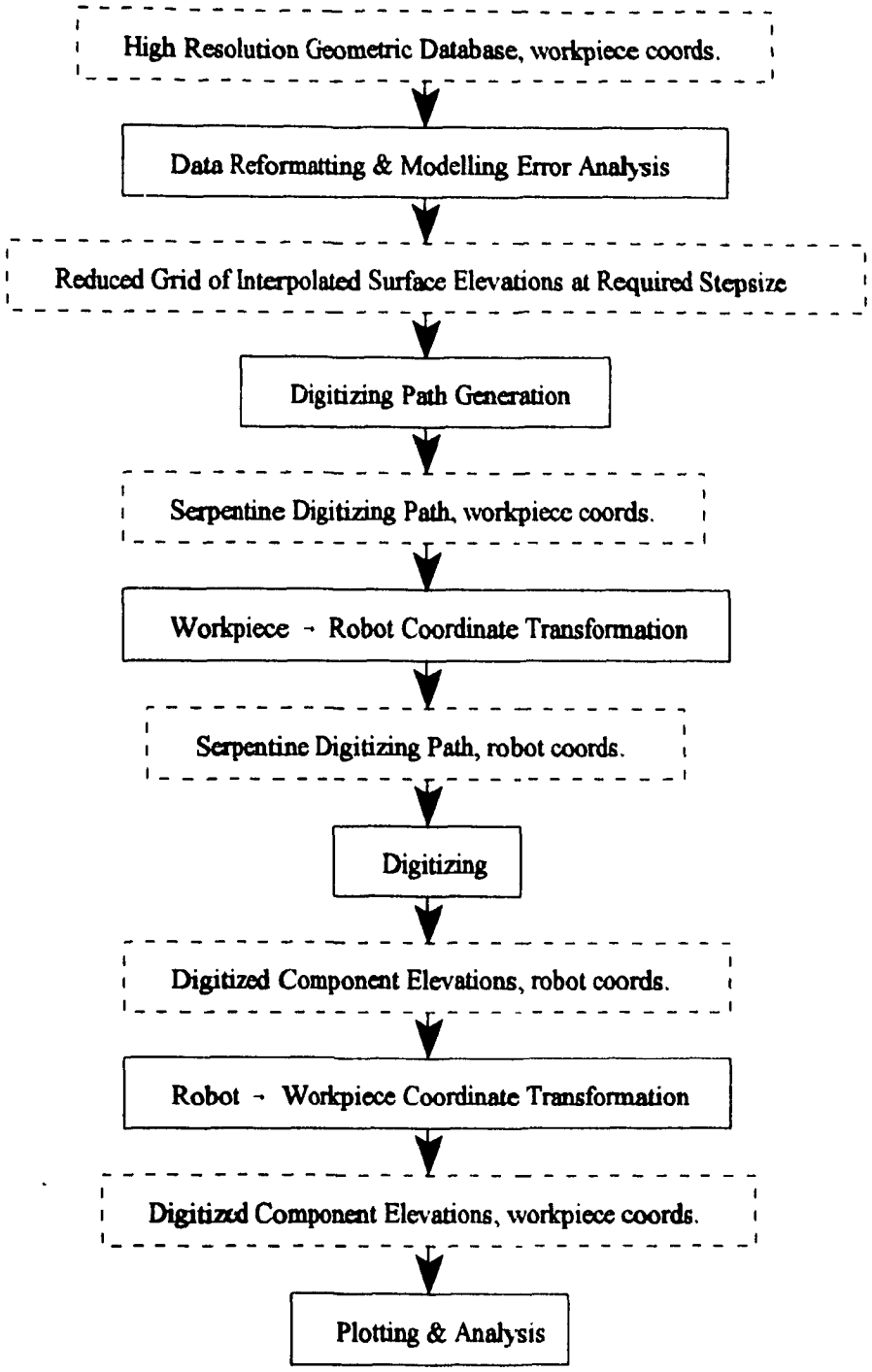


Figure 2.3: Flowchart of Digitizing with Off-Line Path Generation

X machining point X-axis displacement, robot coordinates, mm,

Y machining point Y-axis displacement, robot coordinates, mm,

Z machining point Z-axis displacement, robot coordinates, mm,

α robot α-joint displacement, degrees,

β robot β-joint displacement, degrees,

V machining point speed, mm/s,

S optional sample-on-request flag.

The first three displacements X, Y, Z listed above are specified as Cartesian coordinates. The X and Y displacements are converted for internal use by the workcell controller into the robot system of cylindrical coordinates Machining point speeds V represent the desired speed en route to the set of joint displacements given in the current set of parameters For sampling events, a machining point speed of zero is specified, and the previous set of joint displacements is repeated A *dwell* occurs in conjunction with each sample-on-request, during which the robot is maintained stationary for 250 milliseconds prior to sampling. This allows time for dissipation of transient vibrations present in the robot structure, and thus contributes to improved digitizing accuracy

2.3 Digitizing with On-Line Path Generation

2.3.1 Definition and Requirements

This section presents a description of the principal characteristics required in an automated

robotic system to perform digitizing with on-line path generation Following this, an outline is provided for the types of scenario in which on-line digitizing path generation becomes a necessary or preferable alternative to off-line methods This section concludes with a summary of input information required for performing digitizing with on-line path generation using the Yamaha Zeta-1 robotic deburring workcell

Digitizing with on-line path generation is the process of sampling surface geometry without *a priori* knowledge of the surface Robotic systems performing such an operation must have the capability to perform autonomous navigation over surfaces in real-time Extrapolation of local geometric trends is therefore necessary to predict an approximate location for each successive digitized point

It is often desired to refurbish components which have undergone slight warping deformations. As pointed out in Chapter 1, Section 1 2 1, such deformations often do not significantly affect component performance, however they render original design data as an unacceptably inaccurate model of reconditioned geometry The real motivation for refurbishment in such cases is invariably the existence of localized surface damage, for example: scratches, cracks, pitting, beads of weld, and the like This type of damage can seriously affeet both performance and operational safetv The major consequence of slight warping deformations is therefore the constraint imposed on methodologies for reconditioning automation.

Off-line digitizing path generation becomes problematic at best for surfaces which deviate significantly from original design data Occasionally, original design data for a component is simply not available Another critical factor for some applications is the reduced

processing time resulting from digitizing with on-line path generation, since with this method data need not be read from a file For any of these situations, use of some alternative to off-line digitizing path generation becomes necessary It is therefore desired that a capability should exist for automated navigation of a digitizing probe over a component surface in real time.

In the present research, an automated digitizing scheme has been developed for use on the CIC deburring workcell As the name implies, this scheme does not require a digitizing path derived off-line from nominal component geometry However, the present scheme is based on provision of such input information as is necessary to adequately define a desired grid of elevations to be sampled In particular, the *xy* projection of the grid must be specified in robot coordinates, and elevations of the first two digitizing points must be provided.

Specification of the *xy* projection for the desired digitizing grid is accomplished with the following information:

(i) *xy* coordinates (in the robot system) of the start point,

(ii) the number of desired points along each of two perpendicular digitizing directions in the horizontal plane,

(iii) a rotation angle for grid orientation relative to the robot coordinate system.

In addition, elevations must be specified for the first two digitizing points. The first of these two elevations allows the workcell to make a reasonably fast initial approach to the first

digitizing point on the component surface The second of these two elevations allows the workcell to begin computing approximate local geometric trends necessary for automated probe navigation These two elevations are therefore measured using the robotic workcell in manual control mode prior to commencement of automated digitizing

2.3.2 Extrapolation-Based Prediction of Elevations

The essential feature of any automated digitizing scheme is the robotic system's ability to navigate over an *a priori* unknown component surface To do so, predictions must be made after each successive sampled elevation as to the expected elevation of the next point In the present work, a linear extrapolation is computed using the two most recent previously encountered elevations A factor of safety is added to this extrapolation in the form of an offset, biasing the prediction slightly upwards This linear extrapolation is computed as follows

$$\begin{aligned} z_N &\approx z_{N-1} + \frac{dz}{dx}\,\Delta x + z_s \\ &\approx z_{N-1} + \frac{z_{N-1} - z_{N-2}}{x_{N-1} - x_{N-2}}\,(x_N - x_{N-1}) + z_s \\ &\approx 2z_{N-1} - z_{N-2} + z_s \end{aligned} \tag{2.1}$$

where z_N is the estimated elevation, z_{N-1} is the last sampled elevation, z_{N-2} the second last sampled elevation, and z_s is the offset added as a safety factor Cancellation occurs in the derivative term above, since the grid is assumed uniform

The cycle of motion for digitizing of each point appears the same as for off-line path

generation, with user-specified clearance and interference as shown in Figure 2 2 If the predicted elevation turns out to be too high the probe will not contact the surface in the expected range In this case, the system begins a loop during which the probe is iteratively moved farther downwards and another sample taken until a displacement is finally registered. With the system properly tuned for a reasonably well-conditioned surface, the majority of sampling points do not require execution of this loop Those sampling events which do miss the component surface on the first try rarely require more than one descend-and-sample iteration before the surface is found It should be noted, however, that this depends on the geometry of the surface that is being digitized It should be mentioned that an alternative to the above-described method would be to allow the probe to descend continuously while monitoring probe data, and to stop this descent as soon as the probe indicates a non-zero reading However, the existing architecture of the workcell controller does not support the implementation of this approach

Tuning this algorithm consists of selecting the amount of upward shift to be added to the linearly extrapolated elevation predictions If this shift is set too large, the system will miss contacting the component surface (on first try) at a high proportion of grid points This will in turn necessitate more frequent use of the descend-and-sample loop, hence the digitizing process will proceed more slowly However, with a large upward shift there is less danger of making an elevation prediction which is too low, so the probability of a probe crash is reduced. Note that the emergency stop switch described in Chapter 1, Section 1 3 3 is not fullproof it only prevents excessive axial (vertical) motion, not lateral (horizontal) motion motion It is therefore desired to set the value of upward shift just large enough so that probe

crashes are prevented In the present work, an upward shift of 5 0 mm has been used

2.4 Pre- and Post- Processing of Geometric Data

2.4.1 Modelling Error Analysis

The question naturally arises as to what digitizing stepsize ought to be used for a given surface Too few sampled points will result in an unacceptably coarse surface representation, whereas too many will result in a prohibitively lengthy digitizing run An unacceptably coarse surface representation will cause large surface modelling errors, and poor reconditioned surface quality Factors affecting choice of a proper stepsize include the following

(i) Surface geometry If a surface is highly curved and convoluted, it becomes necessary to digitize at very fine resolutions to allow for accurate geometric modelling Similarly, if a surface contains discontinuities or extraneous features, surface modelling techniques may not be able to converge to any useful representation if too few samples are taken Surface geometry is thus one of the critical factors affecting digitizing stepsize selection

(ii) Desired modelling accuracy Modelling accuracy serves to set an upper bound on allowable digitizing stepsize. Stepsizes which are too small will also satisfy a given modelling accuracy specification, but ultimately they result in an unnecessary waste of digitizing and computing resources

(iii) Positional accuracy of probe and robot Resolution of the digitizing probe and positional accuracy of the robotic workcell as a whole also impose limitations on

potential accuracy of the digitizing process In Chapter 1, Section 1 3.3, it is mentioned that the probe should be an order of magnitude more accurate than the repeatability of the robot If this criterion is satisfied, only the positional accuracy of the robot need be considered as a limiting factor

(iv) Modelling technique No single surface modelling technique may be regarded as equally applicable to all types of surfaces, and hence a proper choice of modelling technique clearly affects the outcome of surface modelling Any digitizing stepsize determination must therefore take into account the modelling technique used, and how well it is able to represent the surface in question

In the case of a component to be refurbished, one means for determining a proper digitizing stepsize is through analysis of original design data From this analysis, a reduced-order grid of elevations may be extracted to serve as a basis for off-line digitizing path generation. This section will outline such a procedure, as developed by Temple-Raston *et al* [63]

The use of splining to model a component surface allows for significant reduction in demands on both data storage and digitizing It is for this reason that this approach is utilized in the present work In particular, bicubic B-splines (to be described more fully in Chapter 4) are the type of surface interpolation function selected for use on the CIC deburring workcell High-resolution corrective machining toolpath generation may be carried out for a wide variety of surfaces from digitized data using this type of spline

For a given user-specified maximum total surface modelling error, modelling error analysis determines the coarsest permissible digitizing grid This will be the grid which, when

interpolated using bicubic B-splines, yields a maximum modelling error just less than that specified by the user Maximum total surface modelling error is equal to the sum of two types of error· analytic error and geometric error Analytic error is the error due to modelling an arbitrary curve or surface using splines Geometric error is the additional error incurred by use of position-control splining Position-control splining is a surface modelling technique in which the original set of knot points are replaced by a set with different spacing Details of this analysis appear in Temple-Raston *et al* [63] The modelling error analysis algorithm as implemented on the CIC deburring workcell creates a database containing a reduced-resolution grid of estimated digitizing elevations These elevations are linearly interpolated from the original high-resolution NC database and spaced at the newly recommended stepsize

2.4.2 Coordinate Transformations

During the design stage, it is natural for an engineer to select a coordinate system oriented to one or more principal features of a component so as to permit simpler representation of its geometry For purposes of machining or digitizing, a transformation must then be applied to original design coordinates such that the component is properly located on the machine tool worktable in a machine tool-oriented coordinate system In this thesis, the machine tool-oriented system is referred to as *robot coordinates*, and the component-oriented system is referred to as *workpiece coordinates.*

Use of a coordinate transformation is necessary to pre-process a digitizing toolpath generated in workpiece coordinates so that it may be executed in robot coordinates After

digitizing is completed, it is desireable to transform sampled data back from robot coordinates to workpiece coordinates for visualization or analysis Conversely, on-line digitizing path generation may be done entirely in robot coordinates (this will be discussed in more detail in Section 2 4) For both cases, a robot-to-workpiece transformation may afterwards be carried out on the sampled data

In general, robot-to-workpiece and workpiece-to-robot transformations need only consist of a 3D translation and a 3D rotation For this process, six transformation parameters are required three linear translational displacements, and three angular displacements It is usually possible to choose the coordinate systems in such a manner as to set some of these parameters equal to zero This reduces computing time for real-time implementations For example, transformations used on the CIC deburring workcell in the present research involve four nonzero parameters three translations, and one rotation about the vertical axis

Another type of transformation, henceforth referred to in this thesis as a *serpentine path transformation*, is also useful for digitizing path generation This transformation permits time savings by a reduction of required digitizing pathlength In this transformation, every second row of elevations is reversed A typical serpentine digitizing path is shown in Figure 2.4 This transformation results in the shortest possible digitizing path due to the fact that no two successive digitizing points are separated by a distance of more than one grid unit

In the present work, digitizing stepsizes in workpiece-x and workpiece-y directions have been maintained equal It is therefore possible to unambiguously nondimensionalize all three axes of a surface plot with respect to grid size, as in Figure 2 5 In effect, all three axes in Figure 2 5 have been shifted and rescaled (but not rotated) relative to the

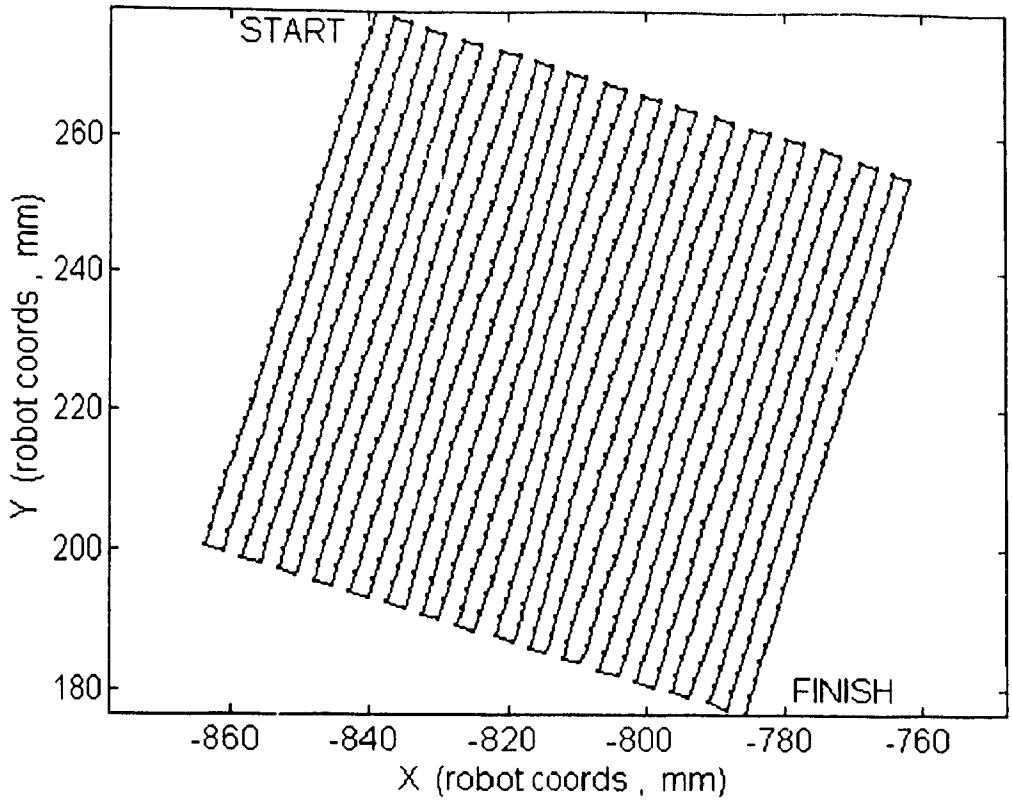


Figure 2.4: Plan View of Serpentine Digitizing Path

workpiece coordinate system The lowest elevation and grid indices have been set equal to zero This surface appears smooth because the digitizing stepsize used (2 71 mm) is much larger than the repeatability of the Yamaha Zeta-1 robot (±0 1 mm)

2.5 Summary

Two approaches to digitizing surfaces have been presented, one based on off-line path generation and the other based on on-line path generation The principal novelty of this work lies in its application to a robotic deburring workcell, however methods presented here are readily transferable to other suitably-equipped machine tools or coordinate measuring

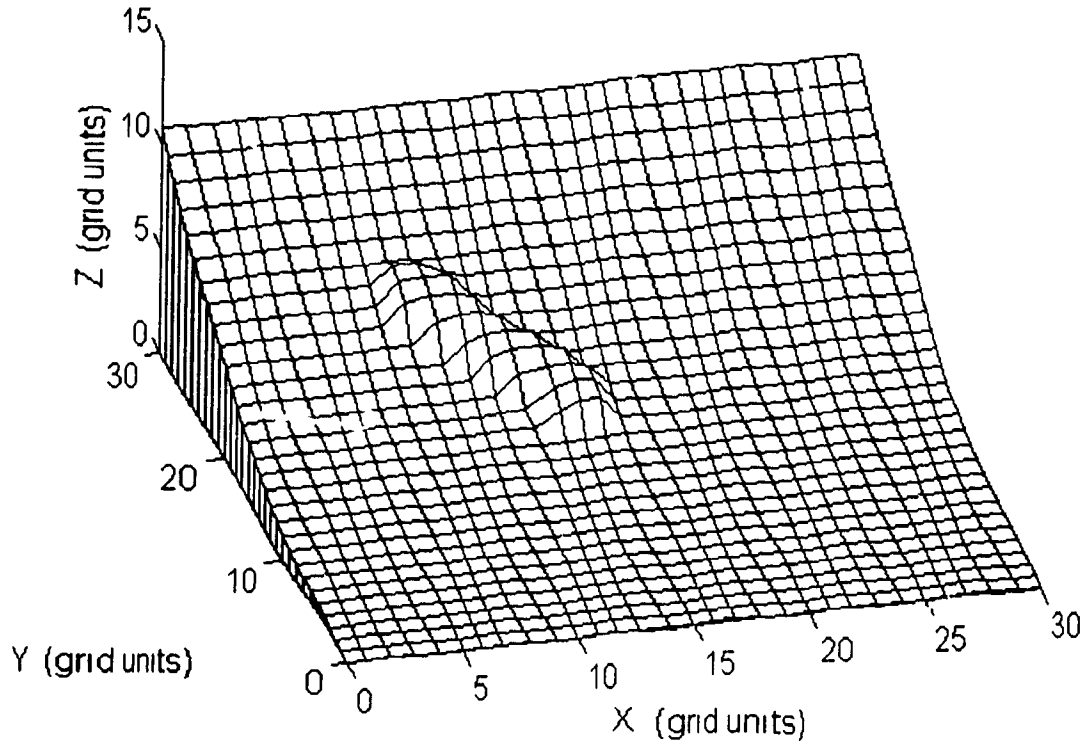


Figure 2.5: Nondimensional Plot of Digitized Surface Elevations

machines This section has summarized the computations required for pre- and post-processing of digitized geometric data, including selection of an appropriate digitizing stepsize (modelling error analysis) and coordinate transformations Results of digitizing using the on-line path generation method discussed in this Chapter are presented in Chapter 5, and serve as a basis for demonstration of the methods for feature recognition and surface reconditioning presented in Chapters 3 and 4

CHAPTER 3 AUTONOMOUS FEATURE RECOGNITION

3.1 Introduction

Autonomous feature recognition is an attempt to mimic on digital computer the capacity in human intelligence for understanding the *concept* of a flaw This recognition process is termed autonomous because it does not reference original design data for purposes of comparison, but rather analyzes surface geometry using sets of predetermined rules The types of surface for which autonomous feature recognition is ideally suited are those with uninterrupted curvilinear continuity, such as planes, concave or convex ellipsoids, paraboloids, hyperboloids, saddles, and the like. The types of surface feature ideally suited to this method include cracks, dents, pitting, scratches and beads of weld However, features which are smaller than the digitizing stepsize used to map a surface are likely to be missed by any recognition scheme.

Industrial application of autonomous feature recognition is possible in automated refurbishment of such components as the control surfaces in fluid power systems or the fan blades in turbine rotor assemblies These are costly items for which it is desired that an efficient means of refurbishment can extend component service life and hence cut operating costs. Since such components may have been distorted, worn or otherwise damaged in the course of normal use, their original design data may no longer be regarded as an accurate geometric model. With original design data thus unsuitable for comparison-based inspection, autonomous feature recognition becomes the necessary alternative

A solution to the problem of autonomous feature recognition is presented in this

chapter, consisting of a two-stage process involving region-growing segmentation followed by fuzzy logic-based classification as reported by Cheng and Roussel [64] The region-growing segmentation algorithm is based on geometric criteria, and is an adaptation of techniques pioneered in image processing by Brice and Fennema [30],Tenenbaum and Barrow [31] and Hoffman and Jain [32] The fuzzy logic-based classifier is an adaptation of a self-tuning multi-input multi-output fuzzy controller scheme developed by Koivo and Viljamaa [40].

The fuzzy controller of Koivo and Viljamaa first constructs input membership functions from the error vector representing differences between desired and actual system output These input membership functions are manipulated by a fuzzy rule base to produce output membership functions, from which defuzzification yields a crisp controller output signal. Output membership functions are defined as fuzzy singletons, whose defuzzification is carried out using the centroid method (the reader is referred to Lee [39] for more details). This procedure is adaptable to classification of surface features using appropriate definitions for membership, rule base structure and defuzzification, as discussed in detail in Section 3 3.

3.2 Regional Segmentation

3.2.1 Region-Growing Algorithm

Conventional region-growing segmentation techniques deal with pixels representing illumination intensity levels or degrees of colouration. In the present work, a computed measure of local surface irregularity has been defined as the criterion to determine regional propagation for use with grids of digitized surface elevations. This growth criterion is

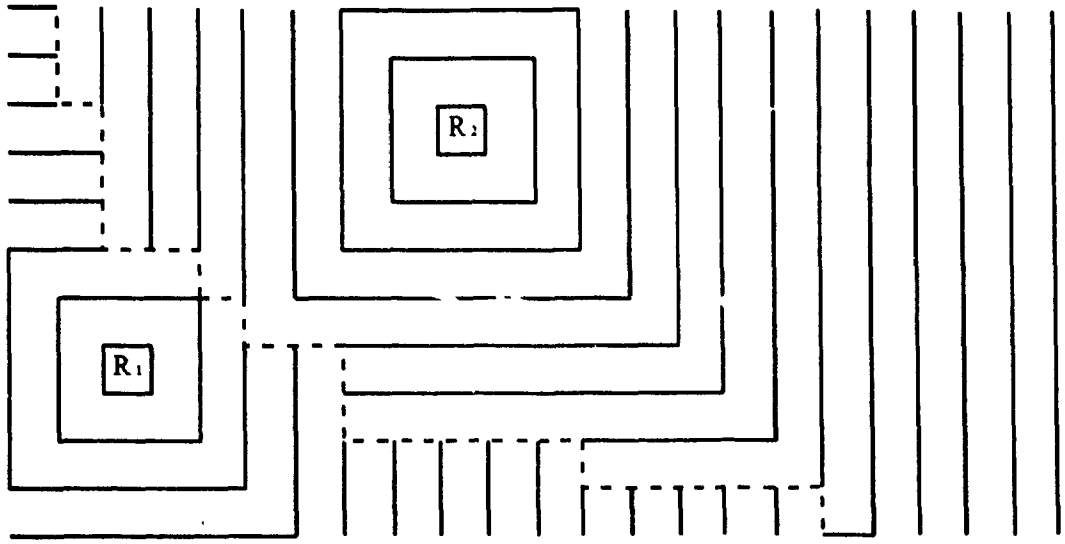


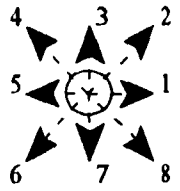
Figure 3.1: Stages of Growth in Two Neighbouring Regions

discussed in detail in Section 3.2.2. The algorithm for regional propagation is first described in this section without explicit consideration of the growth criterion.

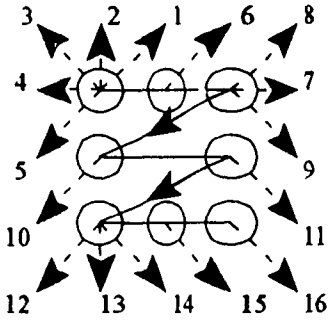
Regions are grown one at a time. Figure 3.1 shows successive stages of growth in two neighbouring regions (one had to be grown first, then the other). The region growing process terminates only when all points on a grid have been assigned regional membership. To start a new region, a seed point must be chosen from the digitized database representing the component surface. The criterion defined for use in seed point selection is described along with the regional growth criterion in Section 3.2.2. Only points with regional membership not yet assigned are considered for seed point selection.

Following this selection, eight surrounding grid points immediately adjacent to the seed are checked for possible inclusion in the currently active region. If any of these points do not hitherto have regional membership *and* are found to satisfy the criterion for regional

Iteration # 1:
(8 points added)



Iteration # 2:
(16 points added)



Iteration # 3:
(24 points added)

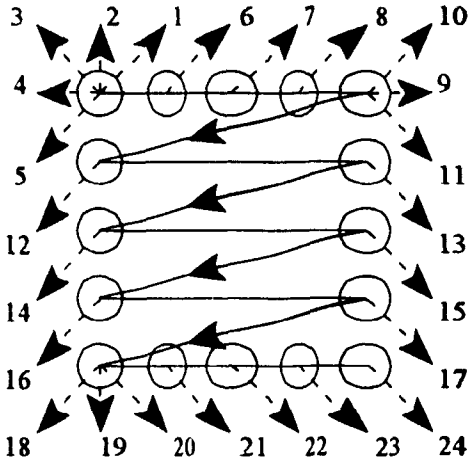


Figure 3.2: The first three rounds of regional growth for the case of complete admissibility of surrounding points. Active points from which growth occurs are circled, and numbering indicates the order in which points are added.

growth, they are assigned membership in the new region Figure 3 2 shows the first three rounds of growth for the case when all surrounding points are found to be admissible In effect, this region has swelled in size by one grid unit in all directions during each round (there is an analogy here with the growth rings of trees!)

During a given round of growth, points which were assigned membership to the currently active region during all previous rounds are systematically scanned to determine if any as yet unclassified points are adjacent to them As each adjacent unclassified point is encountered, it is checked for possible inclusion If the point is found to satisfy the gro :th criterion, its grid indices are written to a temporary file to be stored along with those of any other newly detected points until the current growth round is completed Only after completion of each round is the map of regional memberships updated New rounds of growth are only initiated if the previous round resulted in addition of at least one point

3.2.2 Geometric Criteria for Regional Growth

Referring to Figure 3 3, a vertical angle θ is defined between line segments connecting adjacent points in the grid of digitized elevations Four values of this angle are computed for each interior grid point in the x-direction, y-direction, and principal diagonals labelled as α and β in Figure 3 4 Stepsizes Δx and Δy are taken as constant over the rectangular grid, but it is not required that $\Delta x = \Delta y$

A single half-range bilateral tolerance ϵ_t is set as the maximum allowable variation in θ from one point to the next This tolerance applies to all eight directions along which the four values of θ are defined, that is, $\pm x$, $\pm y$, $\pm\alpha$, $\pm\beta$ If the change in θ along any of these

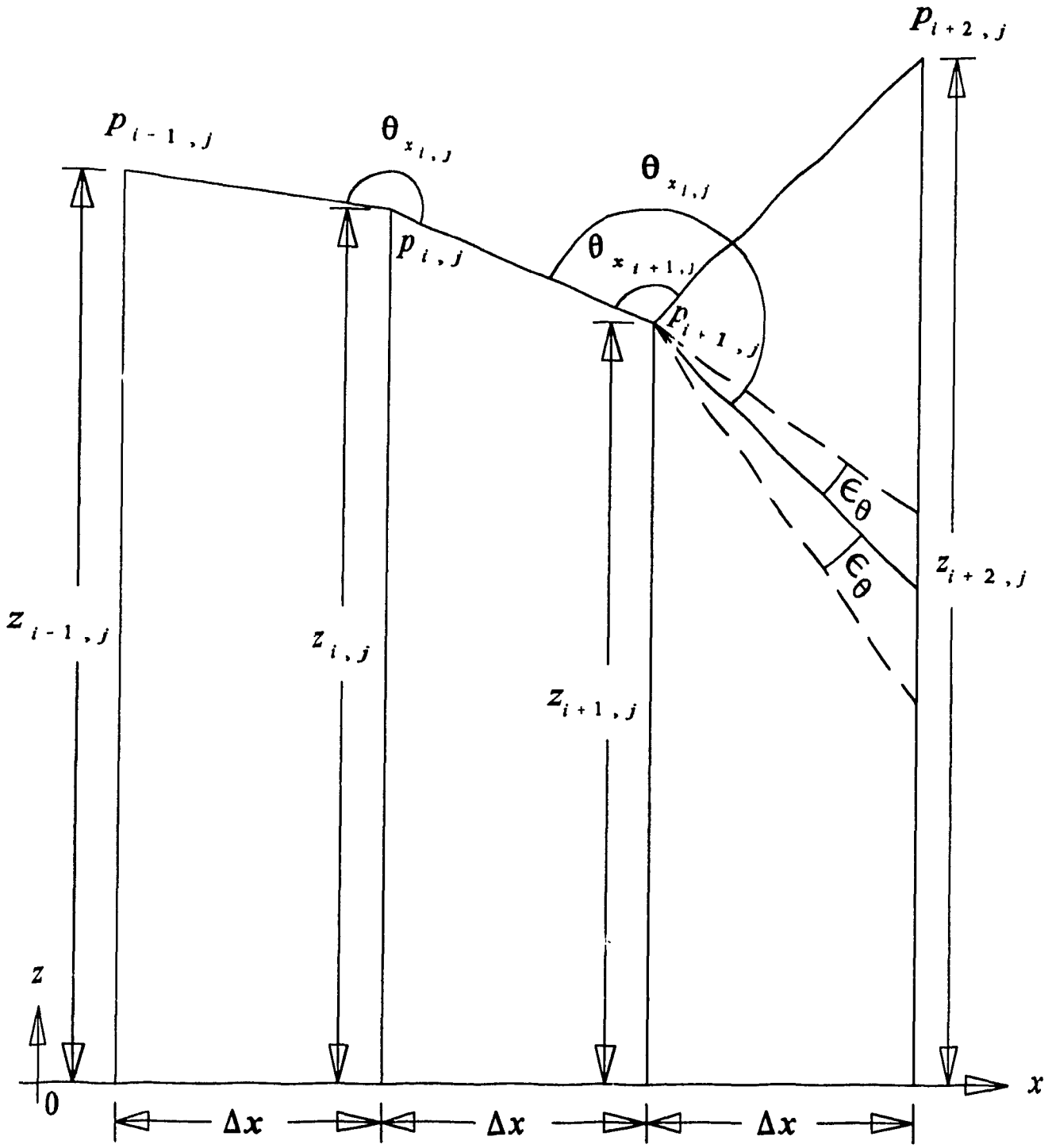


Figure 3.3: Vertical Angle Criterion for Regional Growth.

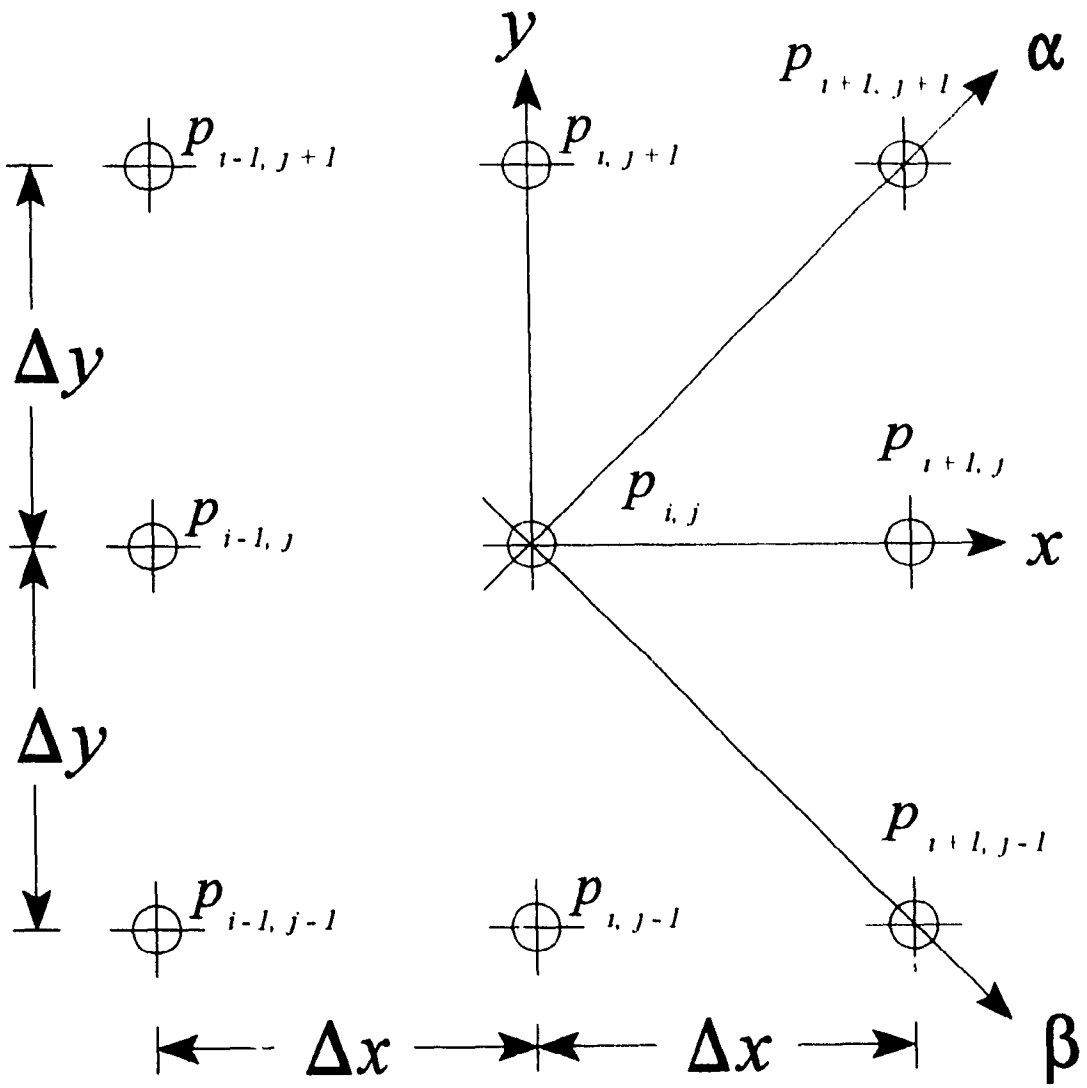


Figure 3.4: Plan view of digitizing grid, with diagonal α and β directions shown, and indicial notation for identification of elevations

eight directions exceeds ϵ_θ, regional growth will be arrested in that direction at that point Figure 3 3 shows an example in which point $p_{i,j}$ is not to be included in the same region as $p_{i,j}$, since

$$| \theta_{x_{i+1,j}} - \theta_{x_{i,j}} | > \epsilon_\theta \qquad (3\ 1)$$

The criterion for seed point selection will now be defined The seed point is chosen (out of those points which do not yet have regional membership assigned) as the point with smallest *roughness index* I_r, defined at grid location (i, j) by

$$(I_r)_{i,j} = \max \{ (r_x)_{i,j}, (r_y)_{i,j}, (r_\alpha)_{i,j}, (r_\beta)_{i,j} \} \qquad (3\ 2)$$

where

$$(r_x)_{i,j} = | \pi - \theta_{x_{i,j}} | \qquad (3.3)$$

with r_y, r_α, r_β defined similarly to r_x In the above equations, subscripts x, y, α, β denote the principal directions as shown in Figure 3 4, and θ is in radians

Seeding of each new region therefore occurs at the 'flattest' as yet unclassified grid point In general, those regions which contain the flattest points are also most likely to be in the most acceptable portions of the surface The most acceptable (and usually largest) regions thereby tend to be grown first. This promotion of clustering in the regional numbering scheme serves as a visual aid during tuning of the segmentation algorithm, and does not otherwise affect the final outcome of classification

Tuning the segmentation algorithm consists of setting the half-range bilateral tolerance

ϵ_θ If ϵ_θ is set too small, segmentation becomes very sensitive, that is, an excessively large number of small regions will be identified on a surface If ϵ_θ is set too high, segmentation will become very insensitive, and only a few large regions will be identified Computer-aided visualization of segmentation results is essential for determining proper tuning Experimental results based on use of visualization-guided tuning are presented in Chapter 5

3.2.3 Regional Membership Assignation

Output of the segmentation algorithm consists of a map written as an integer-valued array which assigns regional membership to corresponding digitized grid elevations This is shown for a typical case in Figure 3 5 Note all regions are contiguous, that is, either simply or multiply connected Since it was chosen first, the seed point for Region 1 is 'flattest' of all seed points in accordance with the selection criterion described in Section 3 2 2 Exactly which point this is cannot be deduced solely from inspection of Figure 3 5 Conversely, the seed point of Region 27 (the highest-numbered region), which also happens to be its only point, is the 'least flat' (roughest) of all seed points encountered in this example

Due to the nature of the regional growth criterion (described in Section 3 2 2), edge points on the digitized grid cannot be considered by segmentation For a large digitized database collected using multiple adjacent rectangular grids, the number of edge points can be minimized by overlapping grids Edge points are labelled in Figure 3 5 as belonging to Region "0", and are not rejected during subsequent classification (described in Section 3 3) Only regions 1 through 27 have therefore been properly identified by segmentation, with region 1 in this case much more extensive than the rest

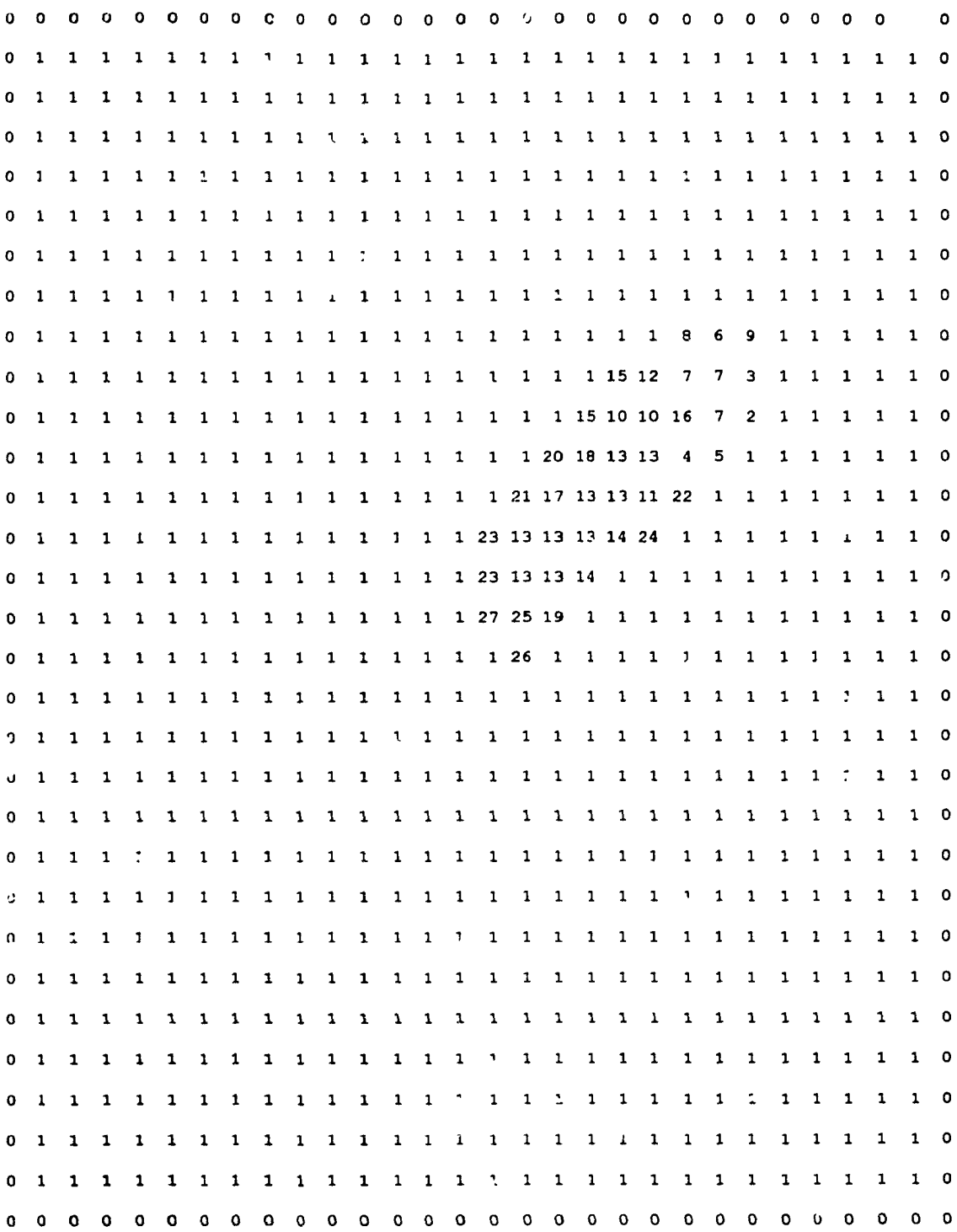


Figure 3.5: Map of Regional Membership Assignation

The extraneous feature on this surface corresponds to the approximately linear feature of the example in Section 5 3 3, Figure 5 28 and is represented by regions 2 through 27 This feature is composed of a large number of smaller regions since it is both irregular and highly curved, and hence regional propagation is locally inhibited A map of regional membership as given by Figure 3 5 serves as input to the regional classification algorithm, to be described in Section 3 3

3.3 Fuzzy Logic-Based Regional Classification

3.3.1 Input Variables and Input Membership Functions: Fuzzification

There is no single obvious geometric property of a region which can be used to reliably indicate acceptability However, it is required that a decision be rendered as to whether distinct regions on a surface are acceptable or not The present work has utilized two criteria in performing this classification, each of which defines membership in fuzzy sets Fuzzy membership quantifies the degree to which each region matches specific predefined criteria, and thus forms the starting point for a fuzzy logic-based decision making algorithm

One criterion used in the present work stems from the observation that rejected regions are likely to be discontinuous or *rough* A regionally-averaged measure of the maximum local deviation from flatness is thus devised for classification

$$(x_r)_R = \frac{1}{(N_P)_R} \sum_R ((I_r)_{i,j}) \qquad (3\,4)$$

where $(N_P)_R$ is the number of points in region R Recalling the definition of roughness index $(I_r)_{i,j}$ in Eqn (3 2), it is apparent that the angles $\theta_{i,}$, $\theta_{i,}$, θ_{α}, θ_{β} have conveniently already

been computed during segmentation and are reused here. In general, regions with large $(x_r)_R$ are most likely to be those which form part of an extraneous feature, and hence are also those which most likely should be rejected

Another useful criterion is defined based on the extent of regional areas On any surface which is a realistic candidate for successful refurbishment, it is reasonable to expect that the majority of data points belong to undamaged, that is, acceptable regions Rejected regions (those consisting of cracks, dents, pitting, scratches, beads of weld, and the like) must therefore occupy but a small fraction of the total component surface This leads to the heuristic assumption that smaller regions are most likely destined to be rejected.

For classification purposes, an adequate measure of regional populations may be defined as the number of points contained in the region Rectification of the variable representing regional area may be carried out to give it the same numerical trend as $(x_r)_R$ between acceptance and rejection The following definition has therefore been adopted in the present work·

$$(x_a)_R = -(N_P)_R + w_a \qquad (3.5)$$

where w_a is a constant, to be defined below

Input membership functions associated with the two classifier input variables $(x_a)_R$ and $(x_r)_R$ are computed linearly as follows·

$$\left.\begin{aligned} (\mu_{L_a})_R &= -\frac{1}{w_a}(x_a)_R + 1 \\ (\mu_{H_a})_R &= \frac{1}{w_a}(x_a)_R \end{aligned}\right\} \qquad (3.6)$$

and

$$\left.\begin{aligned} (\mu_{L_r})_R &= -\frac{1}{w_r}(x_r)_R + 1 \\ (\mu_{H_r})_R &= \frac{1}{w_r}(x_r)_R \end{aligned}\right\} \qquad (3.7)$$

where w_a and w_r are the *widths* of the input membership functions, and subscripts *L* and *H* represent Low and High respectively. Widths are set equal to the maximum value attained by each type of input variable for a given surface. w_a is the area of the most extensive region (as measured by number of points), and w_r is the largest $(x_r)_R$. In the course of extensive testing with experimentally-sampled surfaces (as documented in Chapter 5), the above criteria have been found effective, but other forms of input membership function are certainly also possible.

3.3.2 Fuzzy Rule Base Structure

A rule base structure consisting of four rules (Eqn. 3.8) has been adopted for the fuzzy classifier, as shown in Table 3.1. In this table, letters '*L*' and '*H*' represent membership functions corresponding to the input variable in whose column they occur. To illustrate this, consider Rule # 3, which contains a mix of High and Low membership input variable $(x_a)_R$ is mapped to its corresponding High membership function $(\mu_{H_a})_R$, and input

variable $(x_r)_R$ is mapped to its corresponding Low input membership function $(\mu_{L_r})_R$ The number of rules N in this rule base is computed from

$$N = (N_{imf})^{N_{in} \cdot N_{out}} = 2^{2 \cdot 1} = 4 \tag{3.8}$$

where N_{imf} is the number of input membership functions per input, N_{in} is the number of inputs, and N_{out} is the number of outputs

Table 3.1: Fuzzy Rule Base

Rule	$(x_a)_R$	$(x_r)_R$	*Output*
1.	*L*	*L*	$(\mu_{out\ 1})_R$
2	*L*	*H*	$(\mu_{out\ 2})_R$
3	*H*	*L*	$(\mu_{out\ 3})_R$
4	*H*	*H*	$(\mu_{out\ 4})_R$

3.3.3 Output Variables and Output Membership Functions: Defuzzification

Table 3.1 defines a set of logical operations which are based on the definition of intersection between fuzzy sets. Once again considering Rule # 3 as an example from Table 3.1, we have

$$(\mu_{out\ 3})_R = \min\{(\mu_{H_a}), (\mu_{L_r})\} \tag{3.9}$$

The above equation defines a fuzzy singleton-type output membership function. In the present method, there are four output membership functions one corresponding to each fuzzy rule

Singletons are real-valued scalars defined in t. · range [0, 1]

Defuzzification is accomplished by the formula

$$\left.\begin{aligned}(x_{out})_R &= \frac{1}{N}\sum_{i=1}^{N}\left[p_i(\mu_{out_i})_R\right] \\ &= \frac{1}{N}\,\boldsymbol{p}^T(\boldsymbol{\mu}_{out})_R\end{aligned}\right\} \qquad (3\ 10)$$

where

$$\left.\begin{aligned}\boldsymbol{p} &= \begin{bmatrix}p_1\\p_2\\p_3\\p_4\end{bmatrix} = \begin{bmatrix}0\ 1\\0.35\\0.65\\0\ 9\end{bmatrix} \\ (\boldsymbol{\mu}_{out})_R &= \begin{bmatrix}(\mu_{out_1})_R\\(\mu_{out_2})_R\\(\mu_{out_3})_R\\(\mu_{out_4})_R\end{bmatrix}\end{aligned}\right\} \qquad (3\ 11)$$

The vector $\boldsymbol{p}$ gives the *placement* of corresponding entries in the output fuzzy singleton vector The centroid of output membership functions is thus computed for each region by Eqn (3 10) Fuzzy classifier output represents an amalgamated index of acceptability a positive-definite real scalar in the range [0, 1] corresponding to each region on the surface A threshold logic unit is used to sort defuzzified output into Accept or Reject categories, based on selection of the value of x_{out} Tuning of the classifier is therefore carried out by adjustment of the threshold parameter x_{out} A block diagram of the fuzzy classifier is given in Figure 3.6

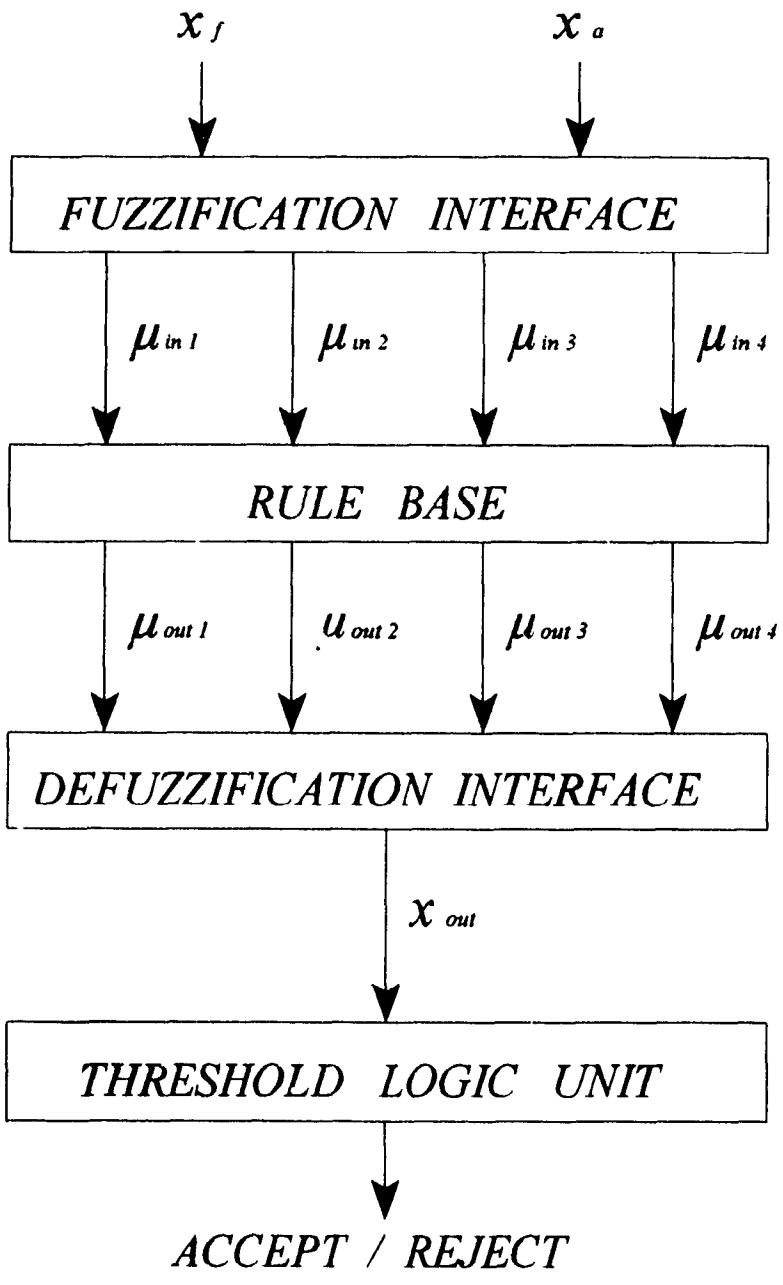


Figure 3.6: Block Diagram of Fuzzy Classifier

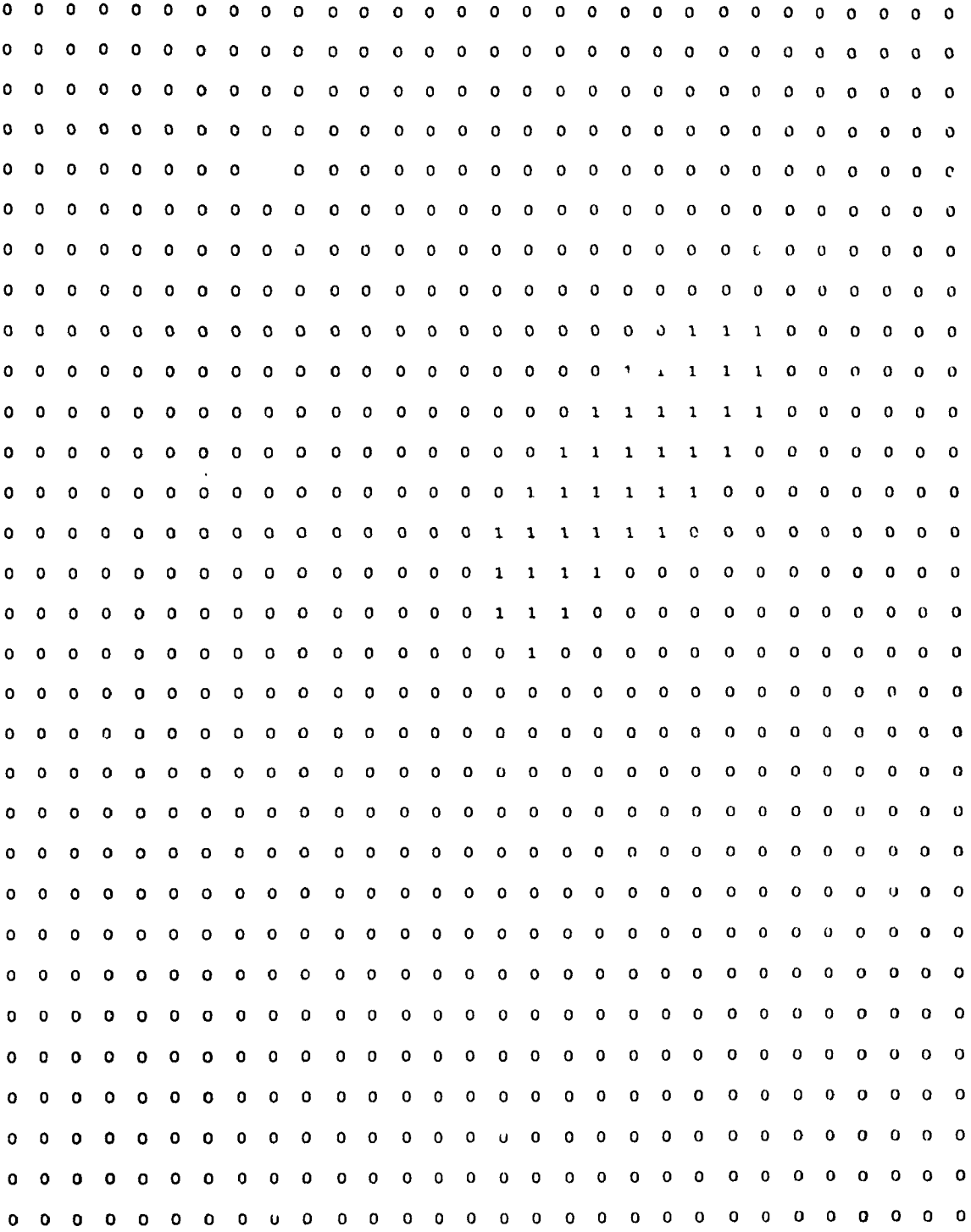


Figure 3.7: Feature Map (corresponding to Figure 3 5)

3.3.4 The Feature Map

Output from the fuzzy classifier consists of a map written as an integer-valued array which assigns Accept or Reject classification to corresponding digitized grid elevations This is shown for a typical case in Figure 3 7, which represents the classification of regions shown in Figure 3 5 Rejected grid points are identified with a "1", and accepted points with a "0" A 3D surface plot highlighting rejected points superimposed on the digitized component surface is useful as an aid in assessing the classification result Plots of this type are shown for experimentally-sampled surfaces in Chapter 5

3.4 Summary

A two-stage procedure for autonomous feature recognition has been presented, along with the geometric criteria required for use with both of these stages Region growing is based on use of a bilateral half-range tolerance applied to the variation of vertical angles subtended by line segments connecting adjacent grid points The fuzzy classifier makes use of aggregate properties computed for each region These properties represent quantified measures of surface roughness and regional area

Segmentation requires the digitized surface data as input, and generates a map of regional memberships as output This map of regional memberships is coupled with digitized surface data to compute values for the two variables corresponding to each region which serve as inputs to the fuzzy classifier A single output (a "0" or a "1" representing Accept or Reject categories, respectively) corresponding to each segmented region is then generated by the classifier. These classifications are given in the form of a feature map as discussed in

Section 3 3 4

Use of a segmentation algorithm prior to fuzzy logic-based classification is justified as a basis for constructing a reliable autonomous feature recognition process This two-stage approach is advantageous because the necessity of attempting to classify the entire surface point-by-point is thus avoided Classifier reliability is enhanced by allowing consideration of only a relatively small number of appropriately selected regions

CHAPTER 4 AUTOMATED SURFACE RECONDITIONING

4.1 Introduction

Automated surface reconditioning consists of the following four major steps

(i) determination of damaged regions on a component surface,

(ii) determination of a desired reconditioned geometry for the damaged component,

(iii) generation of a corrective machining toolpath,

(iv) execution of the corrective machining toolpath (physical reconditioning)

In Chapter 3, the procedure for step (i) has been discussed This chapter describes extensively the methodology for steps (ii) and (iii) Step (iv) is important for verification of the methodologies developed, and to carry out repair of damaged components However, this issue is beyond the scope of the present thesis, due to lack of availability of suitable workpieces Hence, to test the methodologies developed for steps (i), (ii) and (iii), the following procedure is adopted The presence of beads of weld on accurately machined component surfaces is simulated by introducing extraneous materials (e g , quick-drying epoxy glue) on them Digitizing of component surfaces containing these features is carried out using the Yamaha robot Data so collected is processed using the fuzzy logic-based feature recognition technique developed and described in Chapter 3 After identification of rejected features, the surface geometry is interpolated based on trends in nonrejected regions using a least-squares optimal surface-fitting procedure This surface fit establishes a desired geometric model for the reconditioned component Note that original design data cannot be referenced for purposes of comparison in step (iii) in the case of surfaces which have

undergone significant changes in their geometry due to warping As with Chapter 3, this chapter exclusively considers cases for which original design data is not usable

The present approach to step (ii) requires as input a feature map (such as in Figure 3 7) generated by autonomous feature recognition, and a grid of digitized elevations for the damaged component Rejected elevations (those that form part of rejected features) are then replaced such that geometric trends of local nonrejected points are interpolated In effect, the rejected features are blended out of existence in a new database termed the Maximum Likelihood Eventual Surface (MLES)

Corrective machining toolpath generation (step (iii) above) as carried out in the present work involves use of bicubic B-splines to interpolate the MLES determined in step (ii) Required input for step (iii) consists of both the digitized elevations representing damaged geometry, and the MLES For the case of surfaces containing only features that stand proud of the surrounding material, all points on the MLES grid will be at elevations greater than or (approximately) equal to those of the damaged grid The difference in volume subtended beneath these two surfaces is then equal to the amount of material to be removed by corrective machining If the damaged surface contains rejected features whose elevations are below the MLES, machining alone will not suffice these features must be built up with weld and then machined Applications of the present work are therefore restricted to surfaces which have been prepared in advance by filling in any depressions or pitting with weld which may then be machined down to a desired reconditioned geometry Step (iii) generates the machining toolpath based on knowledge of the desired geometry, and other machining requirements, including feed rate, depth of cut, cutter geometry, and robot kinematics This

toolpath consists of (x, y, z) cutter coordinates and cutter axis orientation (A, B) in robot coordinates

This chapter first presents a method for computing MLES databases, along with its analytical background This method, based on surface fitting by multivariate optimization, is designed for use with the feature recognition algorithm presented in Chapter 3 The second topic of this chapter is the use of bicubic B-splines for computing elevations and normals which allow tracing of the MLES at resolutions higher than that used for digitizing These elevations and normals are the basis to generate the toolpath for corrective machining.

4.2 Determination of a Desired Reconditioned Geometry for Damaged Components

4.2.1 The Use of Surface Fitting

In computing MLES grid points, use of surface fitting is considered preferable to curve fitting for the following reason curve fits are only sensitive to geometric trends along a particular trace of a surface, whereas surface fits attempt to match local geometry in all directions. The present work has therefore employed surface fitting of local nonrejected elevations It must next be recognized that exact interpolative fits cannot be obtained, since there are sampling errors inherent in the digitized database representing actual (damaged) component geometry In other words, since nonrejected elevations do not describe a perfectly smooth surface, there will be limits as to how closely local geometry can be modelled by surface fitting Any computed interpolating surface must therefore be expected to pass *near* rather than *through* local nonrejected points

The next issue concerns what type of surface fit should be used In the most general

case, components will have slight arbitrary warping deformations which cause them to differ from original design geometry For the present method, it has therefore been conservatively assumed that precise geometric information cannot be extracted from the original design A choice of interpolative surface function must be made, however, and a basis for that choice must be formulated

One heuristic rationale for selection of appropriate forms of interpolative surface function may be derived from identification of the essential geometric characteristics of turbine rotor blades They are *smooth* (curvilinear continuity), but with continuous *variations* in curvature A family of functions with such properties is given by the following bivariate polynomial recommended for surface fitting by Lancaster and Salkauskas [5]

$$z = F_N(x, y) = \sum_{p=0}^{N} \left[\sum_{q=0}^{N} \left(a_{pq} x^p y^q \right) \right] \qquad (4.1)$$

This represents the most general bivariate polynomial function possible up to and including degree N. Choosing $N \geq 1$ gives rise to surfaces (bilinear, biquadratic, bicubic, biquartic, biquintic, etc) which exhibit continuously varying curvature, and hence these cases are of interest for fitting to data representing curved surfaces

With a form of surface function chosen, it remains to decide how the coefficients a_{pq} in Eqn. (4.1) should be determined Since, as mentioned previously, an interpolating surface function may only be expected to pass *near* rather than *through* local digitized nonrejected points, use of an optimization scheme to achieve a best fit naturally suggests itself In performing a surface fit to replace a particular rejected point, one need only consider nearby (local) nonrejected points rather than all points on the surface It is neither

necessary nor desireable to attempt global surface modelling

Two further issues must be addressed at this stage The first concerns how local nonrejected reference points should be chosen for use in an optimization scheme This issue is dealt with in Section 4 2 2 The second issue is concerned with how the optimization itself should proceed To answer this, a sensitivity analysis for polynomial functions of the type given by Eqn (4 1) is carried out in Section 4 2 3 The optimization algorithm is presented in Section 4 2 4

4.2.2 Reference Point Selection Strategy

Suitable points chosen as a basis for surface fitting must satisfy two criteria

(i) they must be nonrejected (part of the MLES),

(ii) they must be within some appropriately-chosen local neighbourhood of any rejected point for which a surface fit is currently being computed

Only a local surface fit is attempted, since accurate globally-valid surface modelling functions are more difficult to establish It follows that the best possible grid point replacement is accomplished if a unique local surface fit is computed for each rejected elevation For this purpose, a correspondingly unique local set of nonrejected points whose distribution is centred on the currently rejected point is thus desired for surface fitting

A simple strategy for determining membership in a set of local reference points is to include all those up to a given radius from the rejected point whose desired reconditioned elevation is currently being sought However, in the case of rejected points located deep inside extensive features, this may result in the inclusion of too few reference points With too

sparse a set of reference points, surface fitting will tend to become unacceptably inaccurate Another approach is to specify a minimum number of points that must be included A reference point neighborhood (in the shape of a disk) may then be uniquely sized for each rejected point This would guarantee that the reference point set is never too sparse However the distribution of nonrejected points will still be governed by the distribution of local rejected features

Specifying a minimum number of reference points to be used in sizing all reference point disks is the method that has been employed in the present work For example, if thirty reference points are required, the reference disk radius is incrementally increased until the nearest thirty points are included A plot showing variation of the maximum possible number of points occurring within a reference disk versus radius is shown in Figure 4 1 This irregularly-stepped curve is a discrete analogue to the relationship between area and radius of a circle ($A = \pi R^2$) Rejected points which belong to features consisting of multiple adjacent rejected points will have fewer reference points for a given radius than Figure 4 1 indicates, since rejected points cannot themselves form part of any reference point set

The final issue here concerns how to specify an adequate number of points Determining this depends on a combination of factors Local nonrejected surface geometry will affect the ability of any given type of curve fit to conform to the desired surface The size and shape of rejected features places a restriction on the number of local elevations available for use in surface fitting The smoothness of surface fits also varies with reference disk radius, as will be shown for a series of experimental examples in Chapter 5

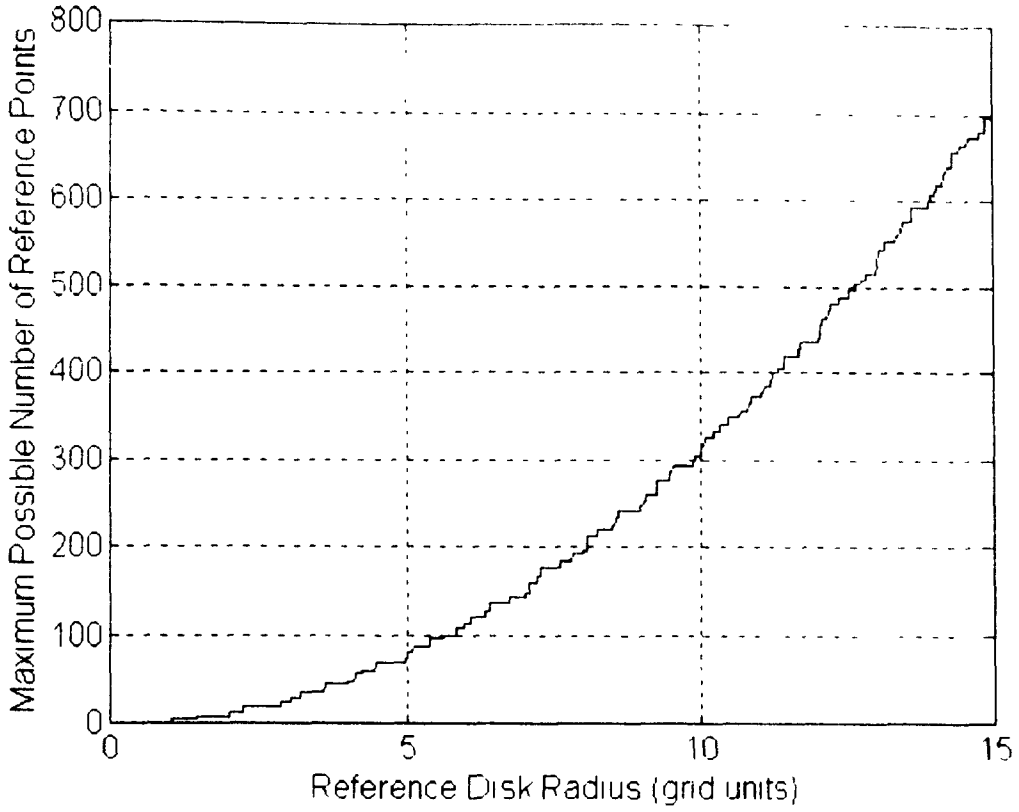


Figure 4.1: The maximum possible number of points contained in a reference disk versus its radius For this curve, the central (rejected) point is not counted, and grid spacing is taken as equal in both x and y directions

4.2.3 Sensitivity Analysis

Sensitivity analysis reveals how Eqn (4 1) may best be used in a surface fitting optimization scheme. In particular, use of the same evaluation point (x_p, y_p) for all grid elevation replacement computations can be justified, along with selection of that evaluation point To examine sensitivity of Eqn (4 1) to variations in the coefficients a_{pq}, a first order Taylor series expansion may be used

$$z = z_0 + \left.\frac{dz}{da_{p,q}}\right|_{(x_0,\,y_0)} \Delta a_{p,q} \qquad (4.2)$$

The partial derivative term appearing in Eqn (4.2) is, to a first order approximation, the sensitivity of z to a unit change in $a_{p,q}$. Computed for Eqn (4.1), this derivative is as follows

$$\frac{dz}{da_{p,q}} = x^p y^q \qquad (4.3)$$

If this derivative is plotted against indices p and q for a particular evaluation point (x_p, y_p), one obtains sensitivity surfaces as shown in Figures 4.2 and 4.3. Note that these surfaces are only defined for positive integral values of the indices. Also, different sensitivity surfaces will result from different choices of evaluation point.

The surface shown in Figure 4.2 is for $x_p = y_p = 0.5$, and shows maximum sensitivity of F_V (recall Eqn (4.1)) to variations in $a_{0,0}$ (the constant term), and mimimum sensitivity to coefficients with large p and q (higher degree terms). Figure 4.3 shows the sensitivity surface for $x_p = y_p = 1.5$. For this evaluation point F_V is most sensitive to variations in higher degree terms and least sensitive to the constant term - a reversal of the situation in Figure 4.2.

Using the same evaluation point for Eqn (4.1) during all grid point replacement computations on a given surface is desireable because in this way the resulting sensitivity surface of F_V becomes invariant. It has been found during testing of the present surface fitting optimization algorithm that use of a single properly-chosen evaluation point gives vastly improved convergence over the alternative of using x and y values from the digitized grid exactly as is. This is because the family of surfaces given by Eqn (4.1) are not highly curved at distances far from the origin, that is, where $R = x^2 + y^2 \gg 1$, regardless of the

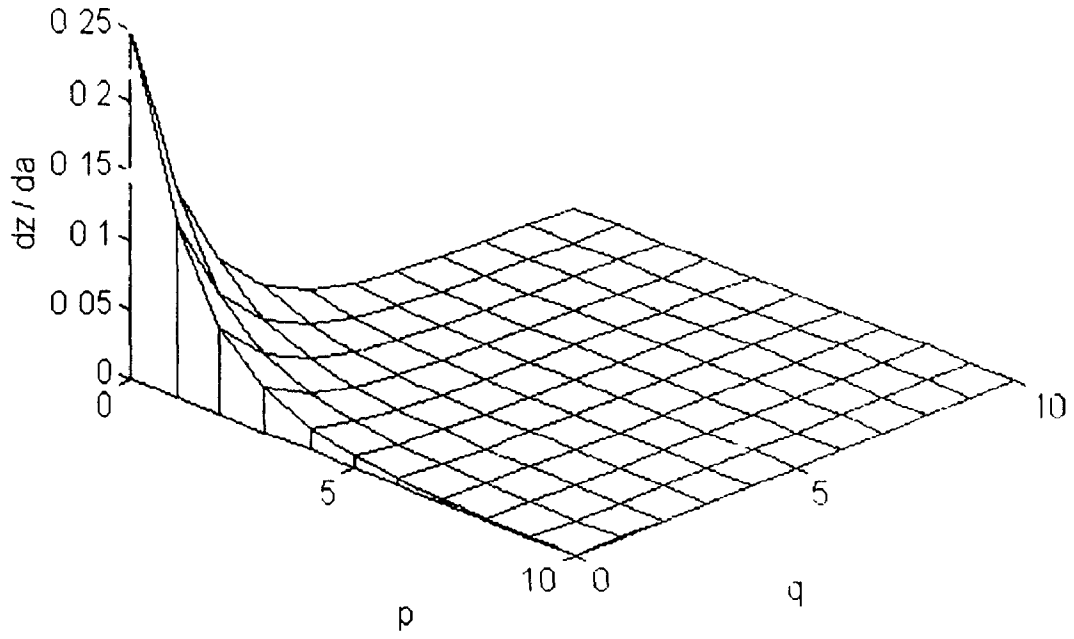


Figure 4.2: Sensitivity surface for Eqn (4 1), evaluated at $x_p = y_p = 0.5$

set of coefficients a_{pq} used Closer to the origin, these surfaces may possess either large or small curvature, depending on the choice of coefficients To maintain a constant evaluation point, x and y coordinates of all elevations in each reference point set must be rescaled and translated

The question now arises as to what evaluation point should be chosen Situating the evaluation point such that $x_p = y_p$ yields an I_{xy} (as given by Eqn (4 1)) which is equally sensitive to variations in both x and y, as demonstrated by Figures 4 2 and 4 3 With no consistent *a priori* expectation of what component geometry is to be encountered, choice of an evaluation point such that $x_p = y_p$ appears reasonable It therefore remains to select for

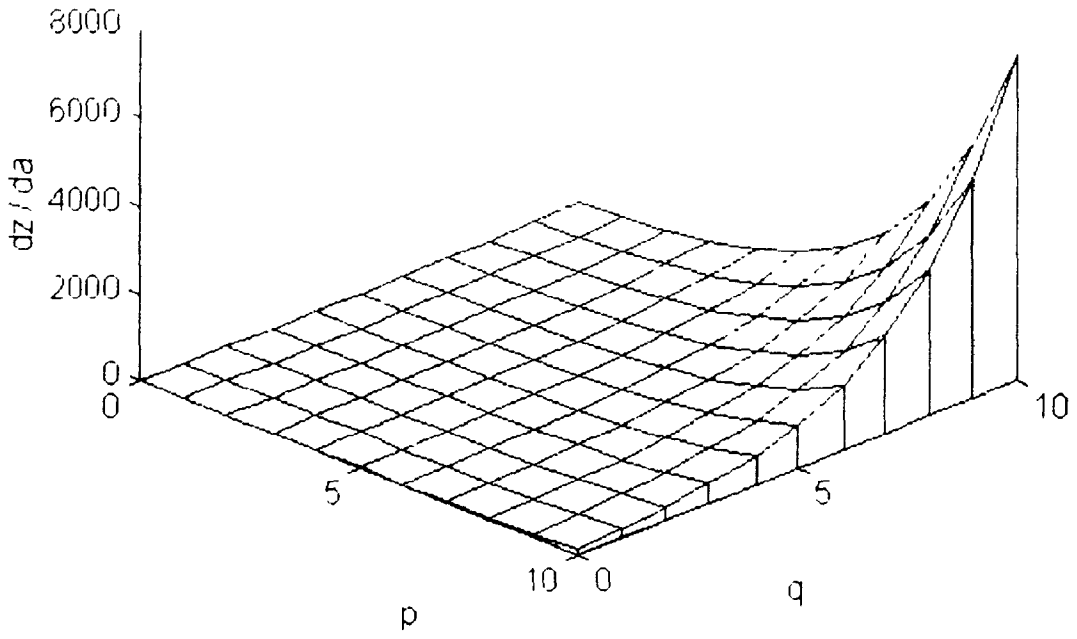


Figure 4.3: Sensitivity surface for Eqn (4.1), evaluated at $x_p = y_p = 1.5$

the magnitude of x_p and y_p

Bivariate surface functions of the form given by Eqn (4.1) exhibit a trend of decreasing curvature as distance from the origin becomes large. This means that when attempting to optimize a surface fit in a highly curved region, the evaluation point is best chosen close to the origin. Setting the evaluation point close to the origin does not adversely affect modelling of relatively flat surfaces (in this case, higher degree terms simply vanish). If the evaluation point is chosen exactly at the origin, however, then Eqn (4.1) loses all sensitivity to coefficients other than a_{00}, which is also undesireable. In the present work, the choice of $x_p = y_p = 0.5$ has been found to be an acceptable compromise for the surfaces

studied in Chapter 5. With a suitable evaluation point thus selected, the surface fitting algorithm may now be discussed in detail.

4.2.4 Multivariate Optimization Algorithm for Surface Fitting

Input to the surface-fitting algorithm consists of two sets of data: an array of digitized component elevations, and a feature map. Output of this algorithm is the MDES data, defined in Section 4.1. The feature map (recall Figure 3.7) is a binary-valued array which assigns Accept / Reject classification to each digitized grid point (0 = Accept, 1 = Reject). Referring to the flowchart in Figure 4.4, the feature map is scanned point by point. Each nonrejected elevation encountered is recorded as output data. For each rejected elevation encountered, surface fitting is carried out, whereby a replacement elevation is computed and recorded as output data. The surface fitting algorithm proper is described below and illustrated by flowcharts in Figures 4.4 and 4.5.

Referring to Figure 4.5, the optimization begins with an initial blind guess for all surface modelling polynomial coefficients $a_{p,q}$ in Eqn. (4.1). An error index, defined in Eqn. (4.4), is computed to measure the closeness of fit between this initial surface model and the reference point data. Each polynomial coefficient is then varied in turn both positively and negatively using a specified stepsize, and resulting error indices are computed for each variation. The set of original coefficients is then updated to include the variation which gave rise to the lowest error index. This new set of coefficients is then varied as before, with corresponding error indices computed, in order to result in a new lowest value of the error index. If none of the variations for a given iteration results in an error index lower than the

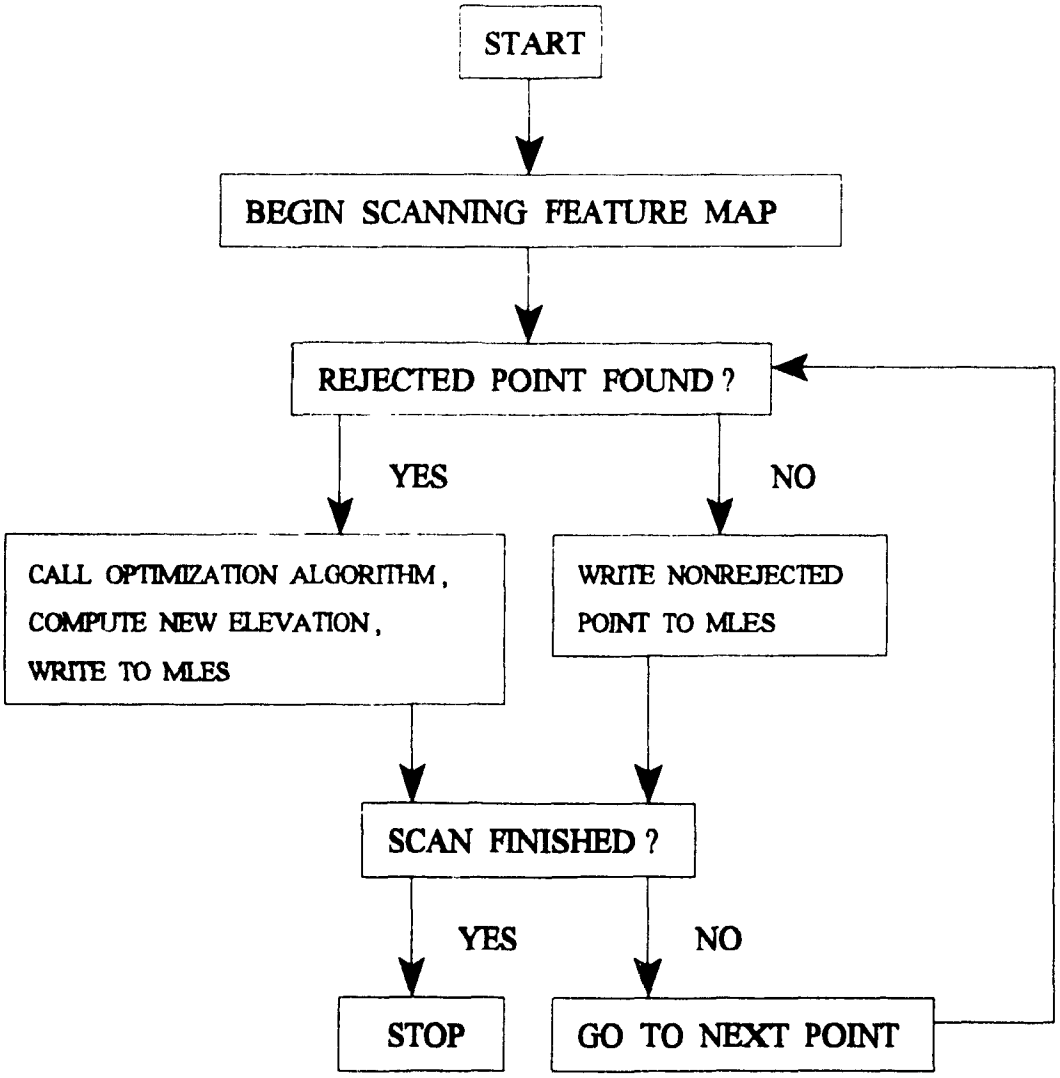


Figure 4.4: Flowchart illustrating action of the shell within which the surface fitting optimization algorithm is called

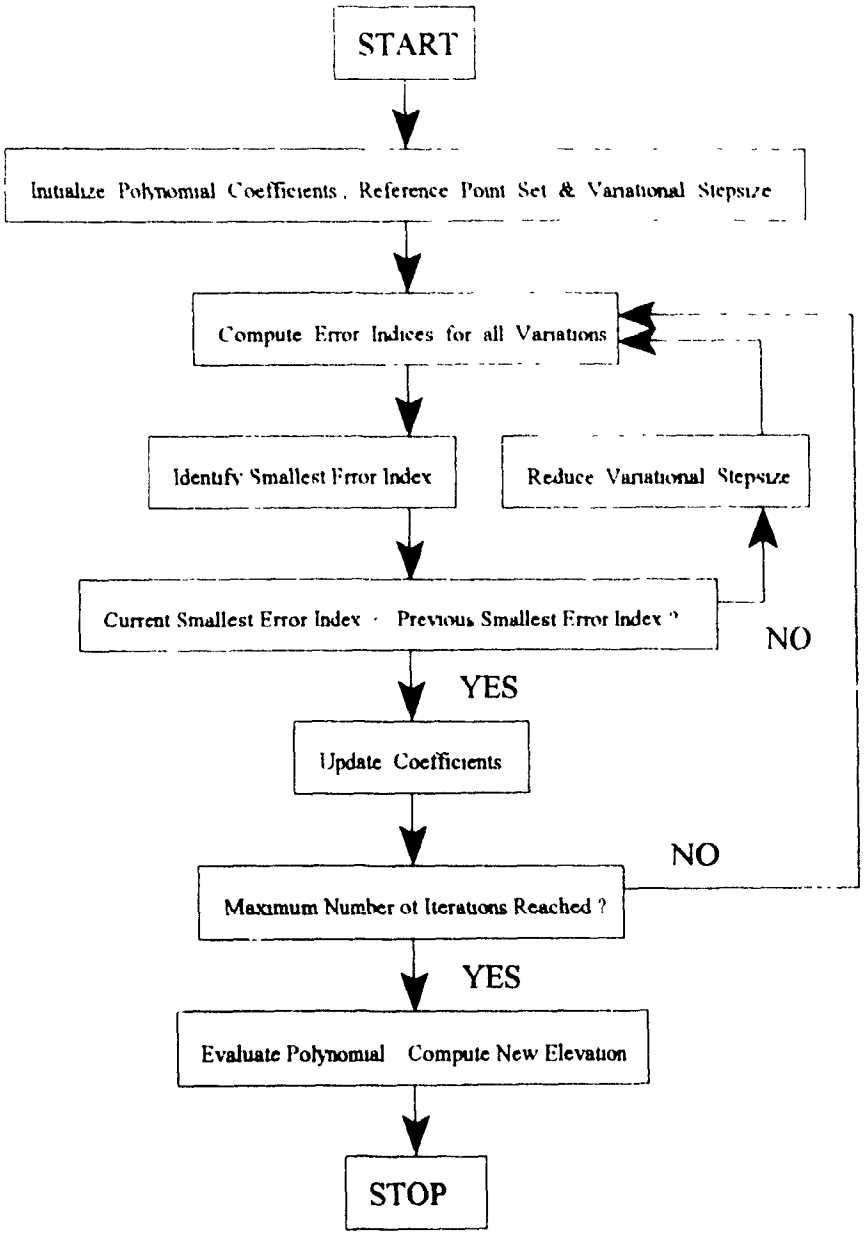


Figure 4.5: Flowchart of Computational Grid Point Replacement

previous iteration, the stepsize is reduced, and the process continued

The present method uses a summation of least squares error index, defined as

$$I_{err} = \sum_{i=1}^{N_p} ((z_{c_i} - z_{d_i})^2) \qquad (4\text{-}4)$$

In the above, z_{c_i} is the ith elevation computed using the current set of polynomial coefficients, and z_{d_i} is the ith digitized elevation This summation is performed over all nonrejected points within the reference disk, indicated in Eqn (4 5) as containing N_p points Eqn (4 4) therefore defines a measure of how well the interpolative polynomial surface function approximates local nonrejected digitized data Note that Eqn (4 4) is by no means the only possible error index definition A summation of absolute value of error could also have been used, expressed as follows

$$I_{err} = \sum_{i=1}^{N_p} |z_{c_i} - z_{d_i}| \qquad (4\text{-}5)$$

The method adopted offers no guarantee of global rather than local convergence Global convergence depends in part on the initial choice of coefficients In the present method, all coefficients are initialized to zero Recall that Figure 4 2 indicates maximum sensitivity to variations of the constant coefficient (a_{00} in Eqn (4 1)) for an evaluation point of $x_p = y_p = 0\ 5$ Constant coefficient variations therefore dominate coefficient set updates in the early stages of optimization In geometric terms, a flat level plane initially rises (or drops) from $z = 0$ to an elevation roughly bisecting the reference point dataset Afterwards, higher order terms become increasingly active, curving the MLES into successively better

approximations of local data

4.3 Splining and Toolpath Generation

4.3.1 Splining of Elevations and Normals

Elevations are computed from a distinct set of bicubic B-spline coefficients corresponding to each rectangular surface patch between digitized points on the MLES These coefficients are computed using a bicubic B-spline algorithm developed by Temple-Raston *et al* [63] An important feature of this algorithm is that it does not attempt *exact* fitting of knot points on the MLES due to the assumed existence of sampling errors Input to this splining algorithm is the MLES data, and output is one set of 16 coefficients representing the values of $a_{p,q}$ in Eqn (4 1) (with $N = 3$) for each MLES patch If the MLES is represented by an $N_1 \cdot N_2$ grid of elevations, there must be $(N_1 - 1) \cdot (N_2 - 1)$ surface patches, and hence $(N_1 - 1) \cdot (N_2 - 1)$ distinct sets of splining coefficients

Outward surface normal vector components are computed directly from elevation splining coefficients Since the splining coefficients for elevations on each patch of the MLES constitute representation of a polynomial surface function, it is possible to compute components of the unit normal vector $\boldsymbol{n}$ in each patch using (the reader is referred to Thomas and Finney [65])

$$\boldsymbol{n} = \frac{\boldsymbol{N}}{|\boldsymbol{N}|} = \frac{\frac{\partial z}{\partial x}\,\boldsymbol{i} + \frac{\partial z}{\partial y}\,\boldsymbol{j} + \boldsymbol{k}}{\sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 + 1}} \qquad (4\ 6)$$

In particular, with elevations in each surface patch given by a bicubic polynomial surface function, both derivative terms in Eqn (4.6) consist of polynomials with 12 coefficients. For example, $\frac{\partial z}{\partial x}$ is quadratic in x and cubic in y

$$\left.\begin{aligned}\frac{\partial z}{\partial x} = {} & 3a_{33}x^2y^3 + 2a_{23}xy^3 + a_{13}y^3 + \\ & 3a_{32}x^2y^2 + 2a_{22}xy^2 + a_{12}y^2 + \\ & 3a_{31}x^2y + 2a_{21}xy + a_{11}y + \\ & 3a_{30}x^2 + 2a_{20}x + a_{10}y\end{aligned}\right\} \qquad (4.7)$$

Output of the splining algorithm therefore consists of a file containing sets of 16 bicubic polynomial coefficients for elevations, and two additional files specifying sets of 12 polynomial coefficients for the derivatives $\frac{\partial z}{\partial x}$ and $\frac{\partial z}{\partial y}$, which may be substituted into Eqn (4.6) to give outward unit normal components

4.3.2 Corrective Machining Toolpath Generation

With elevations and normals known on the MLES, it only remains to produce a corrective machining toolpath. In the present research a demonstration of the deburring workcell tracing a machining path over a surface has been carried out using the MLES data, but actual machining has not been performed. Typical machining-related considerations such as cutter compensation and recommended depth of cut have therefore been omitted from this section.

Figure 4.6 shows a flowchart of machining path generation from MLES grid data. The

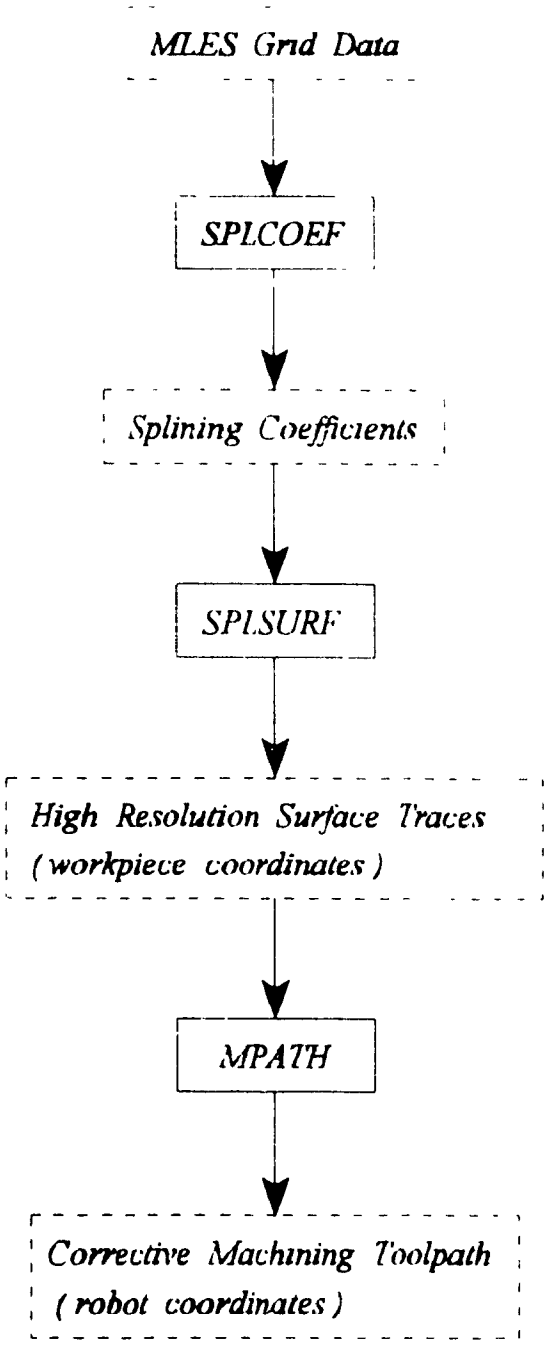


Figure 4.6: Flowchart of Corrective Machining Toolpath Generation

splining algorithm SPLCOEF uses the MLES grid data to generate the following (1) splining coefficients for elevations in each MLES patch, (2) splining coefficients for $\frac{\partial z}{\partial x}$ in each MLES patch (as in Eqn (4.7)) and (3) coefficients for $\frac{\partial z}{\partial y}$ This data in turn serves as input to the function SPLSURF as shown in Fig 4.6, which computes elevations and normals at any user-specified resolution along traces of the MLES

A third function named MPATH computes the surface-tracing path (X, Y, Z, A, B, V), where the first five columns are position and orientation of the tool and column six is the machining point speed The workpiece-to-robot coordinate transformation is performed first, then robot wrist joint displacements are computed from normal components Output of MPATH when saved in a file can be used as the Yamaha robot toolpath when executed by the controller software

4.4 Summary

A method has been presented for automated determination of the desired reconditioned geometry for a damaged component that requires corrective machining This method is designed for surfaces which contain localized extraneous features superimposed on warping deformations To justify selection of the techniques employed, an analytical background has been provided The present method for determination of desired component geometry is based on a multivariate optimization technique in which new elevations are computed to replace those previously identified as defective Output of the autonomous feature recognition algorithm of Chapter 3 (the feature map) is therefore suitable as input to the reconditioning algorithm The present work links computational grid point replacement with splining over

the resulting Maximum Likelihood Eventual Surface (MLES) to generate high-resolution traces of MLES elevations and normals This data is used as a basis for corrective machining toolpath generation Corrective machining has not been documented in the present work, but a demonstration of surface tracing respecting normals was successfully carried out based on the algorithms of Chapters 2, 3 and 4 Integrated use of digitizing, recognition and reconditioning algorithms is thus seen as a methodology for refurbishing certain types of damaged components Chapter 5 next presents experimental verification for a variety of surfaces and features

CHAPTER 5 EXPERIMENTAL VERIFICATION

5.1 Introduction

This chapter presents results of experimental case studies to demonstrate the ability of the modified workcell controller to

(i) digitize component surfaces using the automated digitizing method described in Chapter 2,

(ii) perform autonomous recognition of extraneous features on a variety of machined workpiece surfaces using the method described in Chapter 3,

(iii) demonstrate the accuracy obtainable using the surface reconditioning methods described in Chapter 4

This chapter therefore provides experimental verification of automated digitizing, autonomous feature recognition, and automated surface reconditioning, which together constitute a methodology for automated refurbishment of complex precision-machined components All digitizing for these studies was carried out using the CIC robotic deburring workcell Three precision-machined test surfaces have been used (1) an inclined plane, (2) a parabolic half-trench, and (3) a biquartic saddle

These three surfaces are representative of progressively more complex geometries, allowing for a thorough evaluation of the refurbishment methodology presented in this thesis. The biquartic saddle is particularly significant because it contains essential geometric features of typical gas turbine rotor blades, which are a prime candidate for industrial application of this research Test surfaces have been selected over actual components for use in these case

studies to avoid problems with probe design for interference-free accessibility As an example, consider the deep recesses and narrow gaps between rotor blades in Figure 1 1

Visualizations are provided for digitized grid data representing the three test surfaces in their undamaged condition (i e , with no extraneous features) For comparison, photos of these surfaces, set up on the worktable of the CIC deburring workcell are also shown Two types of feature have been introduced, namely a lump of material (quick-drying epoxy glue), which stands proud of the surrounding surface, and a depression (drilled hole) Three classes of feature are examined (1) a single localized lump, (2) multiple lumps, and (3) elongated beads or ridges These classes represent typical expected refurbishment scenarios in which a damaged area must be ground down and built up with weld prior to corrective machining As a final demonstration, a relatively complex feature has been included in the form of a letter "S" embossed on the biquartic saddle surface

All case studies of this chapter use 31 · 31 grids, with a digitizing stepsize $\Delta x = \Delta y$ = 2 71 mm (0 107 in) This choice of grid and stepsize is based on the modelling error analysis recommendation (Section 2.4 1) computed from a high-resolution database of elevations originally used for machining the biquartic saddle surface To achieve this grid reduction, a modelling error of ± 0 0254 mm (± 0 00100 in) has been specified The same grid and stepsize are also used for the inclined plane and parabolic half-trench surfaces examined in this chapter, although no design data is available for modelling error analysis of these surfaces Note that this recommended stepsize is much larger than the manufacturer's claim of ± 0 1 mm (0.004 in) for repeatability of the Yamaha Zeta-1 deburring robot All surface plots presented in this chapter have a consistently "smooth" visual appearance due to the fine

digitizing stepsize used

All ten case studies of this chapter demonstrate the reconditioning algorithm of Chapter 4 using reference point datasets containing 30 points, and 2000 iterations of a biquadratic polynomial fit Plots of required computation times and observed error trends are also given for the surface reconditioning process Required computation times are shown for various degrees of polynomial surface fit Observed error trends are shown for (1) varying the degree of polynomial surface fitting function, (2) varying the optimization algorithm iteration count, and (3) varying the reference disk radius Detailed discussion of these plots is given in Section 5 4 5

5.2 Digitizing with On-Line Path Generation

Three digitizing examples are presented in this section an inclined plane surface (Figures 5 1 and 5 2), a parabolic half-trench surface (Figures 5 3 and 5 4) and a biquartic saddle surface (Figures 5 5 and 5 6) All elevations have been obtained using the on-line path generation method described in Chapter 2, Section 2 3 No extraneous features are present on these surfaces Digitized elevations shown in Figures 5 2, 5 4 and 5 6 are therefore referenced in Section 5.4 as a basis for performance evaluation of the surface-fitting techniques discussed in Chapter 4, Section 4.2

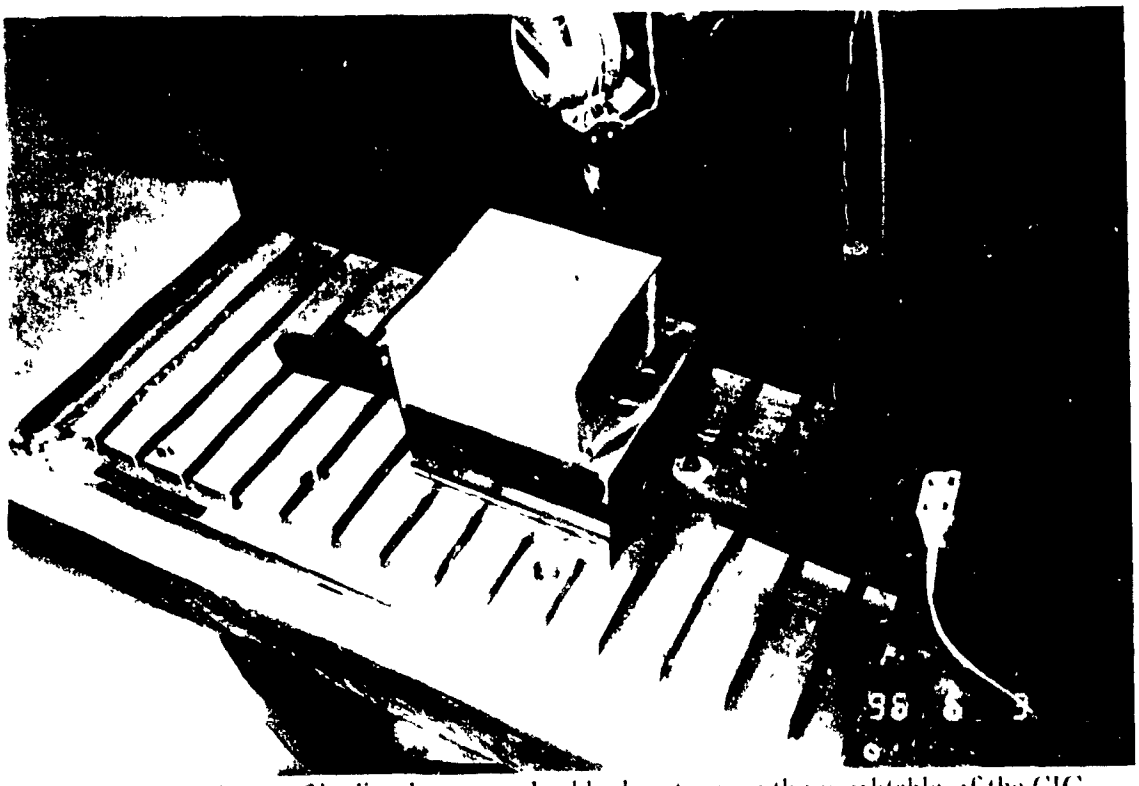


Figure 5.1: Photo of inclined rectangular block set up on the worktable of the CIC deburring workcell for digitizing

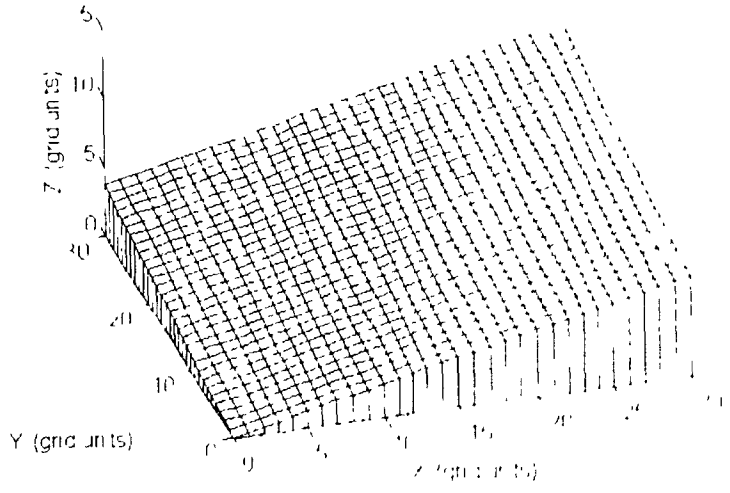


Figure 5.2: Digitized 31 × 31 grid of elevations for inclined rectangular block shown in Figure 5.1, with $\Delta x = \Delta y = 2.71$ mm (0.107 in)

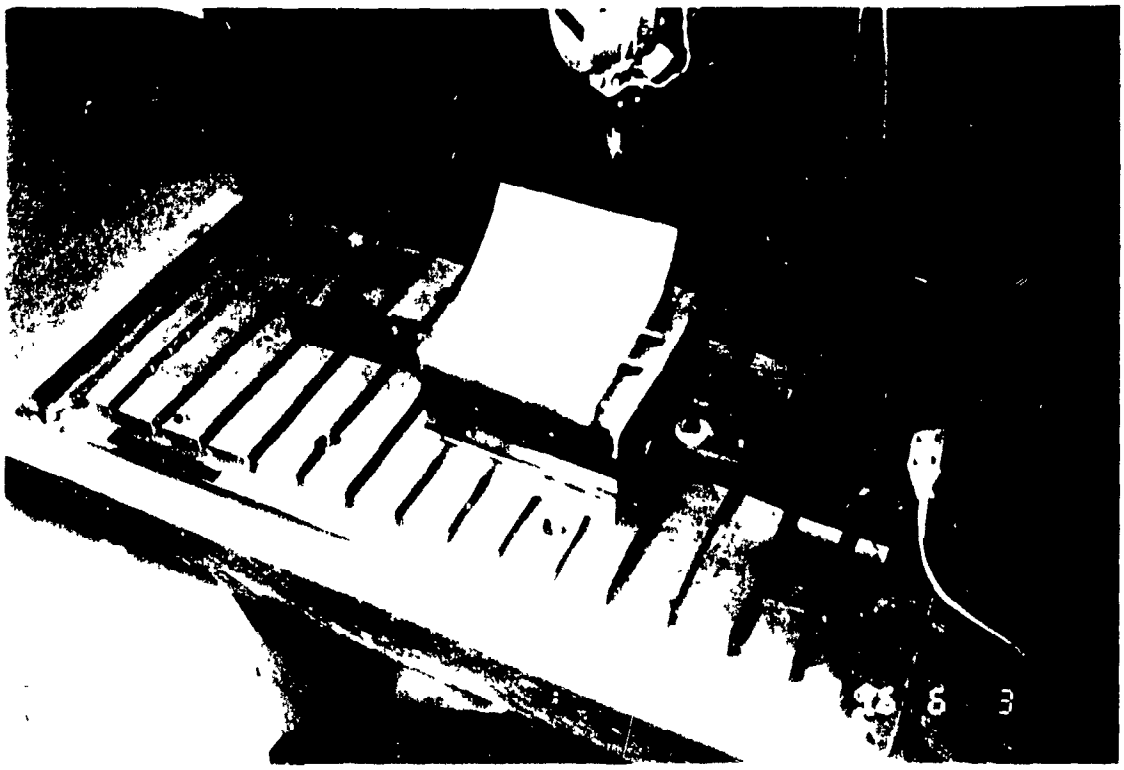


Figure 5.3: Photo of parabolic half-trench surface set up on the worktable of the CIC deburring workcell for digitizing

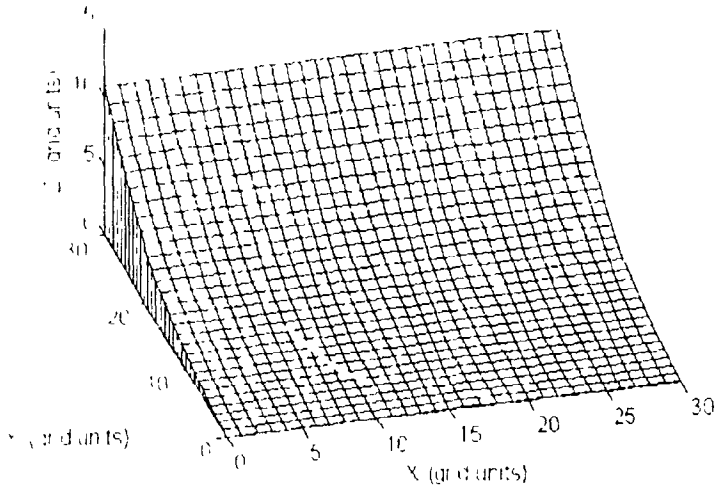


Figure 5.4: Digitized 31 × 31 grid of elevations for parabolic half-trench surface shown in Figure 5.3, with $\Delta x = \Delta y = 2.71$ mm (0.107 in)

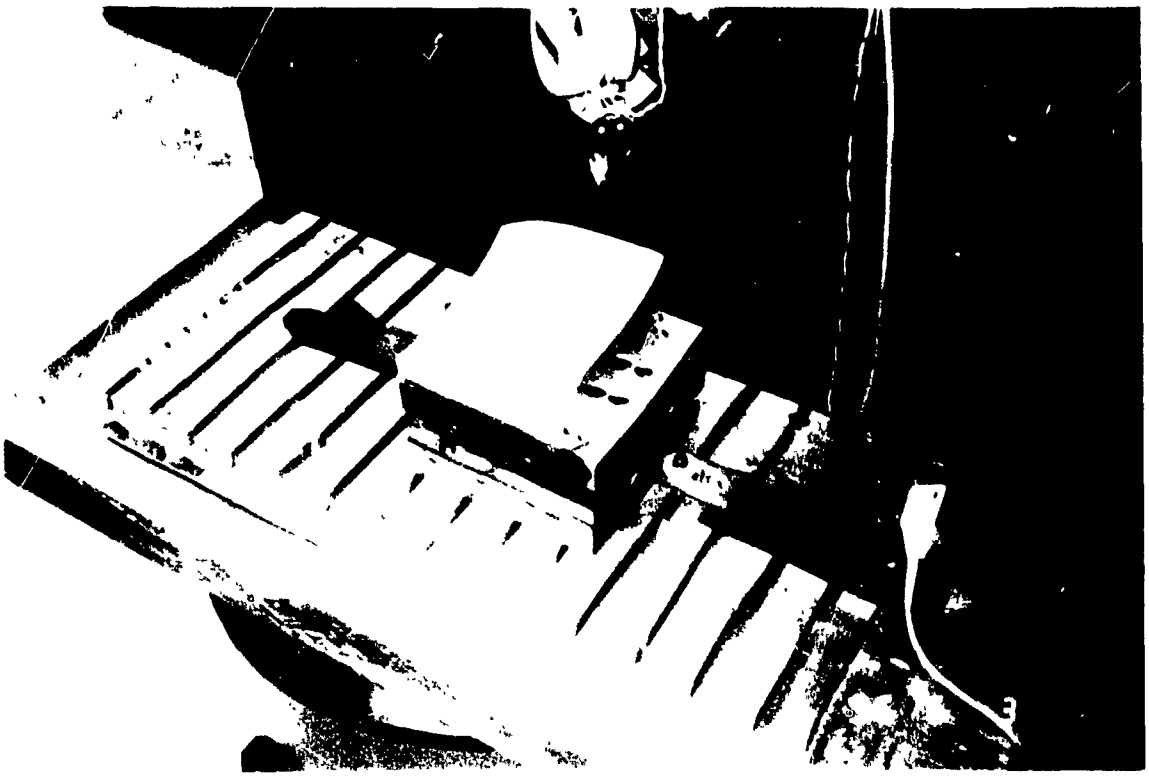


Figure 5.5: Photo of biquartic saddle surface set up on the worktable of the CIC deburring workcell for digitizing

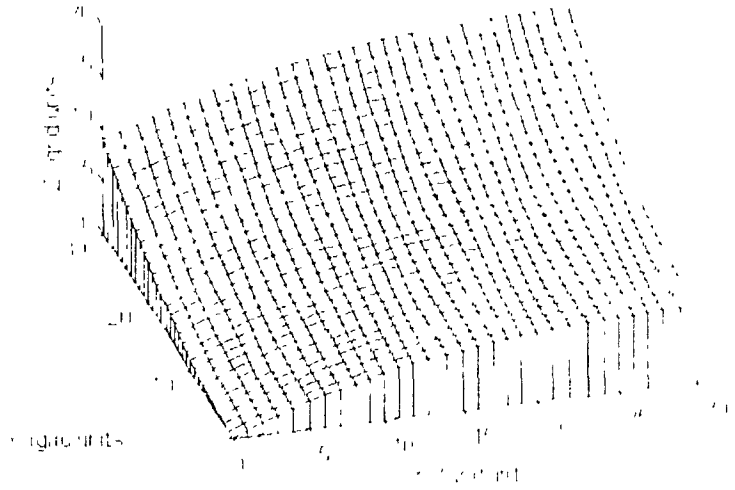


Figure 5.6: Digitized 31 × 31 grid of elevations for biquartic saddle surface shown in Figure 5.5, with $\Delta x = \Delta y = 2.71$ mm (0.107 in)

5.3 Autonomous Feature Recognition

5.3.1 Single Localized Point Features

Figures 5 8 and 5 9 show the results of autonomous feature recognition for the inclined plane surface of Figure 5 7, which contains a single localized point feature This feature is a 9 0 mm (0 35 in) diameter hole drilled in the workpiece shown in Figure 5 1, and comprises 11 digitized grid points Figures 5 11 and 5 12 show the results of autonomous feature recognition for the parabolic half-trench surface of Figure 5 10, which contains a single localized point feature This feature is a lump of epoxy glue applied to the workpiece shown in Figure 5 3, and comprises 1 1 digitized grid points Figures 5 14 and 5 15 show the results of autonomous feature recognition for the biquartic saddle surface of Figure 5 13, which contains a single localized point feature This feature is a lump of epoxy glue applied to the workpiece shown in Figure 5 5, and comprises 16 digitized grid points Figures 5 8, 5 9, 5 11, 5 12, 5 14 and 5 15 demonstrate that the autonomous feature recognition process described in Chapter 3 is able to reliably identify single localized point features on all three examples in this section Figures 5 8, 5 11 and 5 14 show the recognition results in plan view, and Figures 5 9, 5 12 and 5 15 show the recognition results as profiles of shifted elevations All digitized grids shown in this section contain 31 × 31 points, with stepsize $\Delta x = \Delta y = 2\,71$ mm (0 107 in).

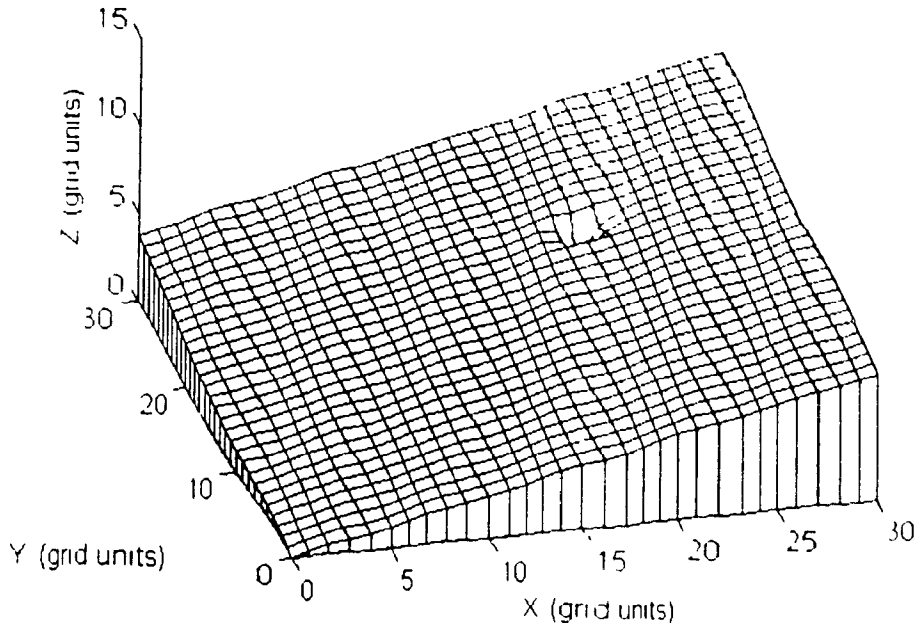


Figure 5.7: Digitized grid of inclined plane surface with localized point feature

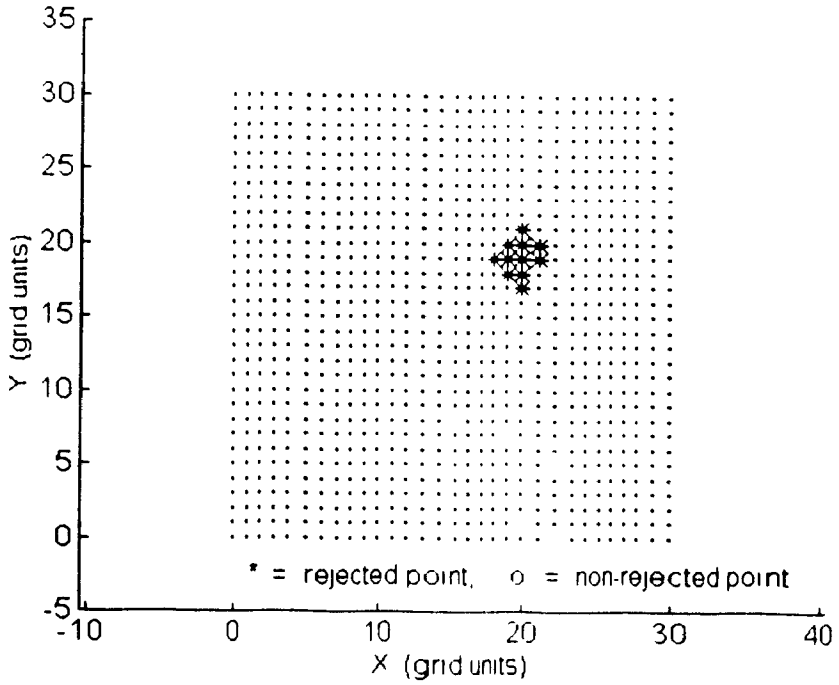


Figure 5.8: Feature map corresponding to Figure 5 7 (plan view)

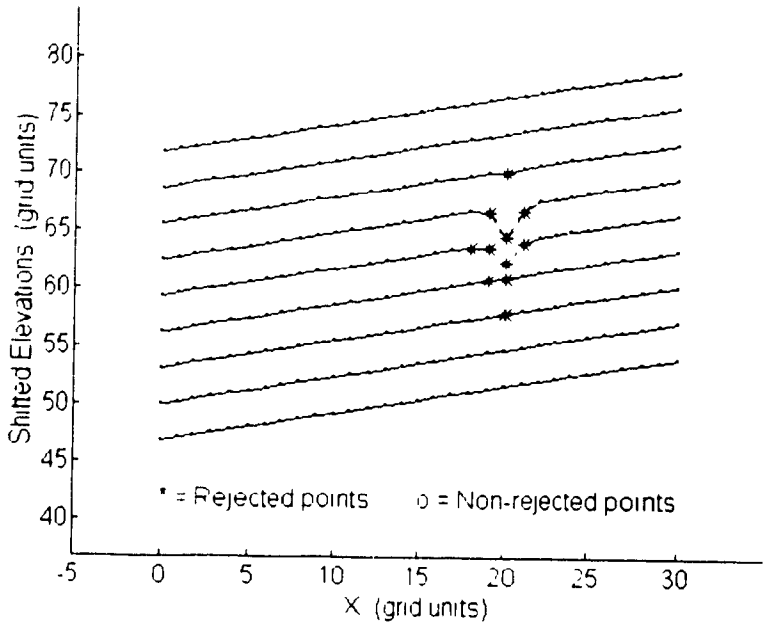


Figure 5.9: Feature map corresponding to Figure 5 7 (digitized traces with shifted elevations)

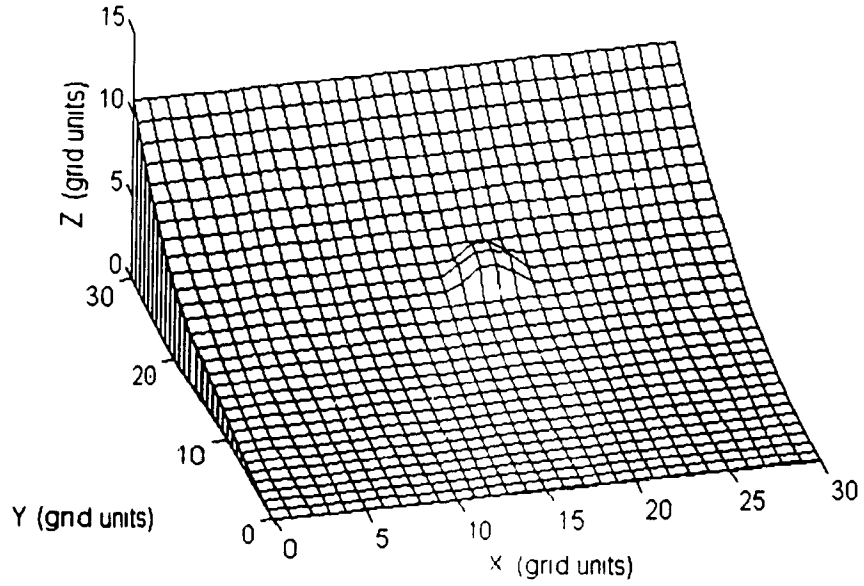


Figure 5.10: Digitized grid of parabolic half-trench surface with localized point feature

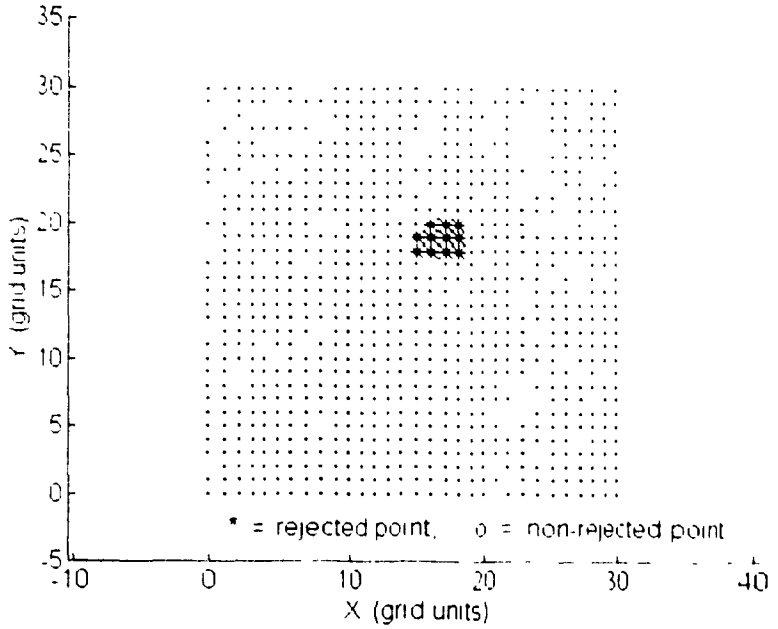


Figure 5.11: Feature map corresponding to Figure 5.10 (plan view)

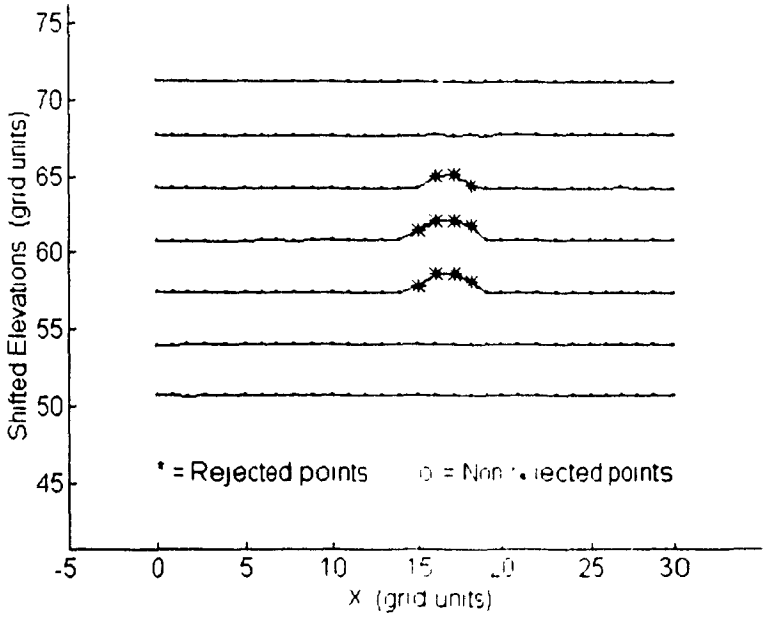


Figure 5.12: Feature map corresponding to Figure 5.10 (digitized traces with shifted elevations)

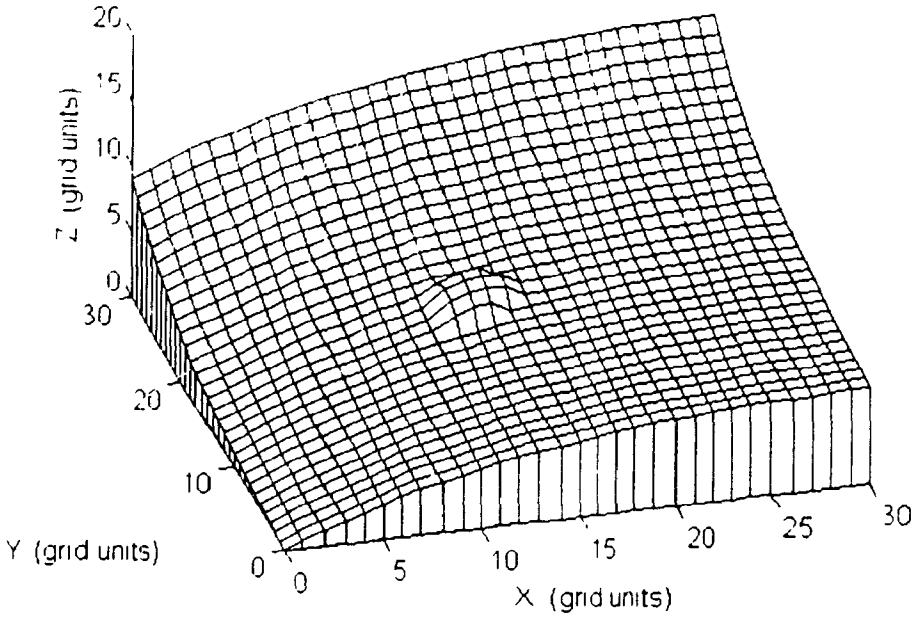


Figure 5.13: Digitized grid of biquartic saddle surface with localized point feature

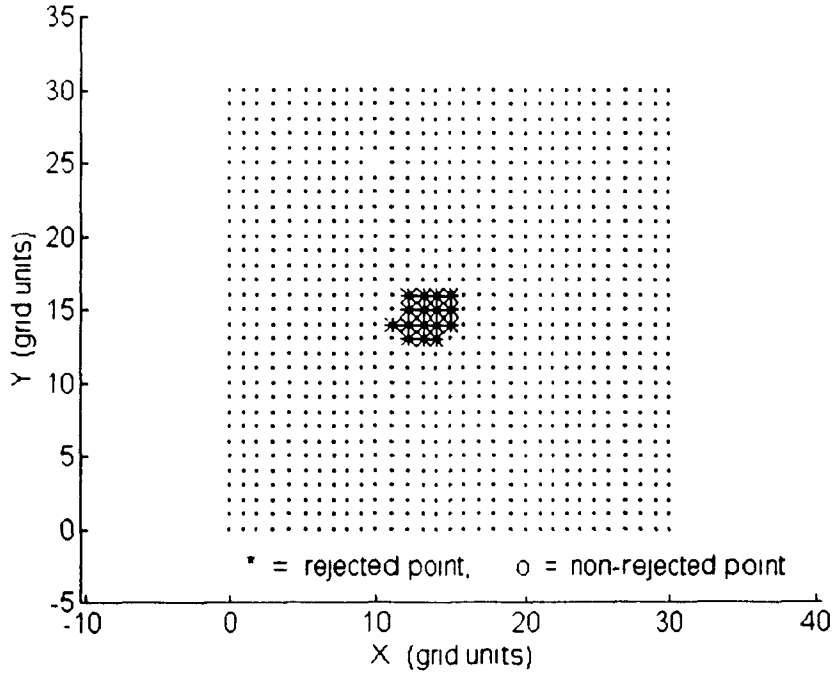


Figure 5.14: Feature map corresponding to Figure 5 13 (plan view)

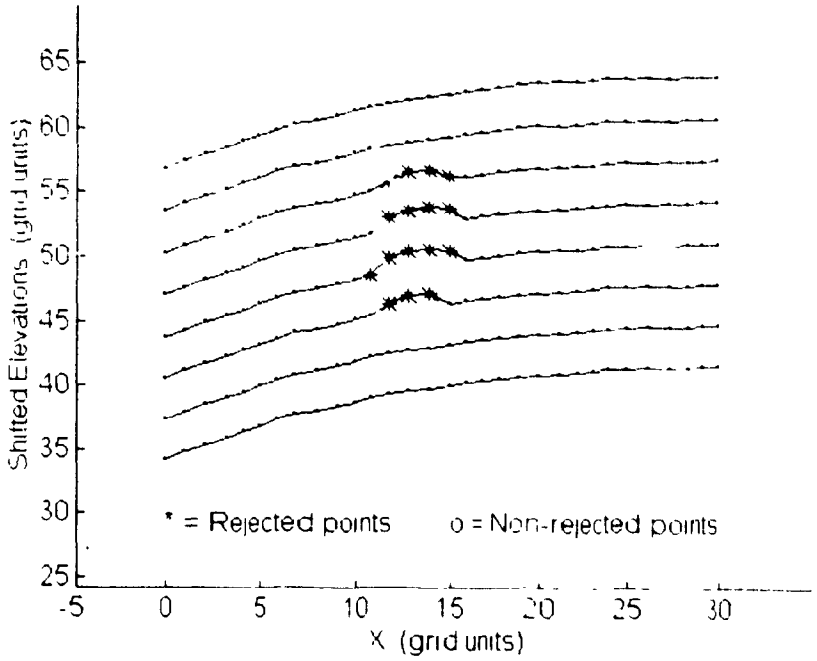


Figure 5.15: Feature map corresponding to Figure 5.13 (digitized traces with shifted elevations)

5.3.2 Multiple Localized Point Features

Figures 5.17 and 5.18 show the results of autonomous feature recognition for the inclined plane surface of Figure 5.16, which contains multiple localized point features. These features are a drilled hole and three lumps of epoxy glue applied to the workpiece shown in Figure 5.1, and comprise a total of 44 digitized grid points. Figures 5.20 and 5.21 show the results of autonomous feature recognition for the parabolic half-trench surface of Figure 5.19, which contains multiple localized point features. These features are four lumps of epoxy glue applied to the workpiece shown in Figure 5.3, and comprise a total of 62 digitized grid points. Figures 5.23 and 5.24 show the results of autonomous feature recognition for the

biquartic saddle surface of Figure 5 23, which contains multiple localized point features These features are four lumps of epoxy glue applied to the surface of Figure 5 5, and comprise a total of 51 digitized grid points Figures 5 17, 5 18, 5 20, 5 21, 5 23 and 5 24 demonstrate that the autonomous feature recognition process described in Chapter 3 is able to reliably identify multiple localized point features on all three examples in this section Figures 5 17, 5 20 and 5 23 show the recognition results in plan view, and Figures 5 18, 5 21 and 5 24 show the recognition results as profiles of shifted elevations All digitized grids shown in this section contain 31 × 31 points, with stepsize $\Delta x = \Delta y = 2\ 71$ mm (0 107 in)

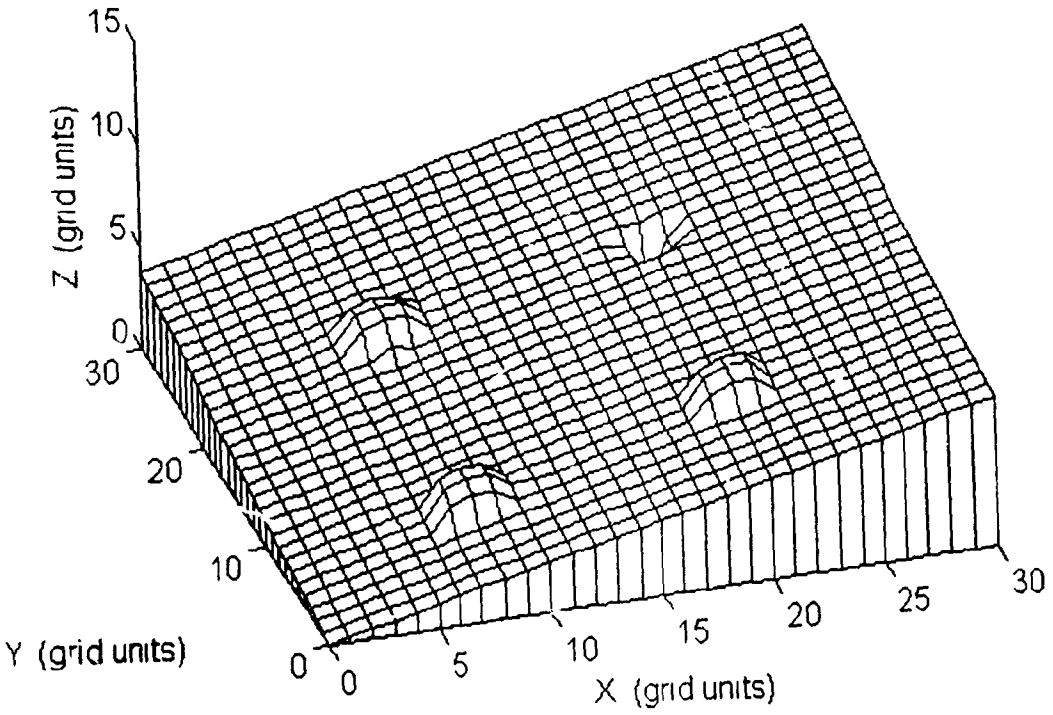


Figure 5.16: Digitized grid of inclined plane surface with multiple localized point features

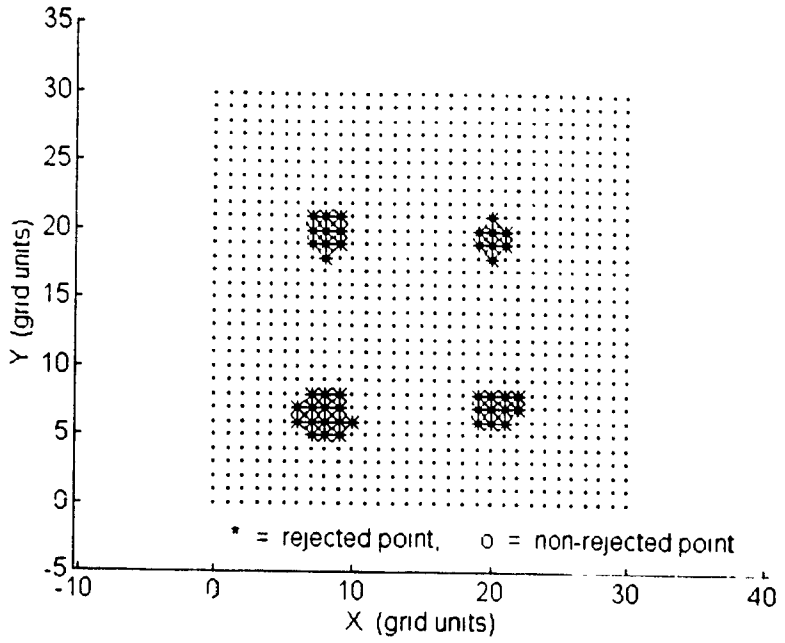


Figure 5.17: Feature map corresponding to Figure 5 16 (plan view)

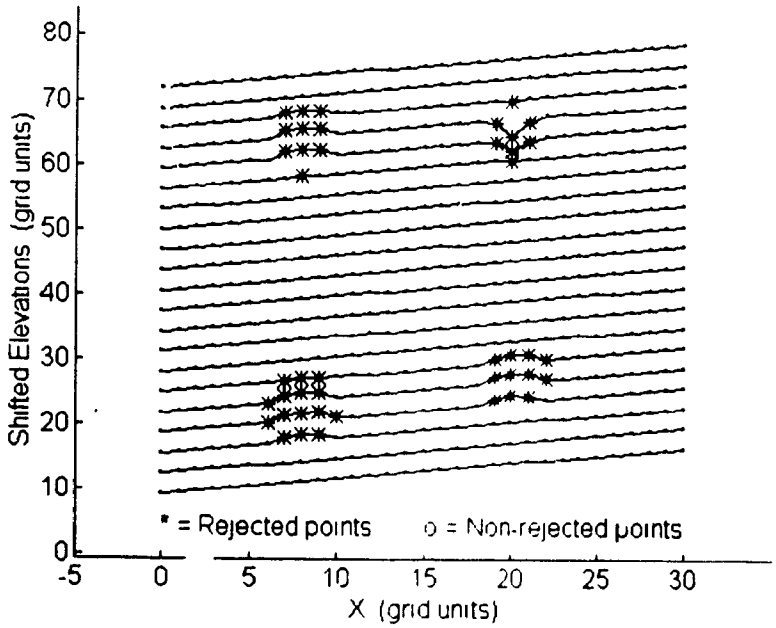


Figure 5.18: Feature map corresponding to Figure 5 16 (digitized traces with shifted elevations)

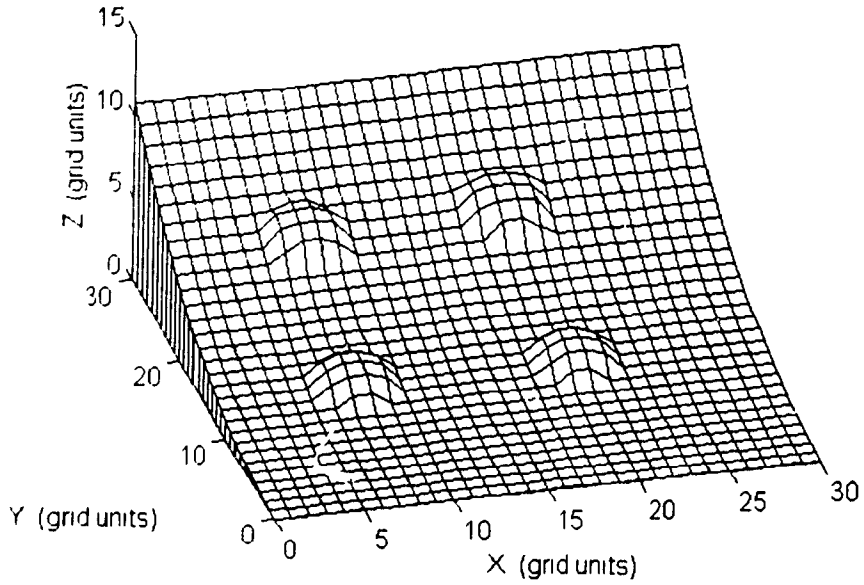


Figure 5.19: Digitized grid of parabolic half-trench surface with multiple localized point features

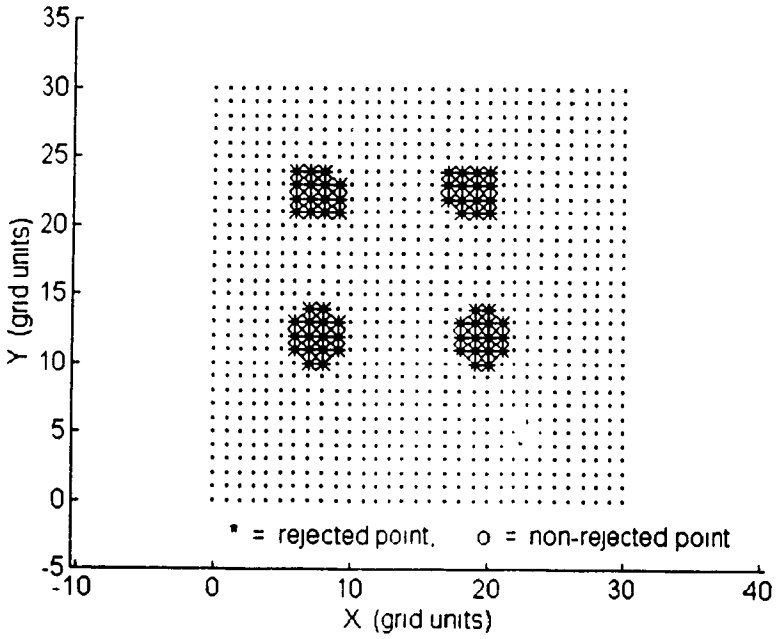


Figure 5.20: Feature map corresponding to Figure 5.19 (plan view).

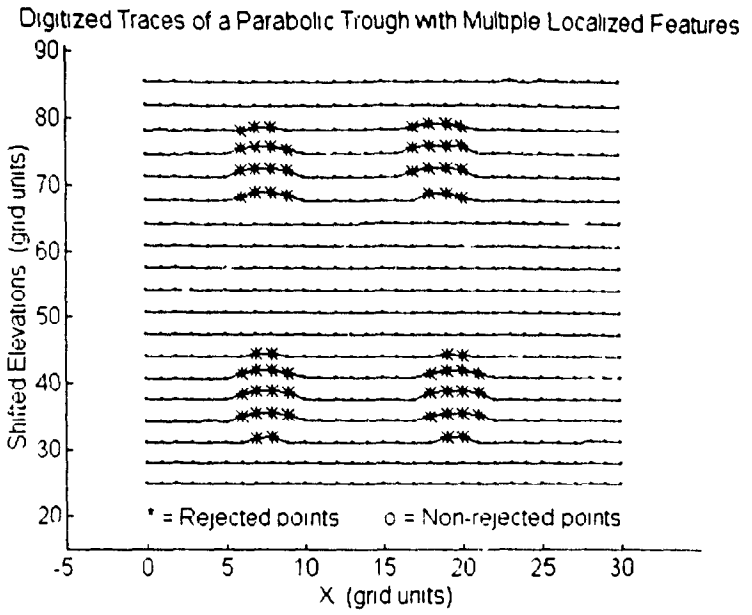


Figure 5.21: Feature map corresponding to Figure 5 19 (digitized traces with shifted elevations)

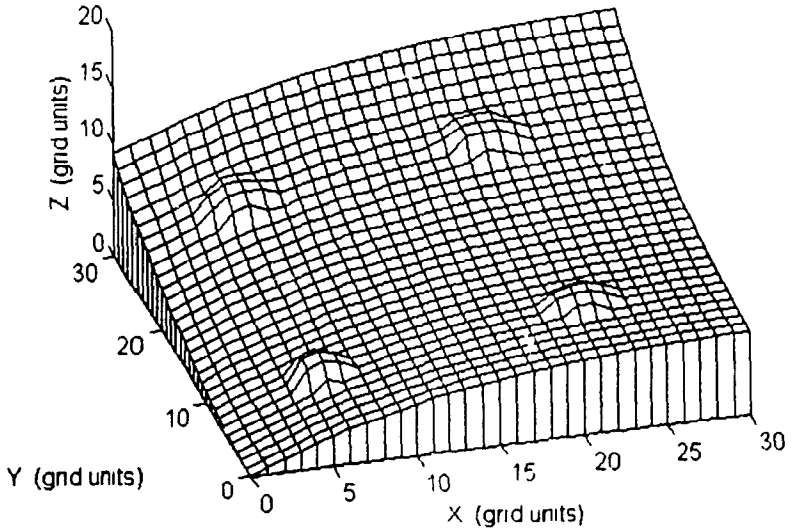


Figure 5.22: Digitized grid of biquartic saddle surface with multiple localized point features.

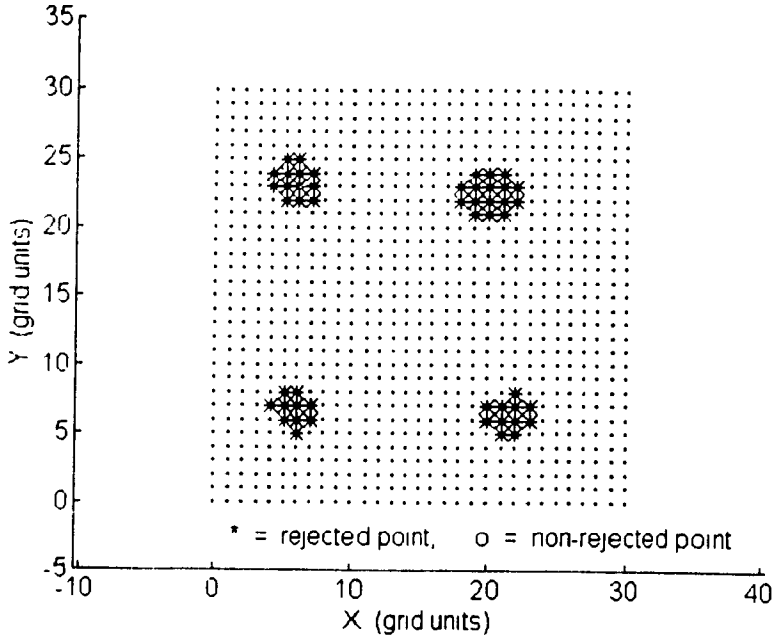


Figure 5.23: Feature map corresponding to Figure 5 22 (plan view)

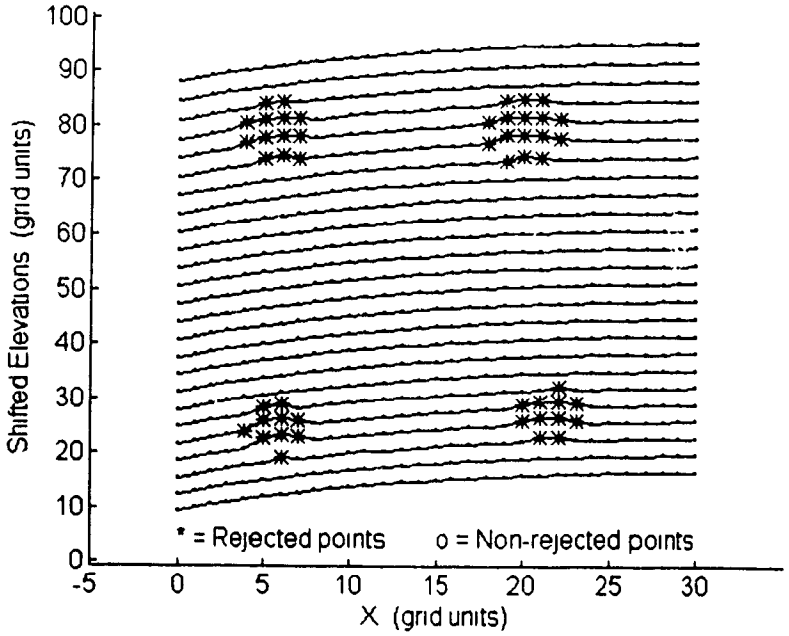


Figure 5.24: Feature map corresponding to Figure 5 22 (digitized traces with shifted elevations).

5.3.3 Linear Features

Figures 5 26 and 5 27 show the results of autonomous feature recognition for the inclined plane surface of Figure 5 25, which contains a linear feature This feature is a bead of epoxy glue applied to the workpiece shown in Figure 5 1, and comprises 35 digitized grid points Figures 5 29 and 5 30 show the results of autonomous feature recognition for the parabolic half-trench surface of Figure 5 28, which contains a linear feature This feature is a bead of epoxy glue applied to the workpiece shown in Figure 5 3, and comprises 40 digitized grid points Figures 5 32 and 5 33 show the results of autonomous feature recognition for the biquartic saddle surface of Figure 5 31, which contains a linear feature This feature is a bead of epoxy glue applied to the workpiece shown in Figure 5 5, and comprises 31 digitized grid points. Figures 5 26, 5 27, 5 29, 5 30, 5 32 and 5 33 demonstrate that the autonomous feature recognition process described in Chapter 3 is able to reliably identify linear features on all three examples in this section Figures 5 26, 5 29 and 5 32 show the recognition results in plan view, and Figures 5 27, 5 30 and 5 33 show the recognition results as profiles of shifted elevations All digitized grids shown in this section contain 31 × 31 points, with stepsize $\Delta x = \Delta y = 2\ 71$ mm (0 107 in).

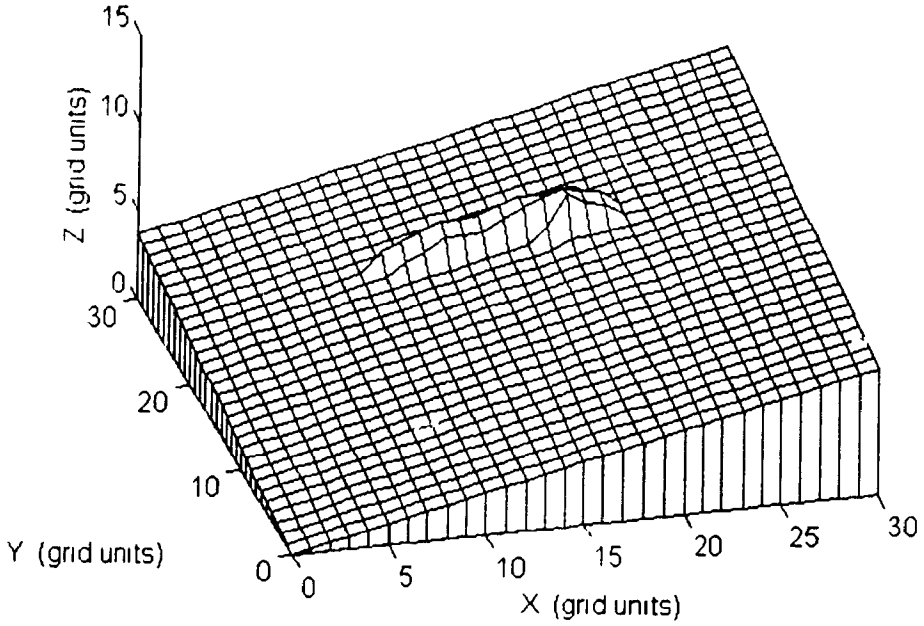


Figure 5.25: Digitized grid of inclined plane surface with a linear feature

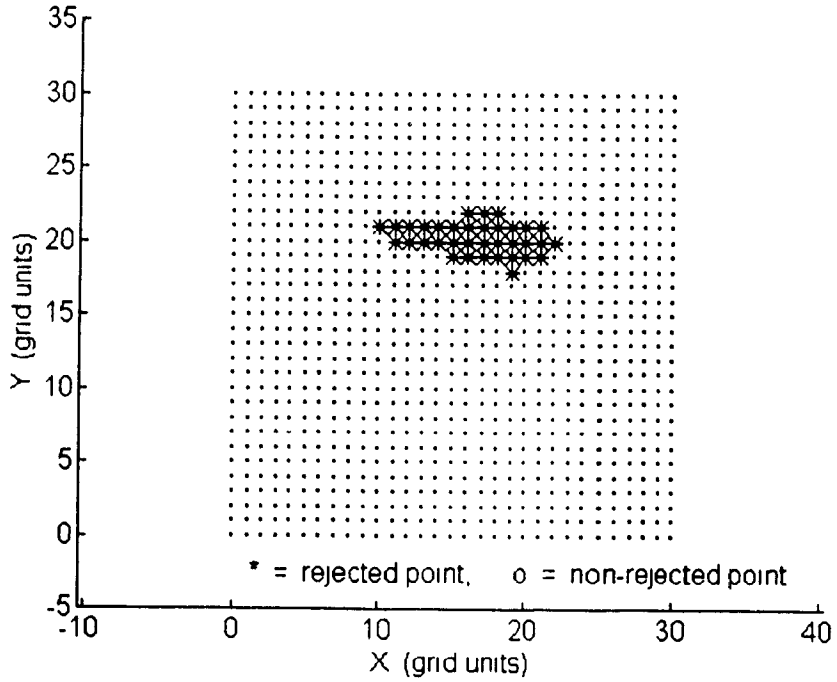


Figure 5.26: Feature map corresponding to Figure 5.25 (plan view).

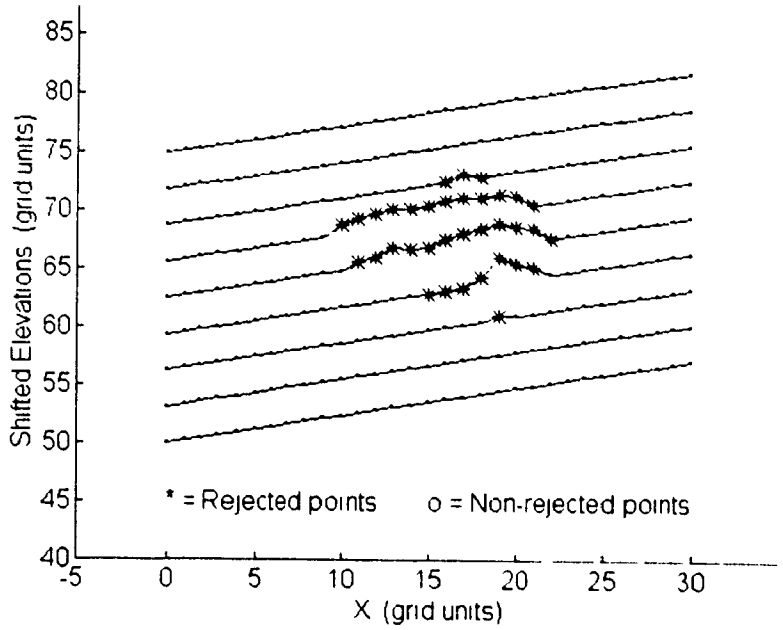


Figure 5.27: Feature map corresponding to Figure 5 25 (digitized traces with shifted elevations)

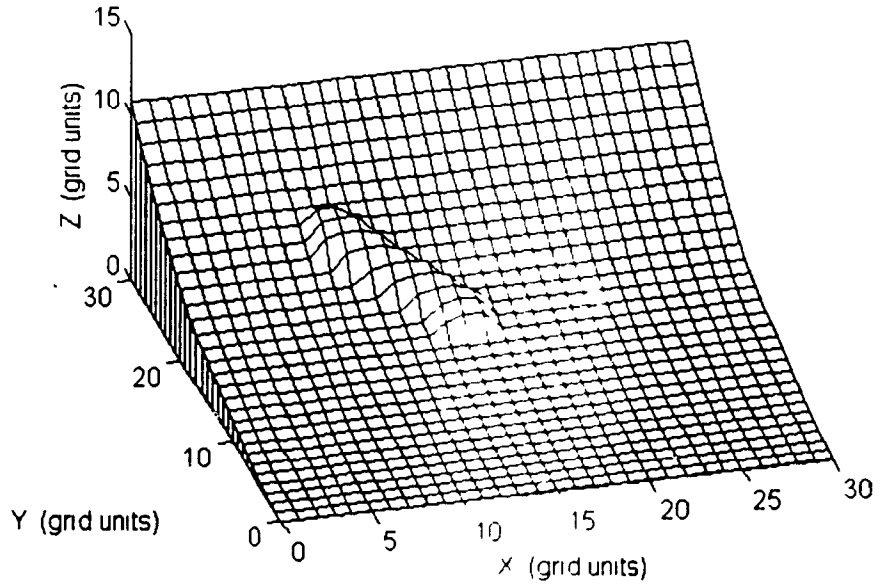


Figure 5.28: Digitized grid of parabolic half-trench surface with a linear feature

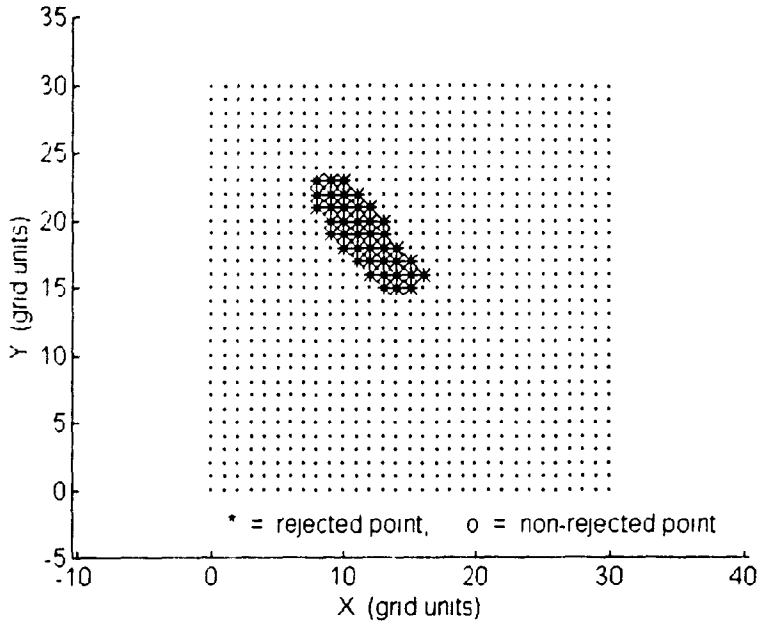


Figure 5.29: Feature map corresponding to Figure 5 28 (plan view)

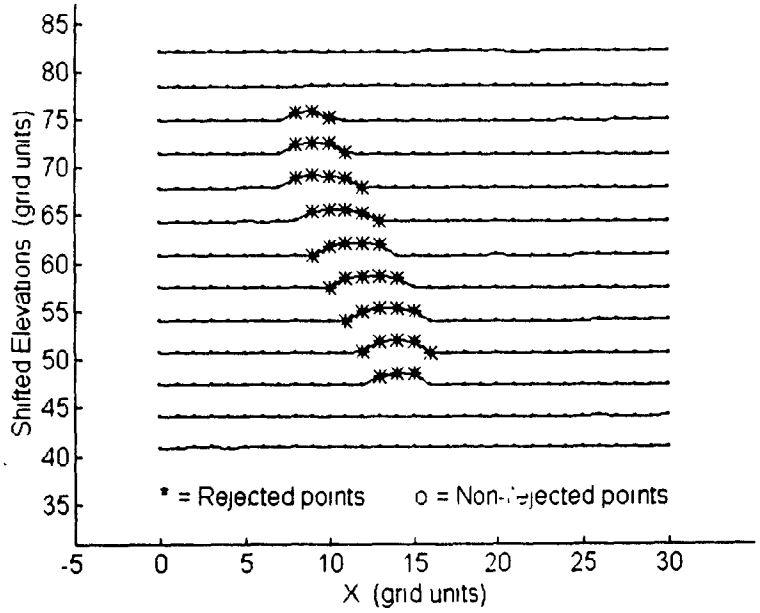


Figure 5.30: Feature map corresponding to Figure 5 28 (digitized traces with shifted elevations).

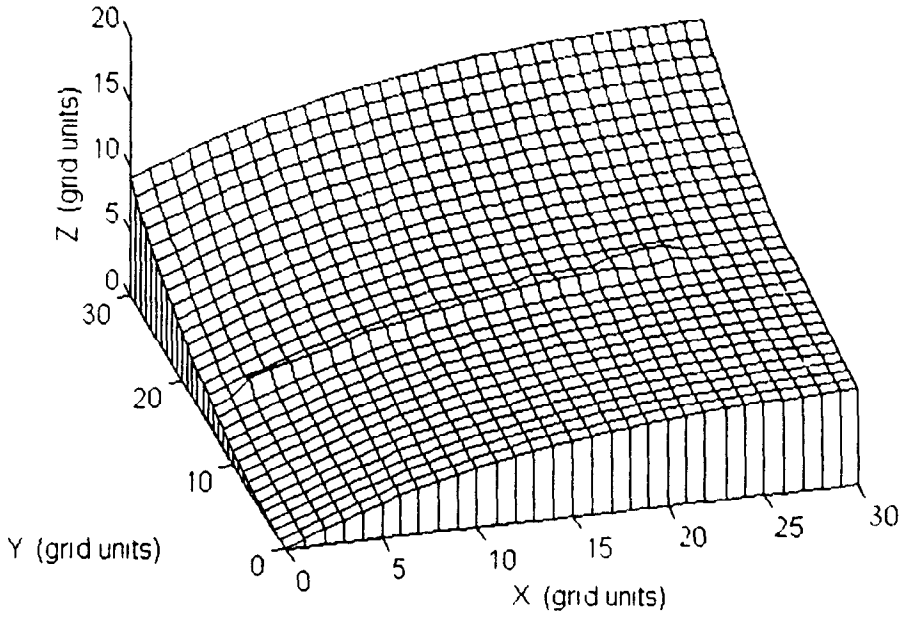


Figure 5.31: Digitized grid of biquartic saddle surface with a linear feature

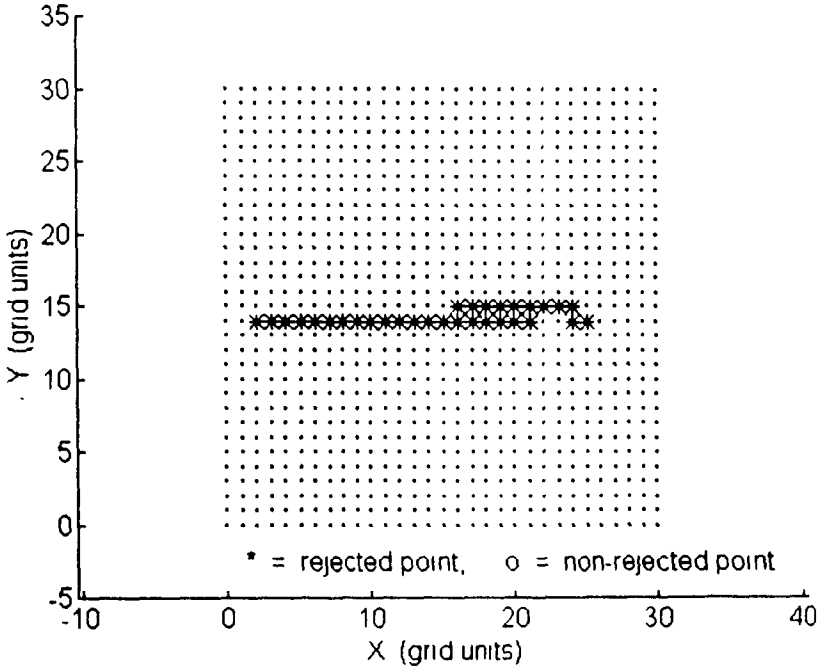


Figure 5.32: Feature map corresponding to Figure 5 31 (plan view)

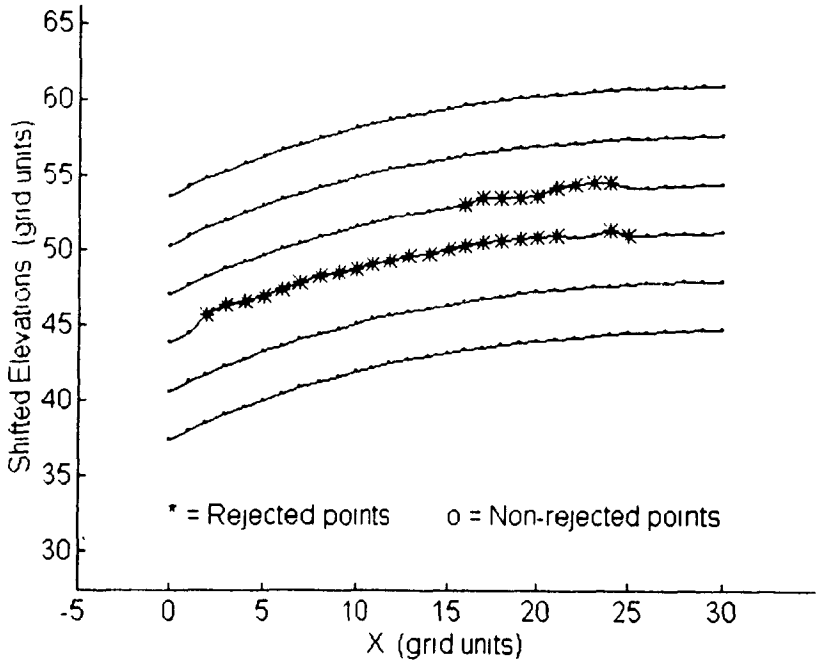


Figure 5.33: Feature map corresponding to Figure 5 31 (digitized traces with shifted elevations).

5.3.4 Biquartic Saddle with an S-Shaped Feature

Figures 5 35 and 5 36 show the results of autonomous feature recognition for the biquartic saddle surface of Figure 5 34, which contains an S-shaped feature. This feature is a bead of epoxy glue applied to the workpiece shown in Figure 5 5, and comprises 65 digitized grid points. Figures 5.35 and 5.36 demonstrate that the autonomous feature recognition process described in Chapter 3 is able to reliably identify the complex S-shaped feature Figure 5 35 shows the recognition results in plan view, and Figure 5 36 shows the recognition results as profiles of shifted elevations. All digitized grids shown in this section contain 31 × 31 points, with stepsize $\Delta x = \Delta y = 2.71$ mm (0.107 in)

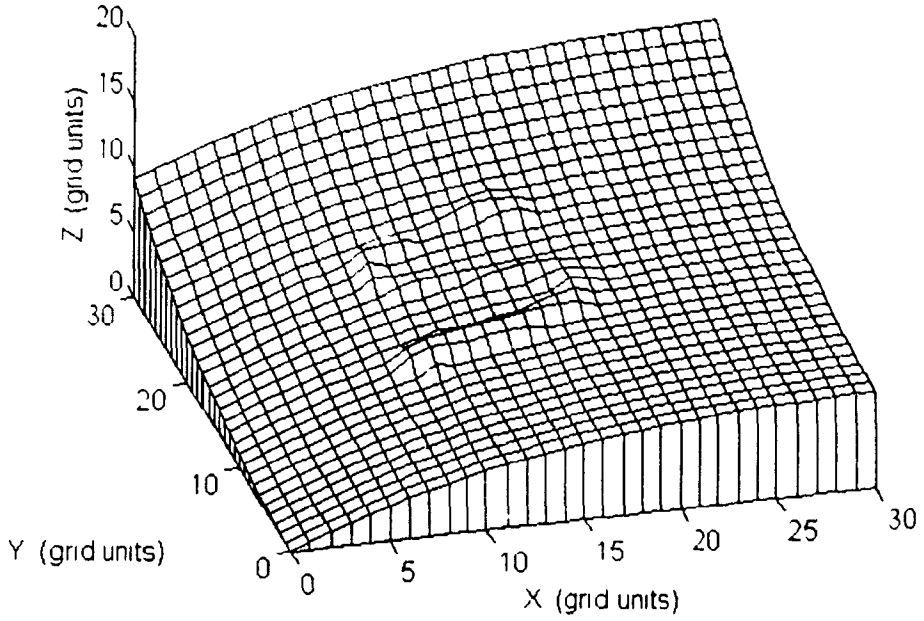


Figure 5.34: Digitized grid of biquartic saddle surface with an S-shaped feature

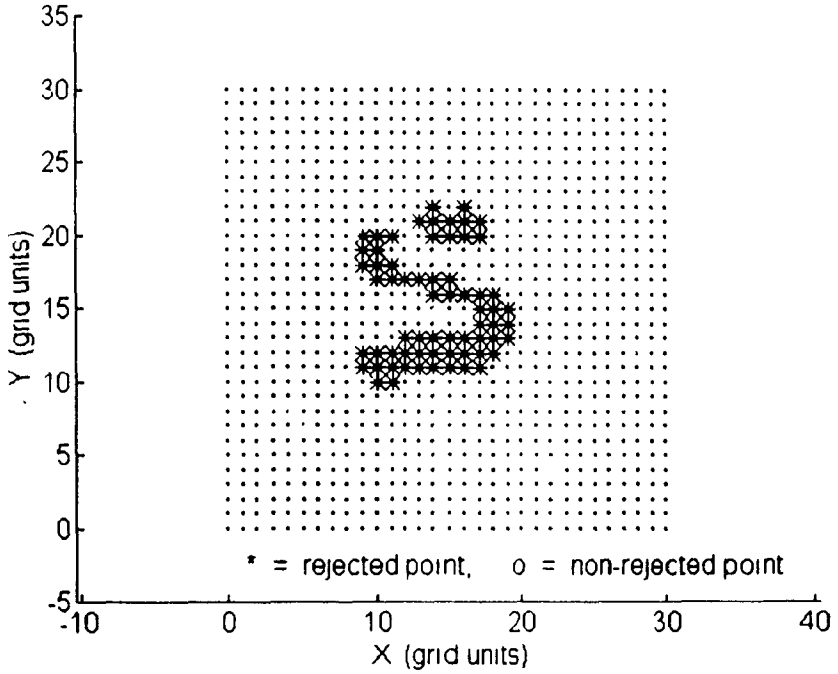


Figure 5.35: Feature map corresponding to Figure 5 34 (plan view)

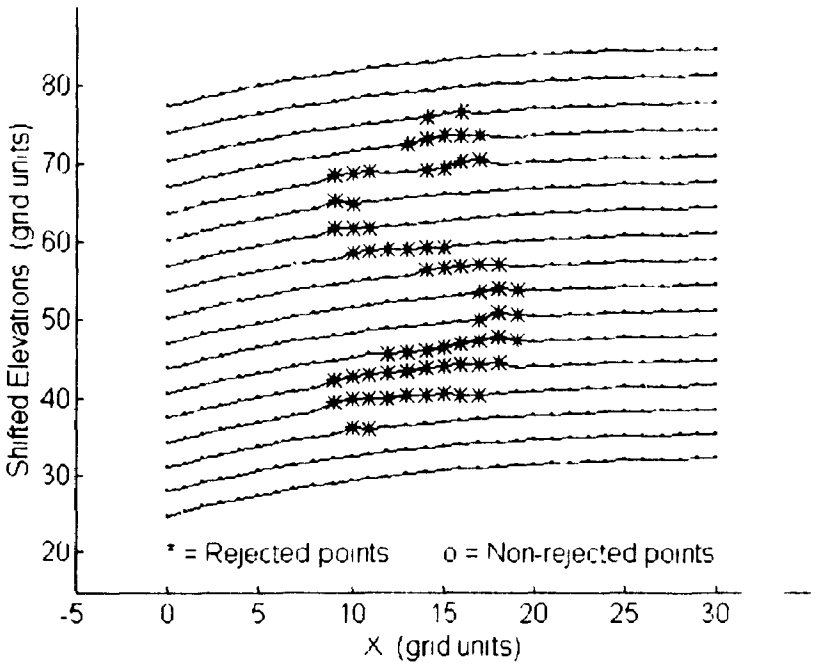


Figure 5.36: Feature map corresponding to Figure 5 34 (digitized traces with shifted elevations)

5.4 Automated Surface Reconditioning

5.4.1 Single Localized Point Features

Figure 5 37 shows the MLES corresponding to the inclined plane surface with single localized point feature of Figure 5 7 Figure 5 38 compares the MLES of Figure 5 37 with the nominal data of Figure 5 2. The MLES of Figure 5 37 is seen to be within an error band of 0.055 grid units (0 15 mm, 0.0059 in) relative to the nominal data of Figure 5 2, and the maximum MLES error in this example is 0 037 grid units (0 10 mm, 0.0039 in). Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0 074 grid units (± 0 20 mm, ± 0 0079 in) of the nominal data

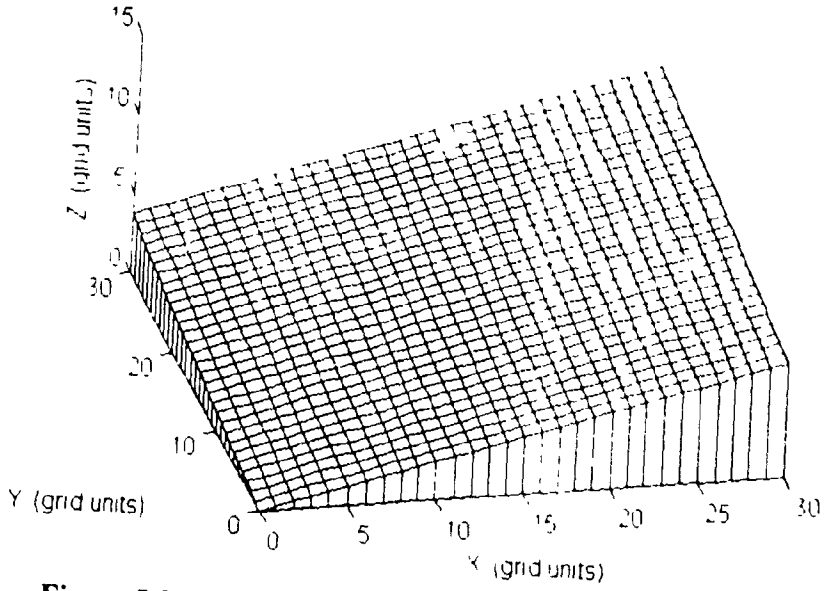


Figure 5.37: MLES corresponding to the surface of Figure 5 7

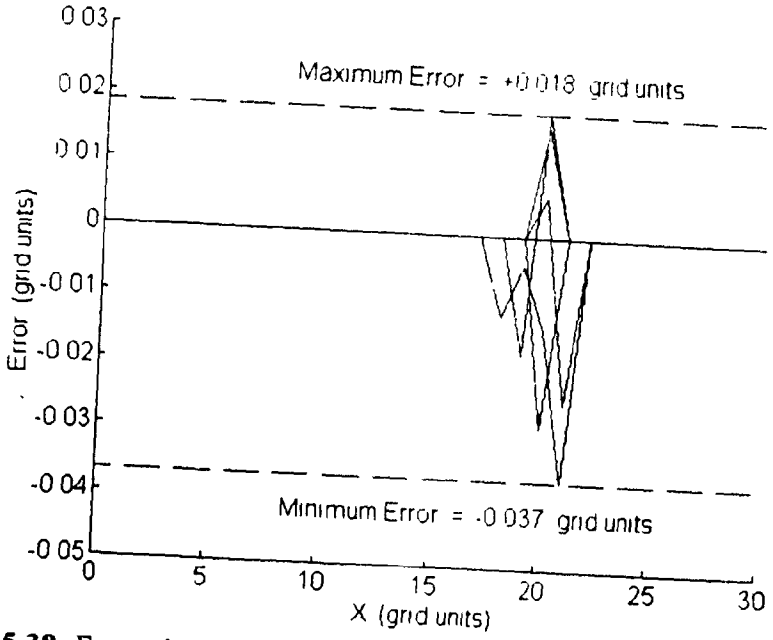


Figure 5.38: Error plot showing the difference between Figures 5 37 (substituted elevations) and 5 2 (original elevations)

Figure 5 39 shows the MLES corresponding to the parabolic half-trench surface with single localized point feature of Figure 5 10 Figure 5 40 compares the MLES of Figure 5.39 with the nominal data of Figure 5 4 The MLES of Figure 5 39 is seen to be within an error band of 0 114 grid units (0 31 mm, 0 012 in) relative to the nominal data of Figure 5.2, and the maximum MLES error in this example is also 0 1 '4 grid units (0.31 mm, 0.012 in) Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0 151 grid units (± 0.41 mm, ± 0.016 in) of the nominal data

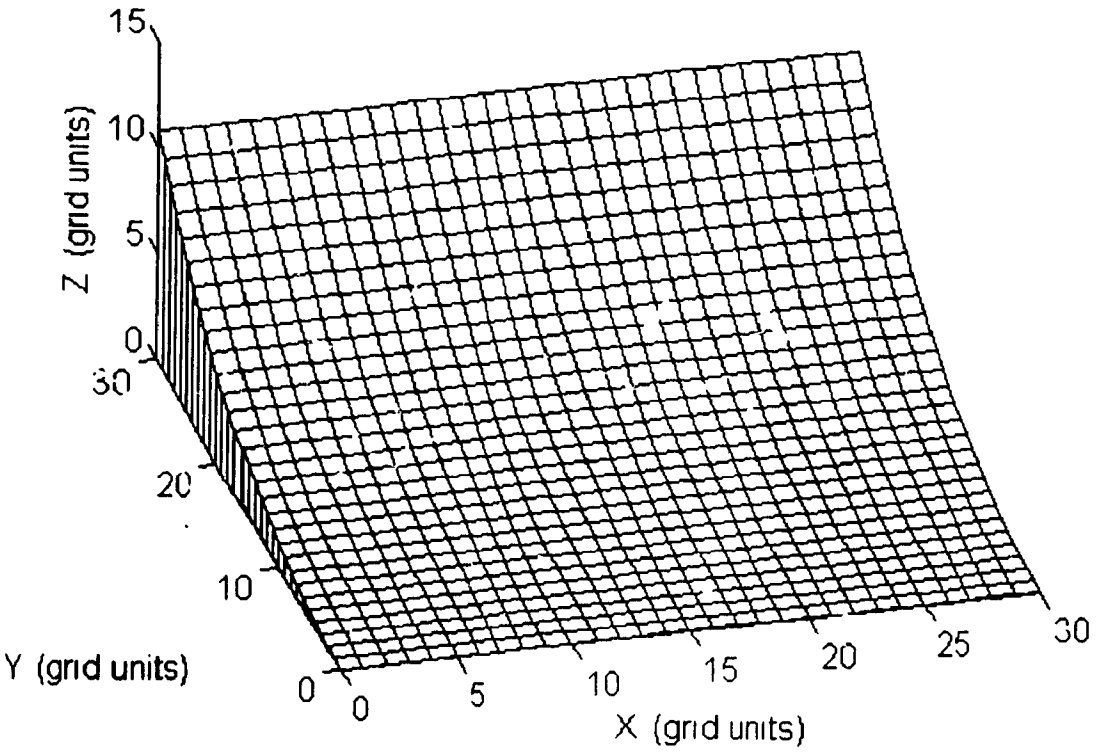


Figure 5.39: MLES corresponding to the surface of Figure 5.10.

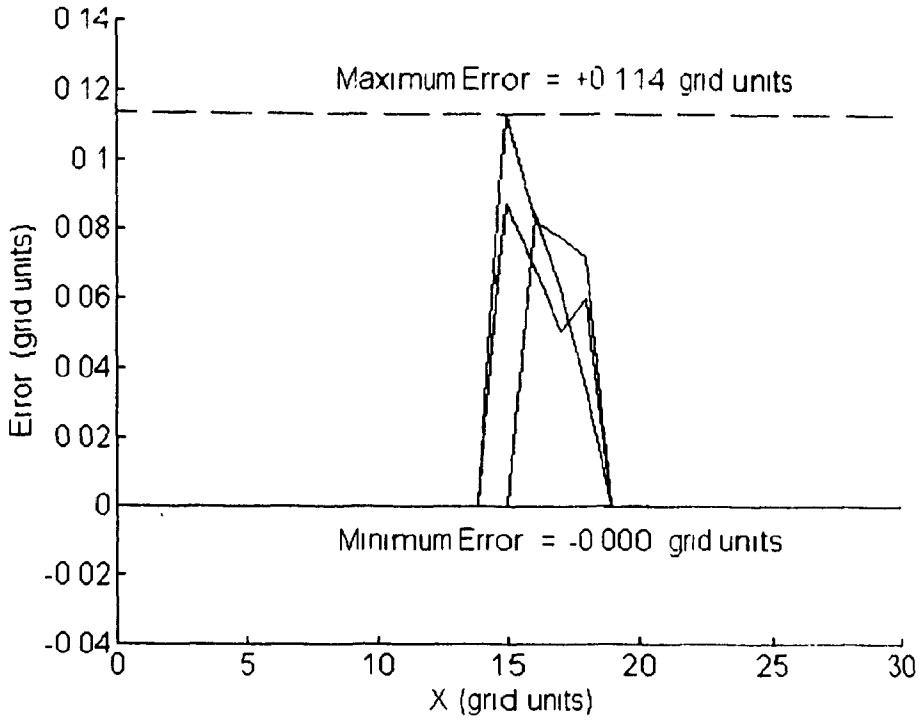


Figure 5.40: Error plot showing the difference between Figures 5 39 (substituted elevations) and 5.4 (original elevations)

Figure 5.41 shows the MLES corresponding to the biquartic saddle surface with single localized point feature of Figure 5.13 Figure 5 42 compares the MLES of Figure 5 41 with the nominal data of Figure 5 6 The MLES of Figure 5 41 is seen to be within an error band of 0.476 grid units (1 29 mm; 0.0507 in) relative to the nominal data of Figure 5 6, and the maximum MLES error in this example is 0 264 grid units (0 72 mm, 0 028 in) Corrective machining using the Yamaha Zeta-1 workcell may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0 301 grid units (± 0.82 mm, ± 0.032 in) of the nominal data

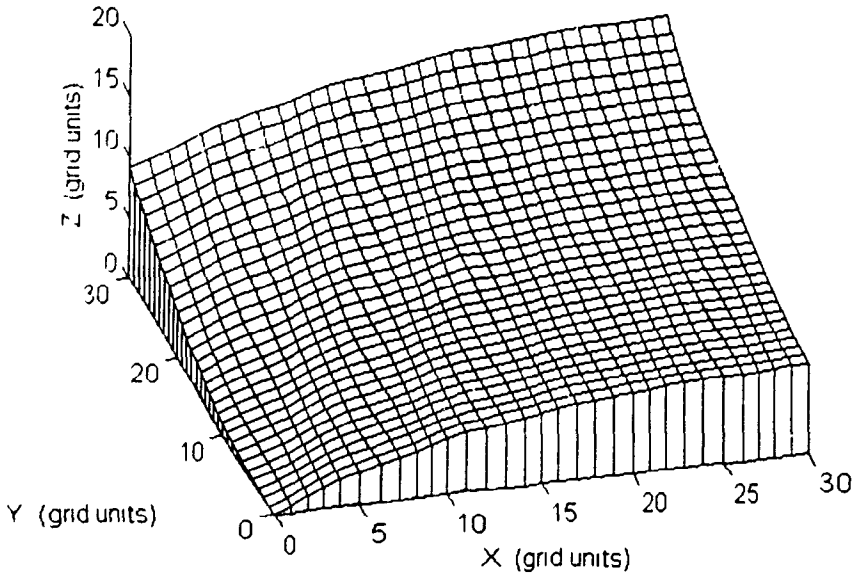


Figure 5.41: MLES corresponding to the surface of Figure 5 13

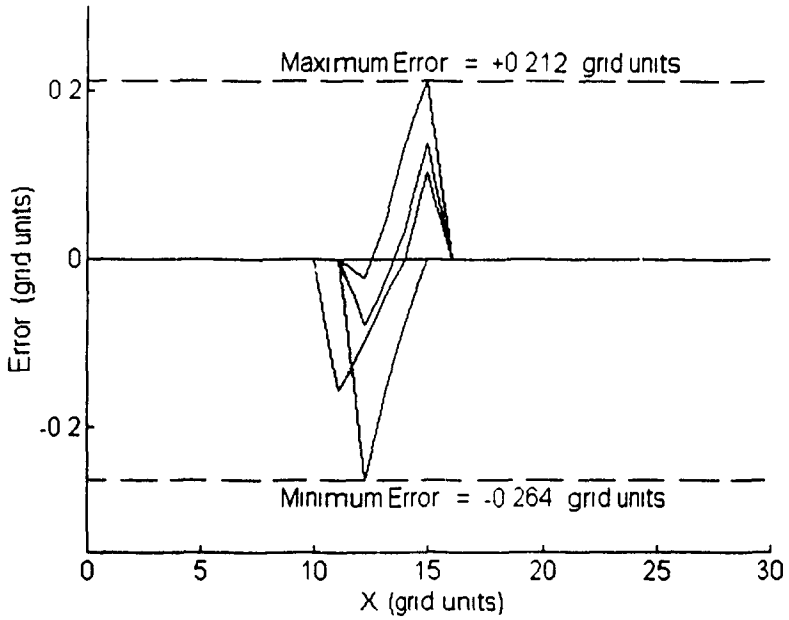


Figure 5.42: Error plot showing the difference between Figures 5.41 (substituted elevations) and 5.6 (original elevations).

In the examples of this section, the maximum error between MLES and nominal surfaces is 0.264 grid units (0 72 mm, 0 028 in) The maximum error expected between reconditioned and nominal surfaces using the Yamaha Zeta-1 workcell is ± 0 301 grid units (± 0 82 mm; ± 0.032 in) These errors are significantly larger than the repeatability of the workcell (± 0 037 grid units; ± 0 10 mm, ± 0 0039 in) Further refinements in MLES determination are therefore of interest. These results may be improved by a variety of means, including increasing the number of iterations carried out by the surface fitting algorithm (effects of increased iteration counts will be discussed in Section 5 4.5), or by using a smaller digitizing stepsize.

5.4.2 Multiple Localized Point Features

Figure 5 43 shows the MLES corresponding to the inclined plane surface with multiple localized points features of Figure 5 16. Figure 5.44 compares the MLES of Figure 5 43 with the nominal data of Figure 5 2. The MLES of Figure 5 43 is seen to be within an error band of 0.069 grid units (0 19 mm, 0 0075 in) relative to the nominal data of Figure 5 2, and the maximum MLES error in this example is 0 060 grid units (0 16 mm, 0 0063 in) Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0.111 grid units (± 0 30 mm, ± 0 012 in) of the nominal data.

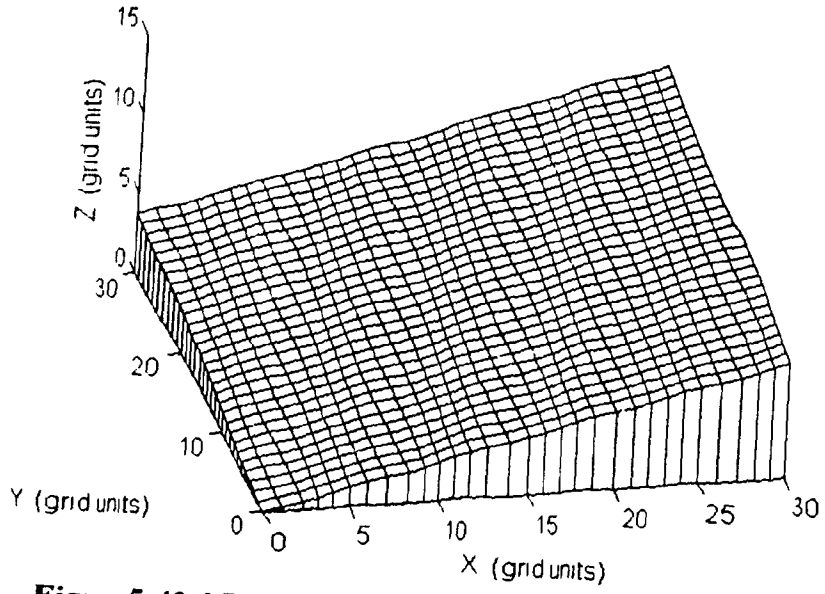


Figure 5.43: MLES corresponding to the surface of Figure 5 16

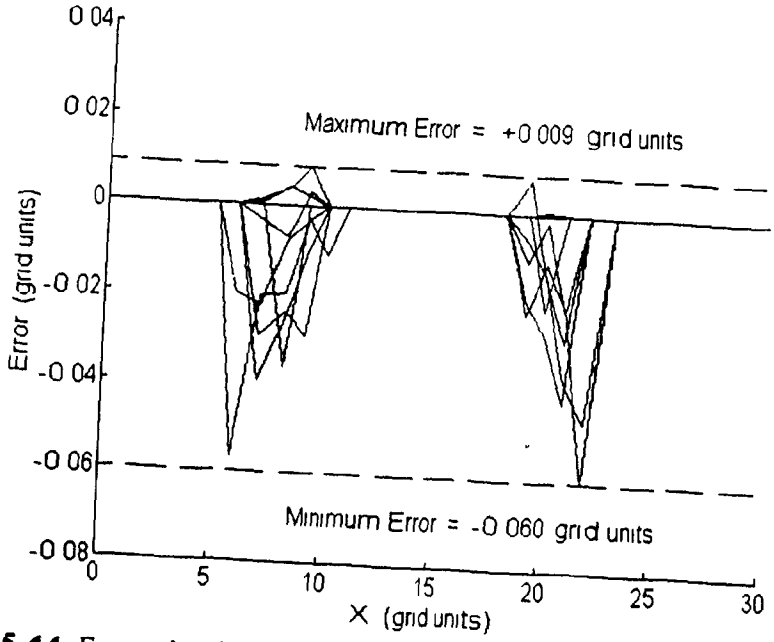


Figure 5.44: Error plot showing the difference between Figures 5.43 (substituted elevations) and 5 2 (original elevations).

Figure 5.45 shows the MLES corresponding to the parabolic half-trench surface with multiple localized point features of Figure 5 19 Figure 5 46 compares the MLES of Figure 5 45 with the nominal data of Figure 5 4 The MLES of Figure 5 45 is seen to be within an error band of 0.179 grid units (0 49 mm, 0 019 in) relative to the nominal data of Figure 5 4, and the maximum MLES error in this example is 0 163 grid units (0 44 mm, 0 017 in) Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0 200 grid units (± 0 54 mm, ± 0 021 in) of the nominal data.

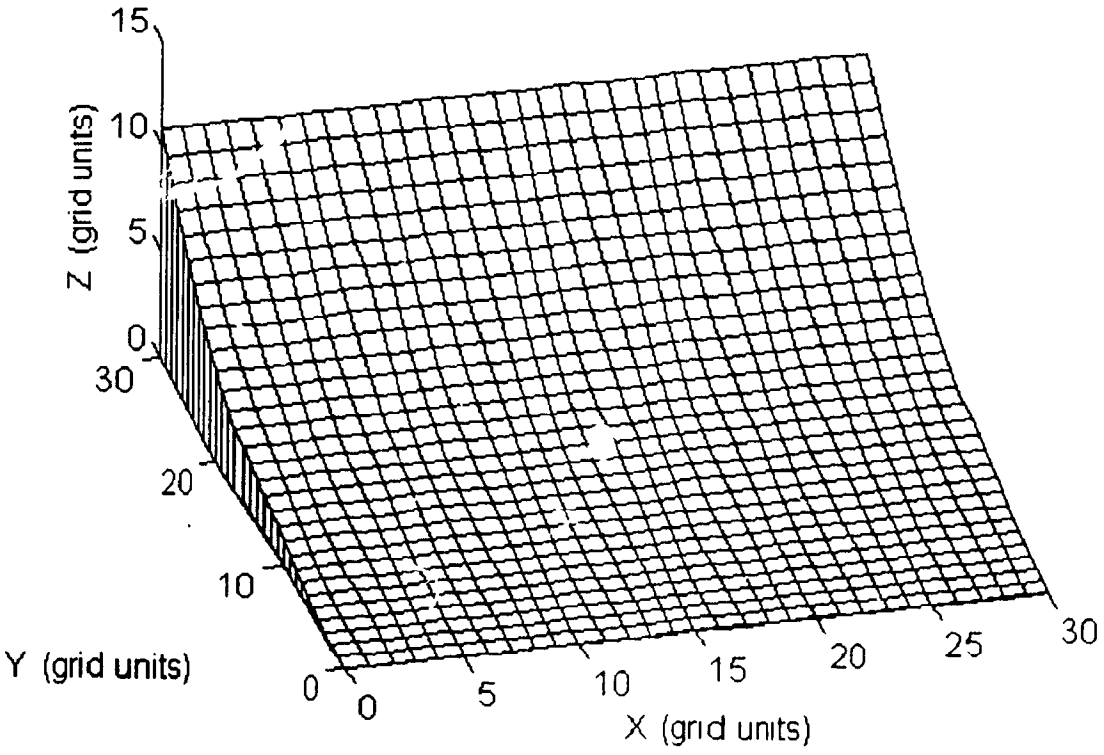


Figure 5.45: MLES corresponding to the surface of Figure 5 19

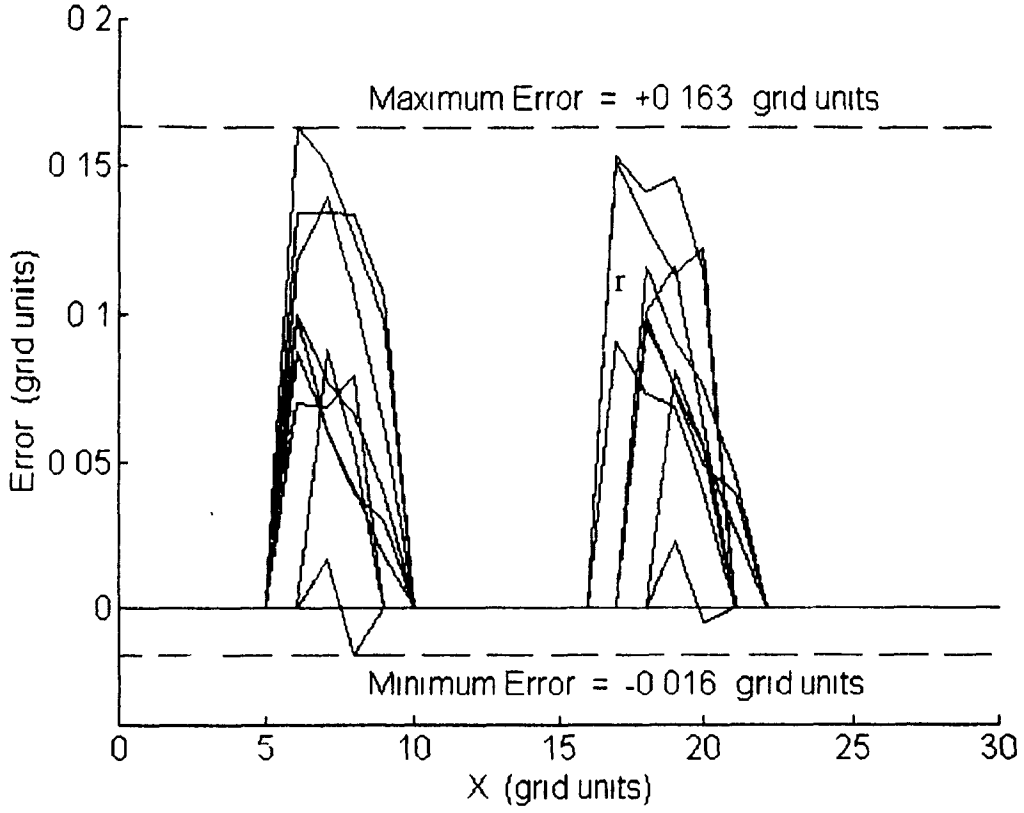


Figure 5.46: Error plot showing the difference between Figures 5 45 (substituted elevations) and 5.4 (original elevations)

Figure 5 47 shows the MLES corresponding to the biquartic saddle surface with multiple localized point features of Figure 5 22 Figure 5 48 compares the MLES of Figure 5 47 with the nominal data of Figure 5 6. The MLES of Figure 5 47 is seen to be within an error band of 0.414 grid units (1.12 mm; 0.0440 in) relative to the nominal data of Figure 5.6, and the maximum MLES error in this example is 0 262 grid units (0 71 mm; 0.028 in) Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0 299 grid units (± 0 81 mm, ± 0.032 in) of the nominal data.

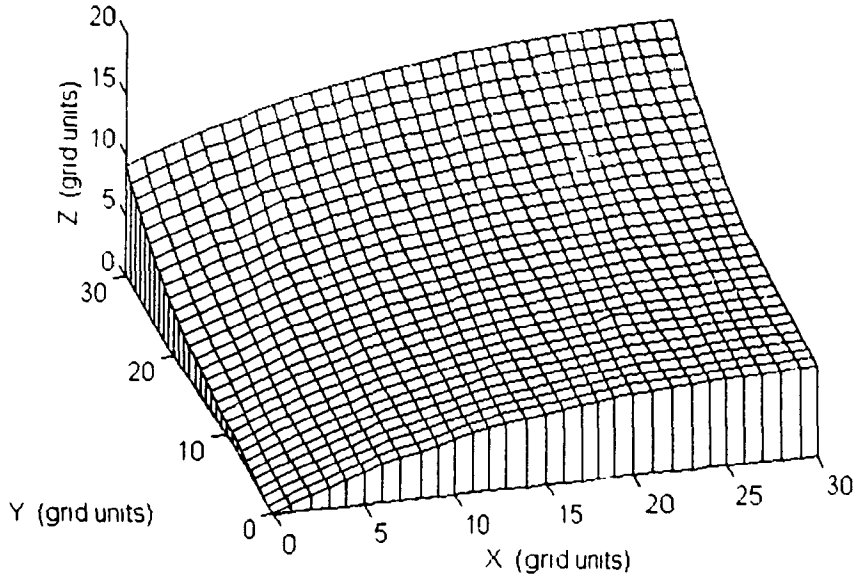


Figure 5.47: MLES corresponding to the surface of Figure 5 22

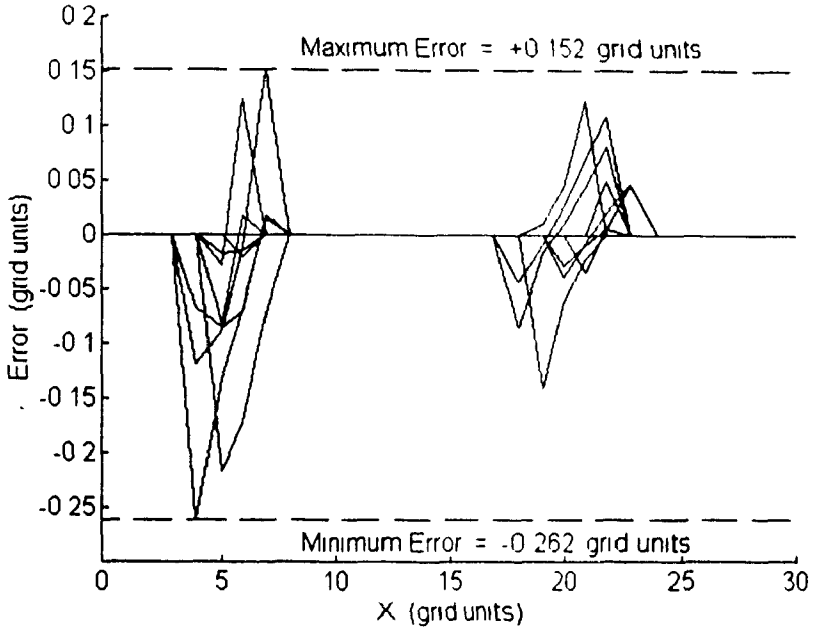


Figure 5.48: Error plot showing the difference between Figures 5 47 (substituted elevations) and 5 6 (original elevations)

In the examples of this section, the maximum error between MLES and nominal surfaces is 0 262 grid units (0 71 mm, O 028 in) The maximum error expected between reconditioned and nominal surfaces using the Yamaha Zeta-1 workcell is ± 0 299 grid units (± 0 81 mm, ± 0 032 in) These errors are significantly larger than the repeatability of the machine (± 0 037 grid units, ± 0 10 mm, ± 0 0039 in) Further refinements in MLES determination are therefore of interest These results may be improved using techniques outlined in Section 5 4 1

5.4.3 Linear Features

Figure 5 49 shows the MLES corresponding to the inclined plane surface with linear feature of Figure 5 25 Figure 5 50 compares the MLES of Figure 5 49 with the nominal data of Figure 5 2. The MLES of Figure 5 49 is seen to be within an error band of 0 1 18 grid units (0.32 mm, 0 013 in) relative to the nominal data of Figure 5 2, and the maximum MLES error in this example is 0 088 grid units (0.24 mm, 0 0094 in) Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0.125 grid units (± 0 34 mm, ± 0 013 in) of the nominal data.

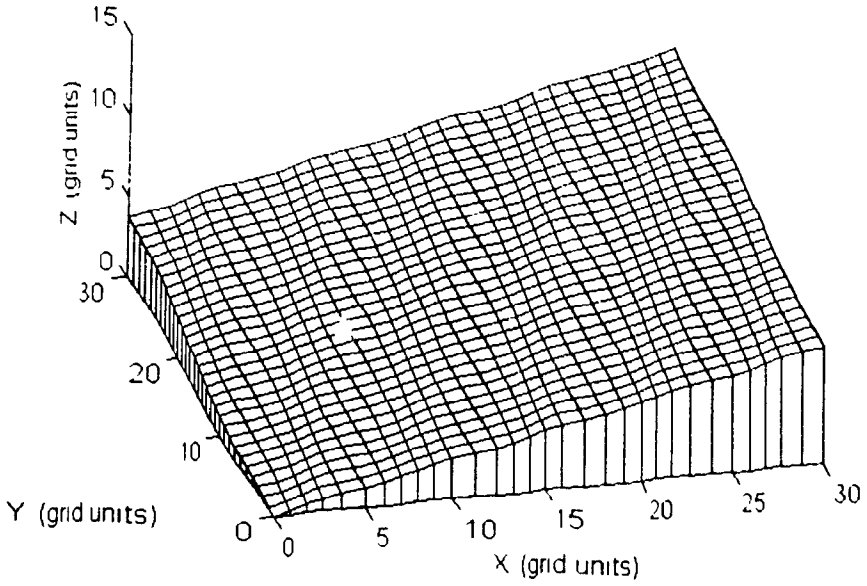


Figure 5.49: MLES corresponding to the surface of Figure 5 25

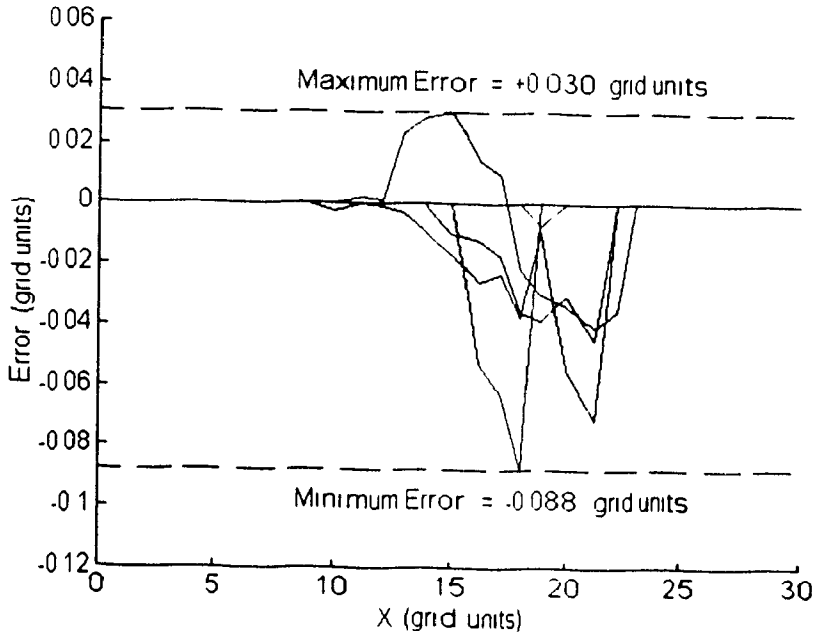


Figure 5.50: Error plot showing the difference between Figures 5 49 (substituted elevations) and 5 2 (original elevations)

Figure 5 51 shows the MLES corresponding to the parabolic half-trench surface with linear feature of Figure 5 28 Figure 5 52 compares the MLES of Figure 5.51 with the nominal data of Figure 5 4 The MLES of Figure 5 51 is seen to be within an error band of 0 282 grid units (0 76 mm, 0 030 in) relative to the nominal data of Figure 5 4, and the maximum MLES error in this example is 0 245 grid units (0 66 mm, 0 026 in). Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0 282 grid units (± 0.76 mm, ± 0.030 in) of the nominal data

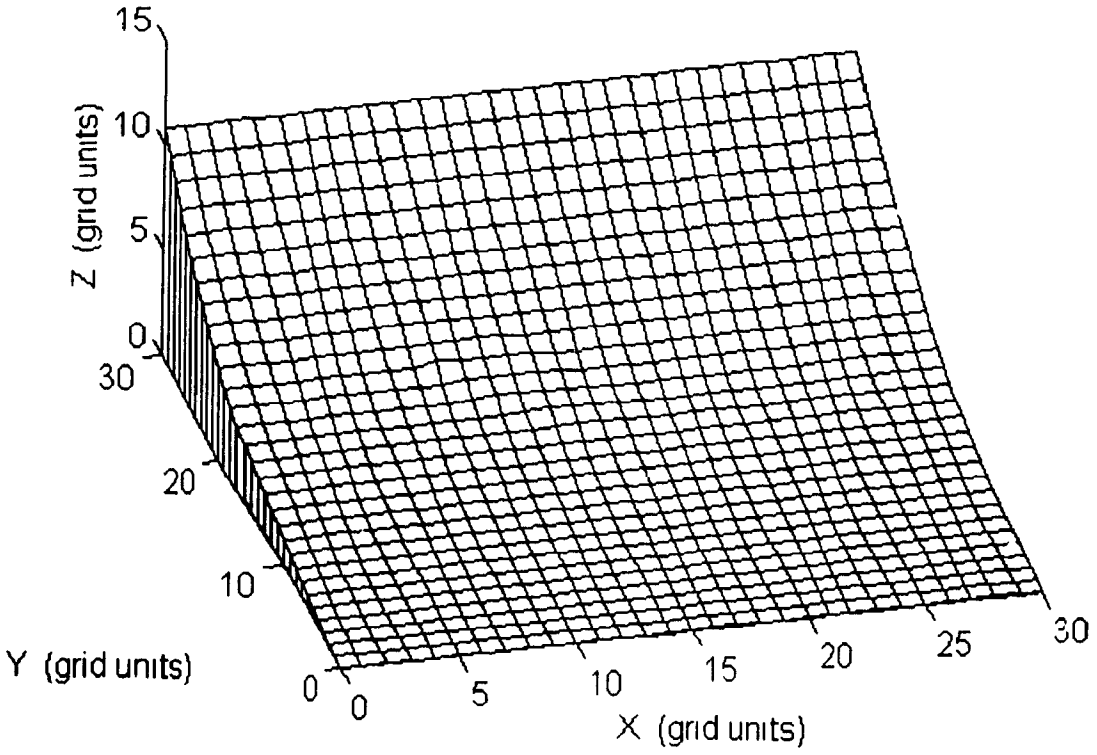


Figure 5.51: MLES corresponding to the surface of Figure 5.28.

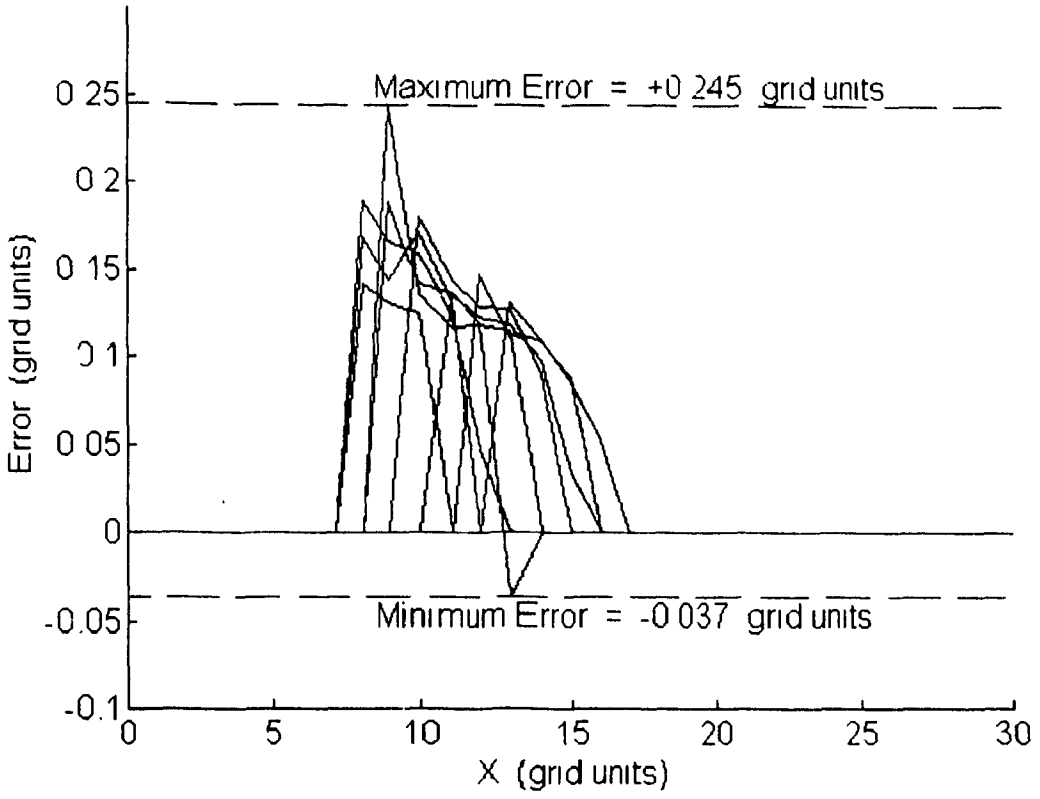


Figure 5.52: Error plot showing the difference between Figures 5 51 (substituted elevations) and 5 4 (original elevations)

Figure 5 53 shows the MLES corresponding to the biquartic saddle surface with linear feature of Figure 5 31 Figure 5.54 compares the MLES of Figure 5 53 with the nominal data of Figure 5.6. The MLES of Figure 5 53 is seen to be within an error band of 0 091 grid units (0.25 mm; 0.0098 in) relative to the nominal data of Figure 5 6, and the maximum MLES error in this example is 0.085 grid units (0 23 mm, 0 0090 in) Corrective machining may therefore be expected to yield a surface that falls within ± (maximum MLES error + machine repeatability) = ± 0.122 grid units (± 0.33 mm, ± 0 013 in) of the nominal data.

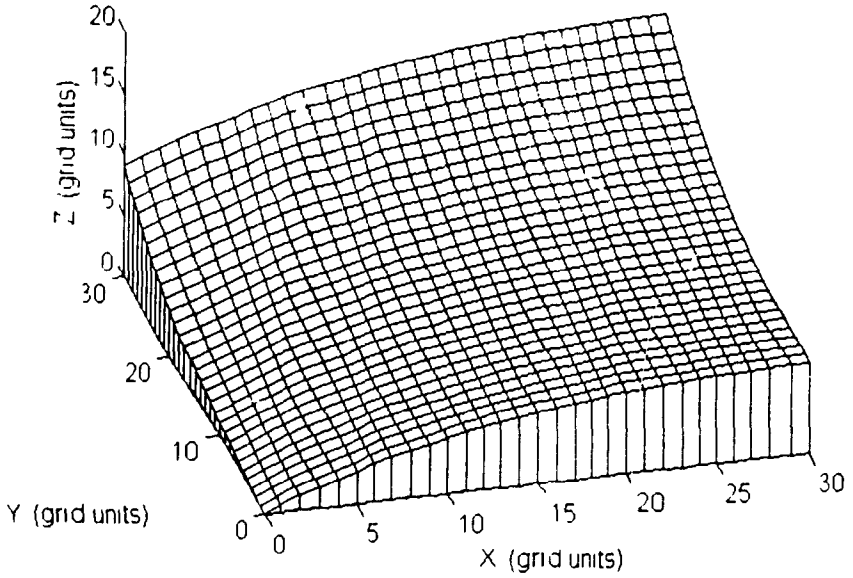


Figure 5.53: MLES corresponding to the surface of Figure 5 31

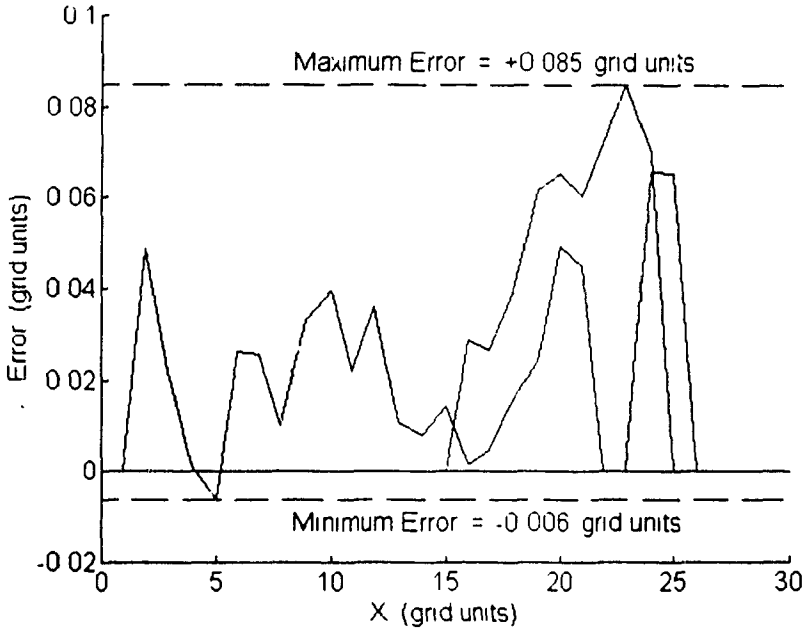


Figure 5.54: Error plot showing the difference between Figures 5 53 (substituted elevations) and 5 6 (original elevations)

In the examples of this section, the maximum error between MLES and nominal surfaces is ± 0 245 grid units (± 0 66 mm, ± 0 026 in) The maximum error expected between reconditioned and nominal surfaces using the Yamaha Zeta-1 workcell is ± 0 282 grid units (± 0 76 mm, ± 0 030 in) These errors are significantly larger than the repeatability of the machine (± 0 037 grid units, or ± 0 10 mm, ± 0 0039 in) Further refinements in MLES determination are therefore of interest These results may be improved using techniques outlined in Section 5 4 1

5.4.4 Biquartic Saddle with an S-Shaped Feature

Figure 5 55 shows the MLES corresponding to the biquartic saddle surface with S- shaped feature of Figure 5 34 Figure 5 56 compares the MLES of Figure 5 55 with the nominal data of Figure 5 6. The MLES of Figure 5 55 is seen to be within an error band of 0 284 grid units (0 77 mm; 0 030 in) relative to the nominal data of Figure 5 6, and the maximum MLES error in this example is 0 152 grid units (0 41 mm, 0 016 in) Corrective machining may therefore be expected to yield a surface that falls within + (maximum MLES error + machine repeatability) = ± 0 189 grid units (± 0 51 mm, ± 0 020 in) of the nominal data

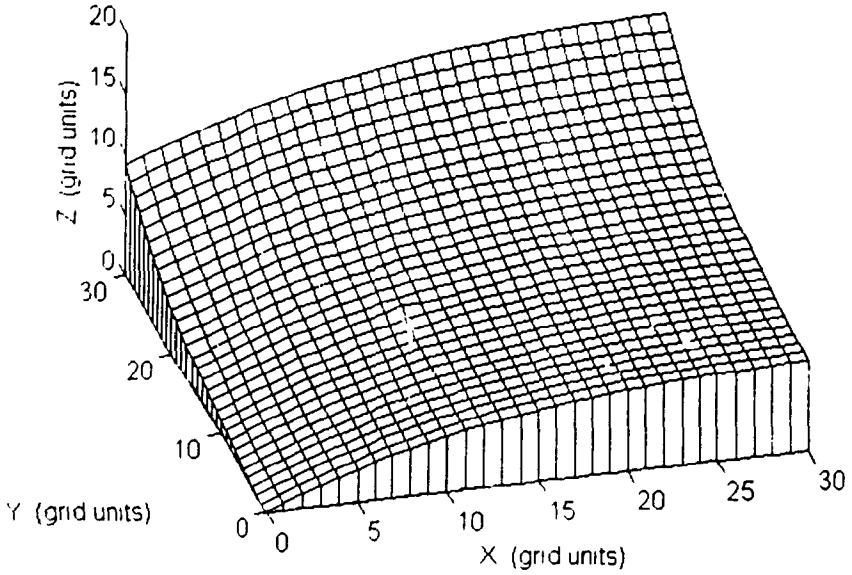


Figure 5.55: MLES corresponding to the surface of Figure 5 34

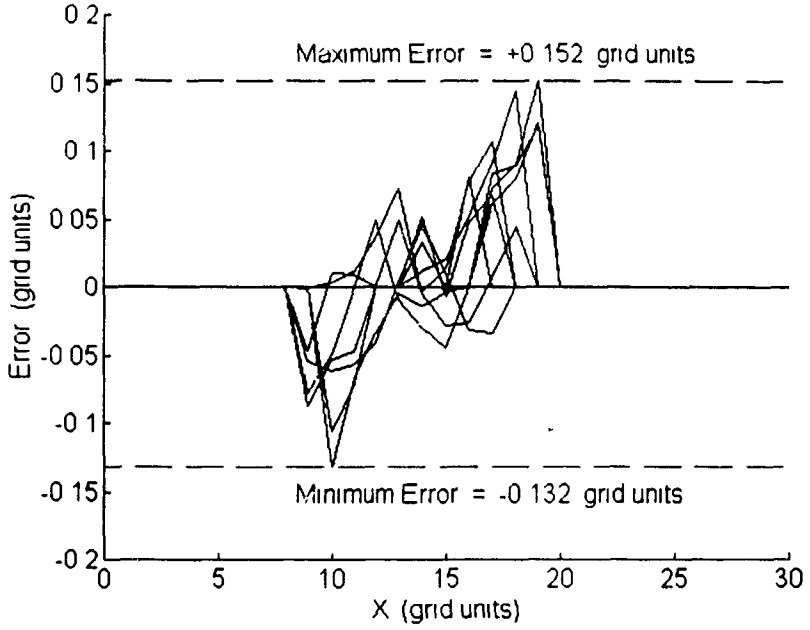


Figure 5.56: Error plot showing the difference between Figures 5.55 (substituted elevations) and 5 6 (original elevations).

In the example of this section, the maximun error between MLES and nominal surfaces is 0 152 grid units (0 41 mm, 0 016 in) The maximum error expected between reconditioned and nominal surfaces using the Yamaha Zeta-1 workcell is ± 0 189 grid units (± 0 51 mm, ± 0 020 in) These errors are significantly larger than the repeatability of the machine (± 0 037 grid units, ± 0 10 mm, ± 0 0039 in) Further refinements in MLES determination are therefore of interest These results may be improved using techniques outlined in Section 5 4 1

5.4.5 Observed Computation Times and Error Trends

Figure 5 57 shows a comparison of required computation times in seconds versus various degrees of polynomial fit using the algorithm of Chapter 4 Times are given for a 50 MHz INTEL 486DX personal computer, using the Turbo C++ version 3 0 programming language [20]. One centrally-located rejected point is computationally replaced on the biquartic saddle surface of Figure 5 6 using 1000 iterations and a local reference point dataset of 30 points The increase in computation time appears to be parabolic, which corresponds to the parabolic increase in number of polynomial coefficients with N in Eqn (4 1)

Figure 5.58 shows a plot of observed error variation with number of points in the reference disk, for $2 \leq N \leq 5$ in Eqn (4 1), and for 2000 iterations of the optimization algorithm. Errors plotted in this figure are the absolute value of the difference between MLES and original undamaged digitized elevations This data is computed for the biquartic saddle, with a single rejected point specified in the centre of the feature map In a typical application of the present method, a plot such as Figure 5 58 would not be possible because the

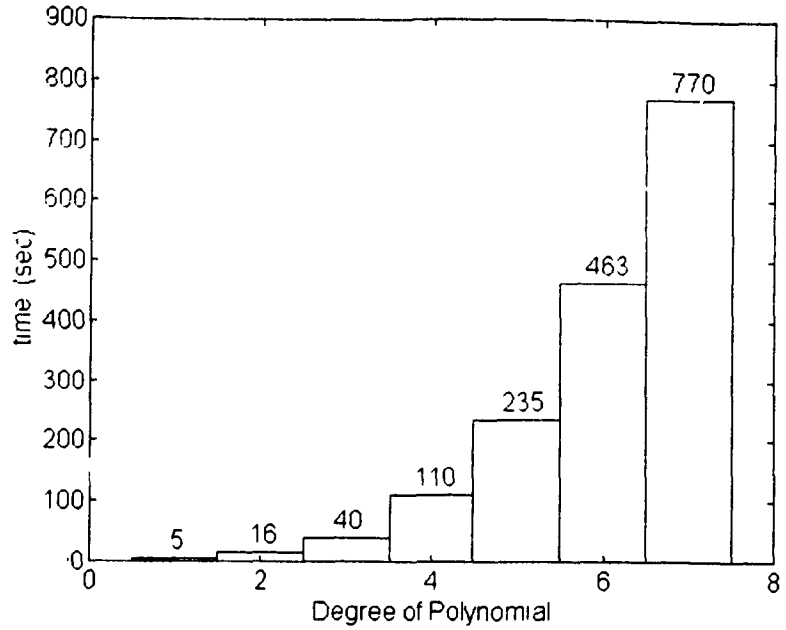


Figure 5.57: Required computation times for various degrees of polynomial surface fit

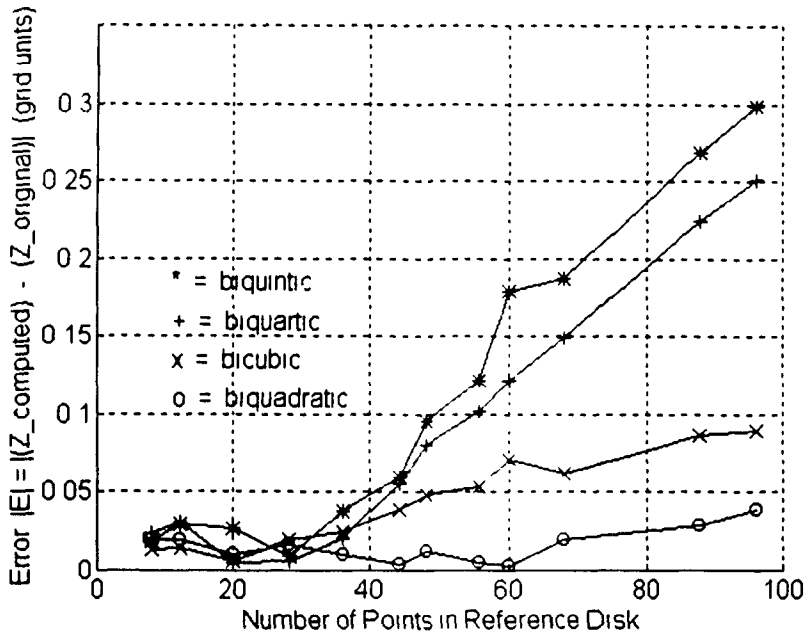


Figure 5.58: Observed error versus reference disk size for various degrees of polynomial surface fit

the component may have undergone slight arbitrary warping deformations, rendering comparison with original design data void The present examples allow error comparisons because the surfaces are known to have no warping deformations

There are two general trends evident in Figure 5 58 First, it can be seen that each type of polynomial fit suffers a decrease in accuracy as reference disk size is increased beyond a certain point This illustrates inability of the optimization algorithm to perform a global fit the contribution of local points becomes increasingly overwhelmed by ever larger numbers of distant points The second trend is an apparently decreasing accuracy for higher degree fits This may be the effect of slower convergence, since higher degree polynomials have more coefficients to optimize The irregularity of these curves reflects the discrete nature of the data, and the existence of sampling errors Note that plotted data points in Figure 5 58 are not evenly spaced along the horizontal axis This is because the number of points in a reference disk varies in an irregularly-stepped manner with increasing radius, as shown in Figure 4 1

Figure 5 59 shows the variation of average error with iteration count for various degrees of surface fitting polynomial The measure of error plotted here is the average of absolute value of error (*not* squared error), taken over a reference disk of 30 points The feature to be recomputed is the same as for Figure 5 58 a single centrally-located point on the biquartic saddle surface A 30-point reference disk has been chosen for this plot so it would correspond directly to the computations done for the digitized examples of this chapter. Average error is seen to decrease smoothly with increasing iteration count, from approximately 0 75 grid units at 200 iterations to 0 25 grid units at 2000 iterations There are

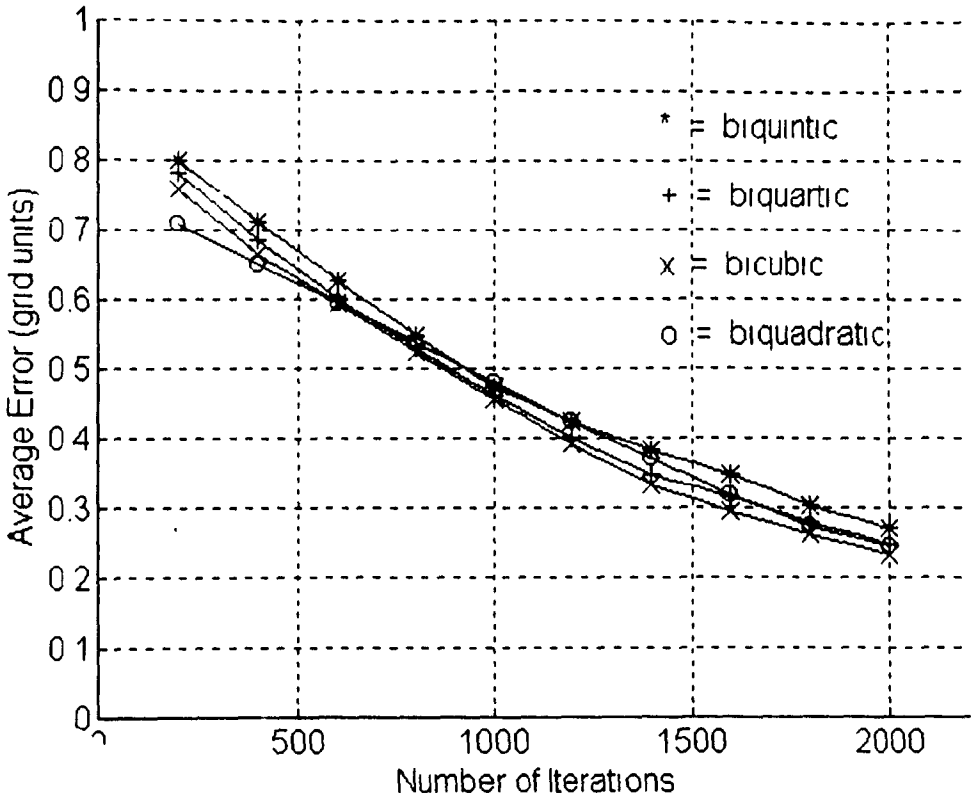


Figure 5.59: Observed average error versus iteration count for various degrees of polynomial surface fit, for a constant reference disk size of 30 points

only small differences in accuracy resulting from use of various degrees of surface-fitting polynomial average errors are within 0 1 grid units of one another over the entire range of iterations plotted in Figure 5 59

5.5 Summary

This chapter has presented a numerical verification of the methodologies developed in Chapters 2 through 4 of this thesis, using ten case studies Tables 5.1, 5.2 and 5.3 summarize results of these case studies. Error bands in these tables are defined as the difference between

maximum and minimum errors as given by the error plots (Figures 5 38, 5 40, 5 42, 5 44, 5 46, 5 48, 5 50, 5 52, 5 54, 5 56) of this chapter For example, the error plot of Figure 5 56 has a maximum error of 0 152 grid units and a minimum error of -0 132 grid units, therefore the error band is 0 152 - (-0 132) = 0 284 grid units

Tables 5 1, 5 2 and 5 3 shows an apparently weak correlation between feature size and error band Stronger correlations likely exist between the error band and a combined measure of feature size and feature aspect ratio To illustrate this, consider the linear features of Figures 5 28 and 5 31 The feature of Figure 5 31 is narrower than that of Figure 5 28 (i e , has higher aspect ratio) Note that the error band of Figure 5 54 (corresponding to differences between the MLES of Figure 5 53 and the nominal geometry of Figure 5 6) is smaller than that of Figure 5 52 (corresponding to differences between the MLES of Figure 5 51 and the nominal geometry of Figure 5 4) Higher aspect ratio may thus be taken as an indication that there are more local nonrejected points available within reference point disks of a given radius

Computation times and error trends have been plotted for the surface reconditioning methods described in Chapter 4. Ratios between the computational times shown in Figure 5 57 provide an indication of similar ratios for other types of computers Required computation times are seen to increase in an approximately parabolic manner This parabolic increase reflects the number of polynomial coefficients to be determined by optimization, which also increases proportional to the square of the degree of polynomial surface fit required

Error trends plotted for the surface reconditioning algorithm show a high dependence

Table 5.1: Summary of Chapter 5 Case Studies - Inclined Plane Surface				
Feature Type	*Total # of Grid Points*	*MLES Error Band* (grid units, mm, in)	*Maximum MLES Error* (grid units; mm, in)	*Predicted Maximum Corrective Machining Error* (grid units, mm, in)
Single Localized Lump	11	0 055, 0 15; 0 0059	0 037, 0 10, 0.0039	0 074; 0 20; 0 0079
Multiple Lumps	44	0 069, 0 19, 0 0075	0 060, 0 16; 0 0063	0 111, 0 30, 0 012
Linear	35	0 118; 0 32, 0 013	0 088, 0 24, 0.0094	0.125, 0 34, 0 013

Table 5.2: Summary of Chapter 5 Case Studies - Parabolic Half-Trench Surface.				
Feature Type	*Total # of Grid Points*	*MLES Error Band* (grid units; mm; in)	*Maximum MLES Error* (grid units, mm, in)	*Predicted Maximum Corrective Machining Error* (grid units, mm; in)
Single Localized Lump	11	0.114, 0 31; 0.012	0.114, 0 31, 0.012	0.151; 0 41; 0.016
Multiple Lumps	62	0.179; 0.49; 0.019	0 163, 0 44, 0.017	0 200; 0 54; 0.021
Linear	40	0 282, 0 76, 0 030	0.245, 0.66, 0.026	0 282; 0.76; 0 030

Table 5.3: Summary of Chapter 5 Case Studies - Biquartic Saddle Surface				
Feature Type	*Total # of Grid Points*	*MLES Error Band* (grid units, mm, in)	*Maximum MLES Error* (grid units, mm, in)	*Predicted Maximum Corrective Machining Error* (grid units, mm, in)
Single Localized Lump	16	0 476, 1.29, 0 051	0 264, 0 72, 0 028	0 301, 0 82, 0 032
Multiple Lumps	51	0 414; 1 12, 0 044	0 262, 0 71, 0 028	0 299, 0 81, 0 032
Linear	31	0 091, 0.25, 0 0098	0 085, 0.23; 0.0090	0 122, 0 33, 0 013
"S"-Shaped	65	0 284, 0.77, 0 030	0 152, 0 41, 0.016	0 189, 0.51, 0.020

of error on degree of polynomial surface fit and reference disk size for a given number of optimization iterations (Figure 5 58) Figure 5 59 shows that there is little difference in error for various degrees of polynomial surface fit at a given reference disk size as the iteration count is increased Overall, these plots indicate that the biquadratic surface fit performs well, both from the standpoint of required computational time (Figure 5 57) and surface fitting error (Figures 5.58 and 5.59).

CHAPTER 6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

A robotic deburring workcell is found to be suitable for the automation of digitizing, feature recognition and corrective machining operations as required by the reconditioning methodology presented in this thesis Types of features which are recognized by this process include lumps, cracks, pitting, scratches, and the like Work described in this thesis has addressed the following

- Development of a scheme to perform autonomous recognition of defective features on the surface of machined workpieces
- Development of a technique to perform computational replacement of rejected surface features as a prior step to corrective machining
- Development of an automated digitizing capability for the Yamaha Zeta-1 deburring robot
- Development of algorithms and demonstration of the techniques in the preceding steps using digitized data collected by the Yamaha Zeta-1 deburring robot.

Feature recognition has been performed without comparison to original design geometry, in a way that mimics on digital computer the ability in human intelligence for understanding the *concept* of a flaw A surface fitting scheme has been developed using multivariate

optimization to interpolate geometric trends in nonrejected digitized elevations, thus establishing a desired reconditioned geometry for damaged components This thesis presents the addition of two capabilities for the Yamaha Zeta-1 deburring workcell, namely sample-on-request for offline digitizing path generation, and automated digitizing

6.1.1 Autonomous Feature Recognition

An algorithm for autonomous feature recognition has been developed and successfully demonstrated for digitized machined surfaces A two-stage approach to recognition involving segmentation followed by fuzzy logic-based classification has been found to give excellent results, as shown in the case studies of Chapter 5, Section 5 3 All grid points comprising extraneous surface features in these case studies have been correctly identified by the method

Fuzzy logic provides an inferential and quantitative framework for decision making in the absence of crisp rules As such, it is seen in the present work as an effective reasoning process for assessing the condition of a surface There is no single geometric criterion which can reliably serve to identify extraneous surface features on a component surface Use of an amalgamated index formed from multiple criteria using fuzzy reasoning, however, is seen to yield a reliable and robust classifier Applying segmentation prior to fuzzy classification has the benefit of enhancing classifier reliability The classifier is thus allowed to consider only aggregate properties of a few appropriately selected regions, rather than having to deal with an entire surface point by point

The current approach is designed for surfaces containing damaged regions which are small in comparison to the total surface area It is possible that features which are smaller than

the digitizing stepsize or that blend in smoothly with their surroundings may escape detection These occurrences can be minimized by proper selection of digitizing stepsize and / or proper tuning of the recognition algorithm A digitizing stepsize can therefore be selected as equal to the maximum allowable size of extraneous surface feature Tuning of the recognition algorithm consists of setting the angular tolerance ϵ_0 shown in Figure 3 3, and setting the classifier threshold value x_{crit} shown in Figure 3 6 If ϵ_0 is set too large, fewer regions will be identified with segmentation, leading to a coarser feature map Similarly, if x_{crit} is set too large, fewer extraneous features will be recognized / rejected Recognition results for the examples considered are found to be excellent all extraneous features shown in Chapter 5 have been completely identified

6.1.2 Automated Surface Reconditioning

A method has been developed and successfully demonstrated for determination of the desired geometry to which a damaged component should be machined The approach presented can be readily applied to surfaces whose original design data is no longer an adequate model for the component, due (for example) to the occurrence of warping deformations A multivariate optimization scheme has been chosen to perform surface fitting for replacement of rejected digitized grid elevations. Use of the splining algorithm of Temple-Raston *et al.* [63] has been detailed for generation of a high-resolution corrective machining toolpath over the Maximum Likelihood Eventual Surface (MLES) grid resulting from surface fitting In the present research, the CIC robotic deburring workcell has successfully demonstrated tracing of surfaces using high-resolution toolpaths resulting from this splining technique.

Integration of the digitizing, recognition, and reconditioning algorithms of this research results in a complete methodology for automated refurbishment of complex precision-machined components This technology holds particular promise in applications to more expensive components such as gas turbine rotor blades and control surfaces in fluid power systems The case studies of Chapter 5 illustrate results of surface reconditioning carried out on a variety of digitized surfaces whose geometry is representative of potential applications. Comparison with original geometry (as summarized in Tables 5 1, 5 2 and 5 3) reveals the existence of errors due to the reconditioning algorithm

Table 5 1 summarizes results of computational grid point replacement for an inclined plane surface In this table, it is seen that predicted maximum corrective machining error varies from 0 074 grid units (0 20 mm, 0 0079 in) to 0 125 grid units (0 34 mm, 0 013 in). Table 5 2 summarizes results of computational grid point replacement for a parabolic half-trench surface In this table, it is seen that predicted maximum corrective machining error varies from 0 151 grid units (0.41 mm, 0 016 in) to 0 282 grid units (0 76 mm, 0 030 in) Table 5 3 summarizes results of computational grid point replacement for a biquartic saddle surface. In this table, it is seen that predicted maximum corrective machining error varies from 0.122 grid units (0.33 mm, 0.013 in) to 0 301 grid units (0 82 mm, 0 032 in)

The average predicted maximum corrective machining error in the ten case studies of Chapter 5 is 0 185 grid units (0.50 mm, 0 079 in) Acceptability of such an error depends on what machining / refurbishing surface finish tolerances are actually specified in any given situation If such tolerances demand better results, there are two possible solutions use of a higher number of iterations for surface fitting, and / or use of a smaller digitizing stepsize

Further refinements of this algorithm, to be discussed as future work in Section 6 2 2, are also expected to yield improved performance

6.1.3 Digitizing Strategies

Two digitizing strategies have been developed for use with the robotic deburring workcell at Concordia University's Centre for Industrial Control These strategies are distinguished by their respective methods for path generation, and are described in detail in Chapter 2 of this thesis One strategy is based on offline path generation, and the other on online (real-time) path generation Digitizing with offline path generation has been carried out in a series of steps involving analysis, reduction and transformation of original geometric data for the component Offline path generation is unsuitable for components which have been warped or otherwise damaged so as to differ significantly from original design data In the present work, digitizing with online path generation has been carried out using linear extrapolations to compute expected elevations for each successive digitizing point Linear extrapolation is considered adequate for the present work, since following too fine a prediction for successive elevations leads to reduction of the digitizing indexing clearance (Figure 2 2) This in turn creates a situation which favours probe crashes (against any extraneous / unanticipated surface features that may exist) during the indexing path segment of each digitizing motion cycle

Digitizing with both offline and online path generation has been successfully demonstrated on the CIC deburring workcell, but all results presented in this thesis are based exclusively on use of digitizing with online path generation. Online path generation has been

chosen in the present research due to the addition of extraneous features on the digitized component surfaces These features represent a significant deviation from original geometry, and hence render original geometric data unuseable as a basis for offline path generation

6.2 Future Work

6.2.1 Autonomous Feature Recognition

Possible areas of improvement for the autonomous feature recognition method developed in the present work include the following

(i) use of alternate geometric criteria for regional segmentation,

(ii) extension of the segementation and classification methods to nonuniform grids,

(iii) transputerization of recognition codes for to permit simultaneous digitizing and analysis,

(iv) use of additional geometric criteria for fuzzy classifier input variables,

(v) use of alternate fuzzy rule base structures, membership function profiles, and defuzzification strategies,

(vi) automatic tuning of the recognition method

It should be possible to improve the performance of segmentation - the first of two stages in the autonomous feature recognition algorithm described in the present research - in a variety of ways For example, the effectiveness of using alternative geometric criteria may be investigated. Such criteria may be derived from computed measures of curvature, local

variations in normal vector direction cosines, local variations in volume enclosed under the digitized surface, or local variations in elevations compared with extrapolated trends One possible benefit from use of alternate criteria may be the inclusion of grid edge points, which are not given proper regional assignation with the present scheme Another useful enhancement may result from generalization of the segmentation method to nonuniform grids An immediate consequence of this would be relaxation of the assumption that rectangular digitizing grids are *perfectly* rectangular This assumption is justified in the present work only because the digitizing stepsize is much greater than the repeatability of the workcell.

Transputerization of segmentation codes may also be used to improve both speed and maximum allowable grid sizes This type of operation must be preceded by a partitioning of the data into rectangular sub.egions to be analyzed in parallel Reassembly of the partitioned data is then required after completion of the analysis. A twofold or greater increase in computation speed may therefore be expected from transputerization Grid partitioning also provides a way around array size limitations imposed by computer memory Transputerization is more important for segmentation than classification, since computer memory requirements are significantly greater during segmentation (this is because segmentation must explicitly analyze all digitized grid elevations, whereas in classification only aggregate properties of appropriately-selected .egions are manipulated)

The second major area for further work in autonomous feature recognition is in fuzzy classifier design Identification of additional criteria for use in defining input variables could possibly lead to improved reliability in the classification algorithm More complex rule base structures could also be explored, for example utilizing intermediate fuzzy sets such as

Medium or even Medium-Low and Medium-High The effect of different input membership function curve shapes could also be investigated For example, bilinear, trilinear, parabolic, or sigmoidal input membership curves might have an impact on classifier performance Lee [39] cites three commonly used defuzzification strategies, the relative merits of which may also be compared for the present work

It may be possible to develop a scheme for automated determination of the tuning parameters used for segmentation and classification These tuning parameters are the half-range vertical angle tolerance ϵ_θ used in segmentation, and the threshold value x_{crit} used after classifier defuzzification. For the half-range vertical angle criterion, a tuning scheme could be based on analysis of the variation in total number of segmented regions versus ϵ_θ For the classifier defuzzification threshold value, tuning could be carried out based on analysis of the variation in total number of rejected regions versus x_{crit}

6.2.2 Automated Surface Reconditioning

Possible areas of improvement for the digitizing strategies developed in the present work include the following:

(i) use of alternate error index definitions,

(ii) use of alternate surface fitting functions,

(iii) experimental demonstrations of corrective machining.

As mentioned in Section 4 2 4, the error index of Eqn (4 4) is by no means the only

possible error index definition. An index based on absolute value of error may also give good results Another possibility is to incorporate a weighting function which reduces the importance of more distant reference points Conceivably, such a scheme could even dispense with the necessity of sizing a reference disk, since with sufficiently small weights attached to more distant points, the entire elevation grid can be included

Alternative forms of the surface fitting function given by Eqn (4 1) are also worth investigating For example, functions involving transcendental or complex terms may be effective In addition, it may happen in industry that the general form of surface function describing a deformed component surface is already known, and only the coefficients need to be determined A surface fitting scheme based on multivariate optimization such as that described by the present thesis can be adapted to work for all such cases. The predicted maximum corrective machining error for the case studies considered is 0.185 grid units (0 50 mm, 0 079 in) It is expected that the improvements outlined in this section can lead to errors as small as 0 54 grid units (0 20 mm, 0 008 in)

Experimental demonstration of corrective machining based on a toolpath generated by splining over the MLES would also be of interest Such a demonstration would require cutter compensation, and possibly multiple passes of the cutter over the component surface (initial heavier cuts followed by a final light finishing pass) For a more thorough test, it would be of interest to machine actual industrial components (such as the rotor assembly of Figure 1 1) in addition to test specimens of the type shown by Figures 5 1, 5.3, and 5.5, and to compare results with those due to manual grinding and polishing.

6.2.3 Digitizing Strategies

Possible areas of improvement for the digitizing strategies developed in the present work include the following

(i) use of higher order extrapolations,

(ii) development of alternate probe designs, including alternatives to contact probing, for improved access to components with intricate geometry,

(iii) use of digitizing strategies in which the probing axis is not maintained in a unidirectional orientation

More sophisticated extrapolation techniques can be implemented for digitizing with on-line path generation This would allow better prediction of successive digitizing elevations on the component surface Higher accuracy elevation predictions permit a shorter digitizing pathlength, and a consequent reduction in digitizing time requirements However, as pointed out in Section 6.1 3, it is not desireable to allow the probe to index too closely to the surface, since larger unanticipated extraneous surface features may interfere with and possibly damage the probe tip Development / use of alternate probe designs is essential for complete access to components with intricate geometry, such as the rotor blade assembly pictured in Figure 1 1. Alternatives to contact probing may also be investigated, including laser / photographic and electrical inductance techniques Digitizing strategies in which the probing axis is not maintained vertical at all times may also be an effective approach for certain components. For example, it may be advantageous to perform digitizing with the system holding the probing axis approximately normal to the component surface at all times

REFERENCES

[1] S Kalpakjian and K E McKee, *Automated Inspection: An Extended View*, Proceedings of the 8th International Conference on Automated Inspection and Product Control, IFS Publications Ltd , Bedford UK, pp. 13-23, 1987

[2] R J Cottman, *A Guidebook to ISO 9000 and ANSI ASQC Q90*, ASQC Quality Press, Milwaukee, Wisconsin, 1993

[3] C Suntag, *Inspection and Inspection Management*, ASQC Quality Press, Milwaukee, Wisconsin, pp 509-511, 1993

[4] G Farin, *Curves and Surfaces for Computer Aided Geometric Design - A Practical Guide*, Academic Press Inc , New York, pp. 219-233, 1988

[5] P Lancaster and K Salkauskas, *Curve and Surface Fitting - An Introduction*, Academic Press Inc , London, 1986

[6] F Tanaka, P Ikonomov, H. Okamoto and T Kishinami, *Inspection Methods for Geometrical Tolerances Using Coordinate Measurement Machines*, Proceedings of the 4th CIRP Seminar on Computer Aided Tolerancing, Paris, France, pp 325-336, 1995

[7] J R. Harris, S. C. Rockliffe, G Smith and T M Hill, *A Review of On-Line and Off-Line Programming Facilities for CMMs*, 4th International Conference on Factory 2000 - Advanced Factory Automation, pp 581-585, 1995.

[8] J Wauer, *Vibrations of Cracked Rotating Blades*, Rotordynamics '92: Proceedings of the International Conference on Rotating Machine Dynamics, Springer-Verlag,

New York, pp. 68-75, April 1992

[9] G E Thyer, *Computer Numerical Control of Machine Tools*, Industrial Press Inc , New York, pp 7-22, 1988

[10] M Lynch, *Computer Numerical Control for Machining*, McGraw-Hill Inc , New York, pp 289-321, 1992

[11] *Yamaha Zeta-1 Deburring Robot User's Manual, v. 1.2*, Yamaha Corporation, January 1990.

[12] *Operating Manual for Mazatrol M-32*, Yamakazi Mazak Corp , September 1987

[13] Ayyadevara, V R., *Development of an Automated Robotic Deburring Workcell*, M A. Sc Thesis, Department of Mechanical Engineering, Concordia University, Dec 1995.

[14] *IMS B008 User Guide and Reference Manual*, INMOS Corporation, Colorado Springs CO, January 1988

[15] R. Rajagopalan, R M H Cheng, V A Ayyadevara and G Huard, *System Architecture and Edge Tracking of an Automated Robotic Deburring Workcell*, Proceedings of the 1995 IEEE International Symposium on Intelligent Control, Monterey, California, pp. 351-356, August 1995

[16] *Instruction Manual: EG-225 Digital Linear Gauge*, Ono Sokki Co , Ltd , 1994

[17] S. C L. Poon and G. Huard, *Design of a General Purpose Interface Between a Serial Device and an 8-Bit Parallel Device*, CIC Internal Report #0028, Centre for Industrial Control, Concordia University, Montréal, November 1989

[18] *Parallel C User's Guide*, 3L Ltd , 1988

[19] B Kernighan and D Ritchie, *The C Programming Language*, Prentice-Hall, 1978.

[20] H Schildt, *Turbo C/C++: The Complete Reference - Second Edition*, McGraw-Hill, 1992

[21] E T Treywin and D B Edwards, *Automated Inspection and Control for Quality*, Proceedings of the 8th International Conference on Automated Inspection and Product Control, IFS Publications Ltd , Bedford UK, pp 121-144, 1987

[22] A J Spyridi and A A G Requicha, *Automatic Planning for Dimensional Inspection*, Manufacturing Review, vol 6, iss 4, pp. 314-319, December 1993.

[23] *Scanning Probe Upgrades CMM*, American Machinist, Penton Publishing Inc , Cleveland OH, pp 54-56, June 1995

[24] R M H Cheng, K Thanjavur, R Rajagopalan and F Luong, *Automated Probing and Compensation for Part and Fixture Variations for a Yamaha Zeta-1 Deburring Robot*, Proceedings of the 2nd Asia-Pacific Conference on Control & Measurement, pp 291-297, Wuhan-Chongqing, China, 1995

[25] *IDC Series 543 Digimatic Indicator Operation Manual No. 3041*, Mitutoyo Corporation, 1992

[26] *Kaman Instrumentation Manual*, Kaman Sciences Corporation, Colorado Springs, Colorado, 1994

[27] C C. Huang and Y F Zheng, *Integration of Vision and Laser Displacement Sensor for Efficient and Precise Digitizing*, IEEE International Conference on Multisensor Fusion andIntegration for Intelligent Systems, pp. 305-311, IEEE, New York 1994.

[28] J M Brown and C. J. Newton, *Quantified 3D Imaging of Pitted Aluminum*

Surfaces Using Laser Scanning Confocal Microscopy, British Corrosion Journal, vol 29, pp 261-269, 1994

[29] B H Zhuang, J H Zhang, C Jiang, Z Li and W Zhang, *Precision Laser Triangulation Range Sensor with Double Detectors for Measurement on CMMs*, Proceedings of the SPIE The International Society for Optical Engineering, vol 2349, pp 44-52, 1995

[30] C. R. Brice and C L. Fennema, *Scene Analysis Using Regions*, Artificial intelligence, vol 1, pp. 205-226, 1970

[31] J M Tenenbaum and H G Barrow, *Experiments in Interpretation-Guided Segmentation*, Artificial Intelligence, vol 8, pp 241-274, 1977

[32] R. Hoffman and A K Jain, *Segmentation and Classification of Range Images*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol PAMI-9, no 5, pp 608-620, September 1987

[33] C H. Chen, *Statistical Pattern Recognition*, Hayden Book Co , Rochelle Park, New Jersey, 1973.

[34] J Bezdek and S. Pal, *Fuzzy Models for Pattern Recognition*, IEEE Press, New York, 1992.

[35] L. A. Zadeh, *Fuzzy Sets*, Information and Control, vol 8, pp 338-353, 1965

[36] R. Schalkoff, *Pattern Recognition - Statistical, Structural and Neural Approaches*, Wiley, New York, 1992

[37] T. Pavlidis, *Algorithms for Graphics and Image Processing*, Computer Science Press, Rockville MD, 1982.

[38] W K Pratt, *Digital Image Processing*, John Wiley & Sons Inc , New York, 1991

[39] C C Lee, *Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Parts I and II*, IEEE Transactions on Systems, Man, and Cybernetics, vol 20, no 2, pp 404-417, March-April 1990

[40] H N Koivo and P Viljamaa, *Tuning of Multivariable Fuzzy Logic Controller*, IEEE Conference on Fuzzy Systems, pp 697-701, 1993

[41] J Zheng, P Guo and J D Wang, *STFC: Self-Tuning Fuzzy Controller*, 1992 IEEE International Conference on Systems, Man and Cybernetics, vol 2, pp. 1603-1608.

[42] R F Garcia, *Fuzzy Rule-Based Combustion Control on Air Adjustment Applied to a Coal-Fired Power Plant*, 1994 IEEE International Conference on Systems, Man and Cybernetics, vol 1, pp 408-412

[43] G Chen, T T Pham and J J Weiss, *Fuzzy Modelling of Control Systems*, IEEE Transactions on Aerospace and Electronic Systems, vol 31, issue 1, pp 414-429, January 1995

[44] M Sugeno and T Yasukawa, *A Fuzzy Logic-Based Approach to Qualitative Modelling*, IEEE Transactions on Fuzzy Systems, vol 1, no 1, pp 7-31, February 1993

[45] T. Takagi and M. Sugeno, *Fuzzy Identification of Systems and its Applications to Modelling and Control*, IEEE Transactions on Systems, Man and Cybernetics, vol SMC-15 no 1, pp 116-132, Jan. / Feb. 1985

[46] P D. Gader and J M Keller, *Applications of Fuzzy Set Theory to Handwriting Recognition*, Proceedings of the 3rd IEEE Conference on Fuzzy Systems, vol 2, pp

910-917, 1994

[47] J. C H Poon, G M T Man and C H Chan, *The Application of Fuzzy Grammar to Handwritten Character Recognition*, Proceedings of the 1993 IEEE Conference on Computers, Communication, Control and Power Engineering, vol 4, pp 502-505

[48] H Hoppe, T DeRose, T Duchamp, M Halstead, H Jin, J McDonald, J Schweitzer and W Stuetzle, *Piecewise Smooth Surface Reconstruction*, SIGGRAPH '94 Conference Proceedings (Computer Graphics), pp 295-302

[49] M Fang, S Chen and X Wang, *Surface Fitting From Digitized Data Points*, Proceedings of the IEEE International Conference on Robotics and Automation, vol 3, p 1011, May 1993

[50] Y Chen and G Medioni, *Fitting a Surface to 3-D Points Using an Inflating Balloon Model*, Proceedings of the 1994 Second CAD-Based Vision Workshop, pp 266-273

[51] S S Rao, *Optimization Theory and Applications*, Wiley Eastern Ltd , New Delhi, 1984

[52] D S Hochbaum and S Seshadri, *The Empirical Performance of a Polynomial Algorithm for Constrained Nonlinear Optimization*, Annals of Operations Research, vol. 43, iss 1-4, pp 229-248, October 1993

[53] R. G Carter, *Numerical Experience With a Class of Algorithms for Nonlinear Optimization Using Inexact Function and Gradient Information*, SIAM Journal on Scientific Computing, vol 14, iss 2, pp 368-388, March 1993

[54] W. P Fox, *Using Microcomputers in Undergraduate Nonlinear Optimization*, Collegiate Microcomputer, vol 11, iss 3, pp 214-218, August 1993

[55] L K Gillespie, *Robotic Deburring Handbook*, Society of Manufacturing Engineers, Dearborn, Michigan, 1987

[56] G Duelen, H Munch, D Surdilovic and J Timm, *Automated Force Control Schemes for Robotic Deburring: Development and Experimental Evaluation*, Proceedings of the 1992 IEEE International Conference on Industrial Electronics, Control, Instrumentation and Automation, vol 2, pp 912-917

[57] S Liu and H. Asada, *Adaptive Control of Deburring Robots Based on Human Demonstration Data*, Proceedings of the 1992 IEEE Conference on Industrial Electronics, Control, Instrumentation and Automation, vol. 2, pp 946-952.

[58] G M Bone and M A Elbestawi, *Sensing and Control for Automated Robotic Edge Deburring*, Proceedings of the 1992 IEEE International Conference on Industrial Electronics, Control, Instrumentation and Automation, vol 2, pp. 927-933

[59] A G Ulsoy and Y Koren, *Control of Machining Processes*, Transactions of the ASME - Journal of Dynamic Systems, Measurement and Control, vol 115, iss 2B, pp 301-308, June 1993

[60] G Masotti and T Bombardi, *Automatic Production of NC Code for Machining Form Features in Generic Parts*, Computing and Control Engineering Journal, vol 3, iss 6, pp 287-295, November 1992

[61] Y S. Tang and W. S Chang, *Dynamic NC Simulation of Milling Operations*, Computer Aided Design, vol 25, iss 12, pp 769-775, December 1993.

[62] A T J Hayward, *Repeatability and Accuracy*, Mechanical Engineering Publications Ltd., New York, 1977

[63] M Temple-Raston, R M H Cheng and R Rajagopalan, *An Adaptive Geometric Scheme for Refurbishing Engineered Components*, Centre for Industrial Control Internal Report # CIC-0063, Concordia University, Montreal, Canada, August 1995

[64] R M H Cheng and B Roussel, *A Fuzzy Logic-Based Algorithm for Feature Rejection*, Proceedings of the 2nd Asia-Pacific Conference on Control & Measurement, Wuhan-Chongqing, China, June 1995, pp 301-305

[65] R J Thomas and S Finney, *Calculus and Analytic Geometry*, Wiley, New York, 1979