Polygon subdivision for pocket machining process planning

Abhijeet Pramod Makhe
Iowa State University

Polygon subdivision for pocket machining process planning

by

Abhijeet Pramod Makhe

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Industrial Engineering

Program of Study Committee:
Matthew Frank, Major Professor
Frank Peters
Eliot Winer

Iowa State University

Ames, Iowa

2005

Graduate College
Iowa State University

This is to certify that the master's thesis of

Abhijeet Pramod Makhe

has met the thesis requirements of Iowa State University

Signatures have been redacted for privacy

TABLE OF CONTENTS

ABSTRACT

In the process planning for pocket machining, selection of the optimal tool sizes and minimizing the number of plunging operations are among the most important factors in minimizing the machining time.

This thesis presents a new approach for optimal tool selection of arbitrary shaped pockets based on a polygon subdivision technique. The pocket is subdivided to obtain smaller sub-polygons. The tools are selected separately for each sub-polygon and then the optimal set of the tools for the entire pocket is obtained based on minimizing both the machining time and the number of tools used to machine the pocket. Finally, the sub-polygons are sequenced in an optimal order to eliminate the requirement of multiple plunging operations.

The approach presented is an improvement over previous work because it makes an effective use of the polygon subdivision strategy to improve the machining time as well as reducing the number of plunges. The implementation examples of this approach suggest that the machining time can be improved as much as 75%.

CHAPTER 1 - INTRODUCTION

Milling is the mechanical process of removing material from a piece of stock though the use of a rapidly spinning circular milling tool in order to form some desired geometric shape [20]. Pocket milling is one of the most common types of milling processes. In this process, a region of a workpiece has to be cut to a constant depth with some set of milling tools.

When the number of pockets in a workpiece is very large, the machining time increases rapidly. For example, Zelinski [19] discusses toolpath design at the Boeing Company and suggests that pocketing is so common in aerospace machining that improving its efficiency would have a significant effect on the cost of an aircraft. Thus to achieve the objective of reducing the total machining time of a workpiece, it becomes necessary to reduce the machining time of these pockets. There has been a lot of research [1, 2, 3, 4, 5, 6, 7] on pocket milling. Various techniques have been developed varying from simple offset generation to more sophisticated approaches using voronoi diagrams. However, most of the researchers have focused on design strategies using a single tool.

For a simple pocket, a single-tool approach can be very effective but as the complexity of the pocket increases, this approach starts becoming very inefficient and results in higher machining time. The main disadvantage associated with the use of multiple tools has historically been the tool change time but with the advent of rapid tool changers, this has been significantly reduced. The machining time required for a pocket depends on the nature of its boundary, whereby machining time generally increases with pocket shape complexity. The presence of geometric features like islands inside a pocket further increases its complexity leading to additional machining time.

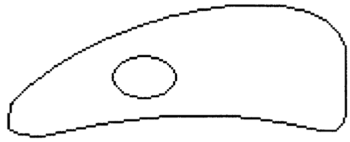

Figure 1.1: Pocket with an island

The tool sizes used for the machining of a pocket significantly affect the total machining time. Larger tools will generally decrease the machining time; however, there are some practical limitations on the size of the tool that can be used. These limitations depend on the boundary and the interior features of the pocket. While using a single tool approach, irregular pocket boundaries and an increasing number of islands requires the use of smaller diameter tools that result in longer machining times. The longer machining times and discontinuous toolpaths in single tool approach makes a strong case for the use of multiple tools for pocket machining. Theoretically, there is no limitation on the number tools that can be used for the machining but in practice, there will too many semi-redundant tools in the process and the solution would not be optimal. The number of tools used should be such that the total processing time is minimized.

The number of times that the cutting tool has to make an initial plunge into the material is also a significant problem while pocket machining. Since plunge milling is done at a considerable slow feed rate, the process has a significant effect on the machining time. The use of multiple tools will almost always form more than one convoluted toolpath center [8, 9, 10, 11], making it necessary to plunge more than once.

From the above discussion, it is clear that the main problem in pocket machining is the selection of the optimal number of tools and minimization of plunging in order to minimize the machining time. This research proposes a new methodology to solve this optimization problem using pocket subdivision and a branch and bound technique.

This chapter briefly presented the problems associated with pocket milling. Chapter 2 reviews the relevant literature, while Chapter 3 provides the specific problem details and the proposed solution. In Chapter 4, the polygon subdivision method is discussed. In chapter 5, the tool selection and tool path design strategy is presented and Chapter 6 provides an implementation of this method with examples. Lastly, chapter 7 presents conclusions and future work.

CHAPTER 2 – LITERATURE REVIEW

There has been a lot of research in the field of toolpath design for pocket machining. There are numerous methods designed for the toolpath within a pocket. These methods vary from simple offset generation or contour parallel methods to complex methods like those based on voronoi mountains.

In the contour parallel method [1, 2, and 12] for machining of arbitrary shaped pockets, row offsets go through decomposition, removal of interfering chains and merging to form the clean offset as illustrated in Figure 2.1. The successive offsets of this clean curve form the toolpath for the pocket.

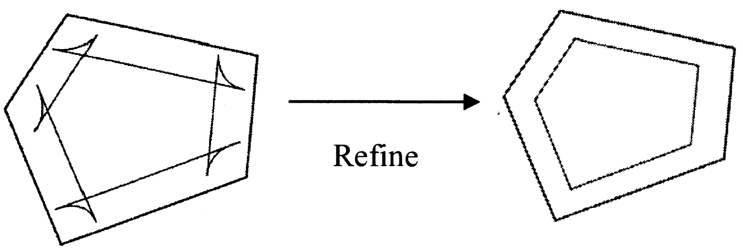

Figure 2.1: Contour Parallel Method

In another approach based on contour parallel machining [3], boundary B-spline curves are converted into Bezier curves by knot insertion until the required tolerance is obtained. The toolpath is designed by intersecting offset removal and decomposition of the profiles. The Bezier convex hull is used to avoid any overcut in the machining.

In pocket machining using distance maps [4], the author constructed the discrete distance map by determining the closest curve segment and minimum distance for each point between the inside boundary and the offset of the boundary and then applied the z-buffer method [5]. In the z-buffer method, a right cone is constructed and moved along the boundary with different color assignment for each curve. The toolpath is generated by the extraction of the characteristic points from the distance map and connection of all the offset profiles.

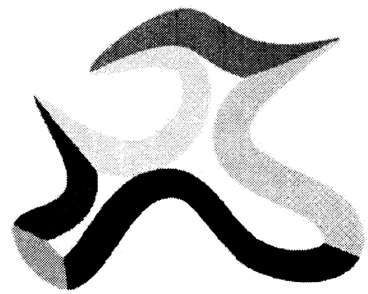

Figure 2.2: Distance Map

In an approach based on the monotonic pouches designed from voronoi diagrams [6], the author first draws the voronoi diagram of the pocket. Next, an imaginary surface over the voronoi diagrams is created by varying the z-coordinate and keeping the x and y-coordinates constant. The mountains obtained in this process are used to determine the location of the bottlenecks in the pocket. The paths for individual pouches are designed and joined together to construct the toolpaths. In this approach, a proximity map of the features is created and used to determine the set of tools that can be used to machine the pocket. According to the author, the largest tool that can be used for the rough cut is the smallest neck or bottleneck and for the finish cut, it is equal to smallest bottleneck or smallest fillet radius.

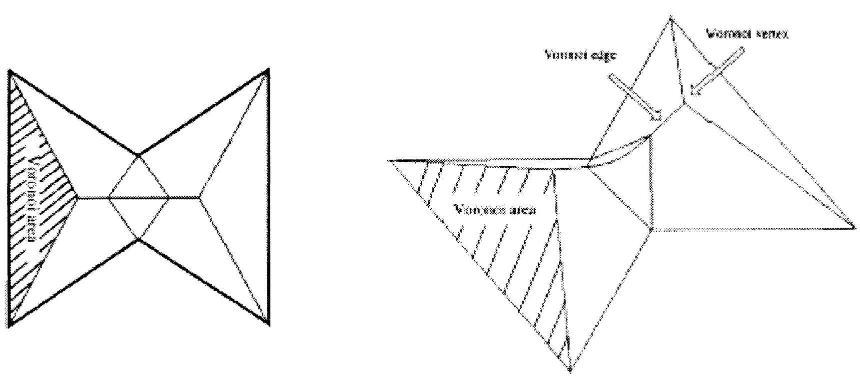

Figure 2.2: Voronoi Diagram and Voronoi Mountain

In the zig-zag machining method [7] the pocket is machined by parallel motion of the tool. This method is useful when the machining tool has some preferred direction.

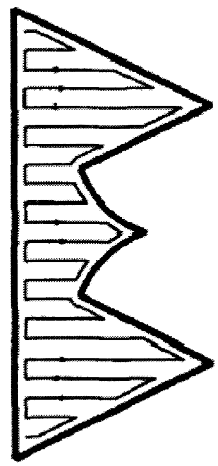

Figure 2.3: Zig-zag Machining

In most of the above methods plunging has been recommended for each of the toolpath centers before machining. Since plunging feed rates are much lower than linear cutting motion feed rates, it takes more time. Efforts are also made to find the optimal tool based on economic constraints, but the authors restricted themselves to simple shapes and zig-zag machining. These are some methods designed for the generation of the toolpaths for the pocket. All the above methods prefer to use a single tool to machine the whole pocket. Unfortunately, there can be a small neck present, which would force the use of a small tool to machine the entire pocket. The increased use of complex pocket in modern manufacturing makes these approaches less effective.

Some of the researchers realized the advantages of the use of multiple tools for pocket machining. Although there are a few algorithms designed for determining tool sizes, sequencing of the tools remains a problem. In addition, the use of multiple tools can increase the number of plungings required. There have been no efforts so far on reducing the number of plungings in order to save machining time.

In the multiple tool selection method based on voronoi diagrams [8], the voronoi mountain of the pocket is created to obtain the unmachined area. The tool sizes are based on these area calculations. The method based on geometric and volumetric calculations [10,11] used feature-based analysis for selection of the tools. There are various types of decision graphs generated to support the system. The time loss due to increased plunging has been ignored while calculating the total machining time.

Since the existing systems do not take into consideration all the important parameters while selecting the tools for the pocket machining, a new method is proposed in this thesis. The proposed method uses multiple tools to minimize the machining time as well as the number of plungings. The next chapter starts with the discussion of the problem framework and solution approach.

CHAPTER 3 – PROBLEM DEFINITION AND SOLUTION OVERVIEW

This chapter begins by presenting the problems associated with the use of a single tool approach. Next, the terminology used in this research is explained and an overview of a proposed new methodology is provided.

3.1 Problems associated with the Single Tool Approach:

Pockets can be defined as the sculptured regions on the face of the workpiece formed by the impression of some shape to a given depth. The main parts of interest in the pocket for this research are the pocket boundary, islands and necks.

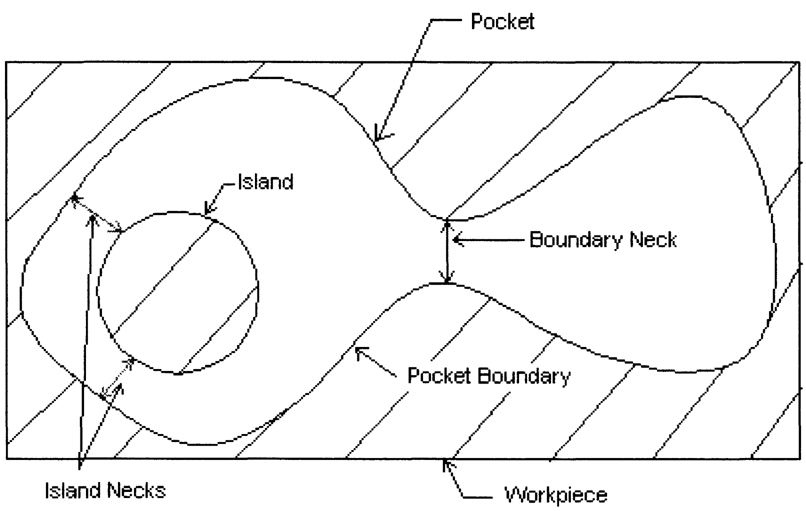

Figure 3.1: Parts of a Pocket

The pocket boundary is the wall defining the region of the pocket. These boundaries define the geometry of the pocket on the workpiece. It is this wall that makes the pocket easier or more difficult to machine a more complex boundary is generally more difficult to machine. The tool size to be used for the machining is partly defined by this wall. An irregular boundary will require a smaller tool and more time while a smooth boundary with larger radii can be machined with a comparatively larger tool in a shorter amount of time.

The other important problem that is caused by the boundary is the formation of the boundary necks. A boundary neck can be defined as a region where the two sides of the boundary wall come very close to form a narrow region. Figure 3.1 shows a boundary neck. The formation of these necks put further restriction on the size of the tool that can be used for machining. If there are several necks in a pocket with varying width, a single tool strategy would perhaps have to use the smallest width of them as the tool diameter. If this tool is very small compared to the overall pocket area, then machining time can be very long.

The regions called *islands* form the other type of the necks, i.e., island necks. An island can be defined as an unmachined region (by design) inside the pocket area. Figure 3.1 show an island present inside the pocket area. Islands, like any pocket boundaries, cannot be overcut while machining. They are considered a major source of problems because their presence increases the complexity of the pocket and requires significant modification of the methodology to design the tool path. The presence of one island leads to the formation of at least two necks, one on either side. The island necks are shown in the figure. These necks also affect the tool path design process by restricting tool size.

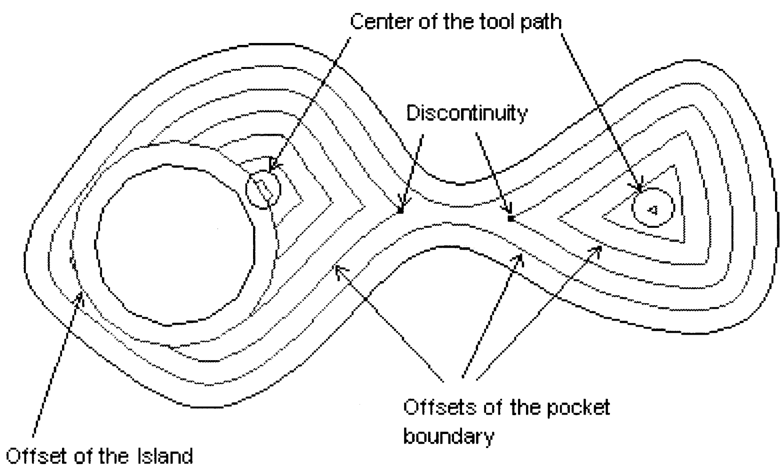

Figure 3.2: Offsets of the pocket boundary and the island

The necks not only complicate the tool path design procedure by tool size restriction but can also lead to discontinuous tool paths in the pocket. Figure 3.2 shows these discontinuous paths. These paths are obtained by offsetting the pocket boundary and the island by a distance equal to the radius of the selected machining tool. One of the advantages of a single tool approach is a continuous tool path; however, the presence of islands and necks lead to discontinuities in the tool path. A discontinuity in the tool path results in more than one tool path center. Figure 3.2 shows the formation of two tool centers because of discontinuity in the tool path. Each of these tool path centers needs to be plunged as suggested by Hansen and Arbab [1]. In plunging, the tool is fed along the direction of its rotational axis down to the pocket depth in order to begin machining. As the number of toolpath centers increases plunging also increases.

Since the boundary necks and island necks significantly increase the complexity of the toolpath in a pocket, they are the prime source of toolpath discontinuity. To overcome the problem of simply using one small diameter tool, this research will use multiple tools to machine the pocket. The new approach will effectively reduce the problems associated with the necks as well as minimize the total pocket machining time.

3.2 General overview of solution approach:

This research work uses a pocket subdivision process to solve the problems of the single tool approach. The subdivision process can be defined as the process of dividing the pocket into smaller regions. Each of these regions is analyzed separately, and then in the tool selection process they will be re-combined so that the continuity of the pocket tool path is maintained.

In this work, the pocket boundaries and islands are defined by freeform curves. The pocket boundaries consist of convex and non-convex regions while the islands are assumed to be non-convex.

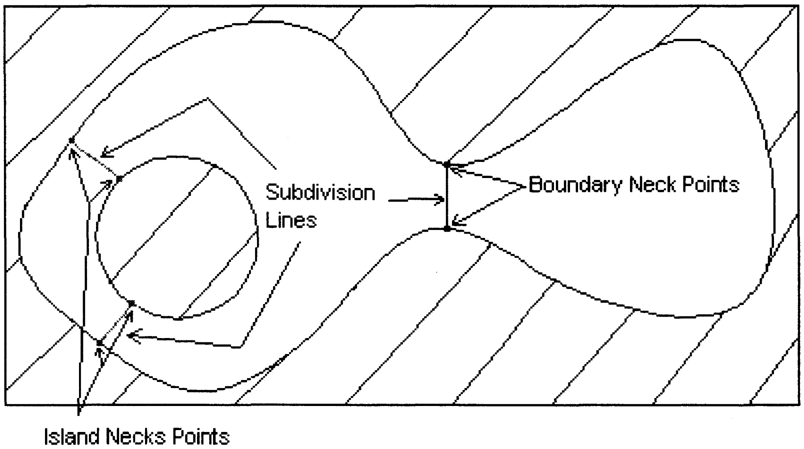

Figure 3.3: Neck Subdivision

The boundary necks and island necks, having the narrowest width in the pocket, are the first subdivision regions. This subdivision helps to isolate the boundary regions that require very small radius tools. These subdivision lines are straight lines between the neck-forming points. Figure 3.3 shows the points responsible for the formation of the necks and the subdivision lines.

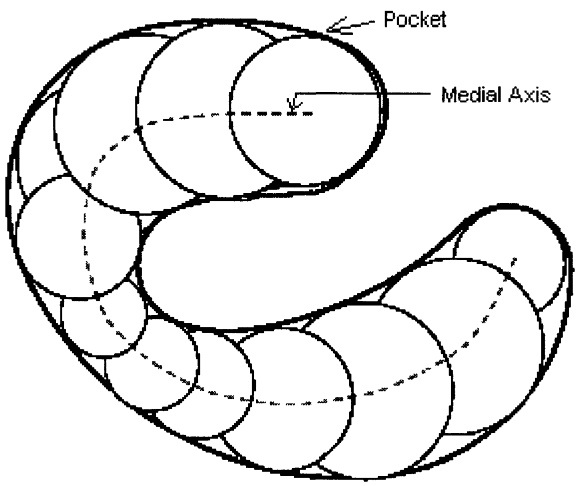

Figure 3.4: Medial Axis or Skeleton

For this subdivision, the medial axis or *skeleton* of the pocket region is used. The skeleton of a pocket is defined as the centerline representation of the pocket region. The pocket boundary is shrunk until a line representation of the pocket is obtained. This line would maintain the connectivity of the different parts of the pocket.

The medial axis can be computed by joining the centers of the inscribed circles in the pocket. In this research, the medial axis of the pocket is computed using an image-morphing technique. An image of the pocket is converted into a monochromatic image. The background of the pocket is painted in black. This image is then used for the morphing purposes and skeleton design. Figure 3.4 shows the medial axis of an arbitrary shaped pocket.

As illustrated in Figure 3.5, neck subdivision divides the pocket into smaller sub-polygons, which do not have any necks.

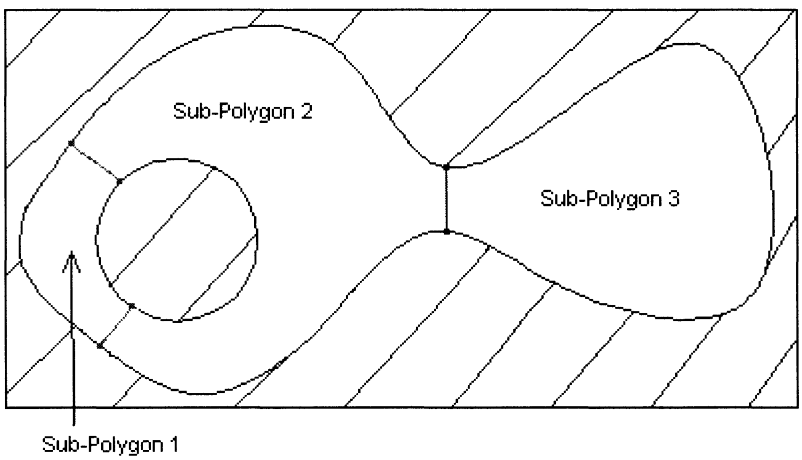

Figure 3.5: Pocket after subdivision

Each sub-polygon is now considered a separate distinct region for analysis and initial tool selection. The necks associated with them aid the reunification of the pocket in the final tool selection process.

The initial tools are selected separately for each sub-polygon depending on its features. A small region in one sub-polygon does not affect the tool selection for other sub-polygons. Thus, this subdivision facilitates the use of multiple tools. In the final tool selection process, the tools responsible for total machining time reduction would not be discarded.

The final tool selection process attempts to optimize the selection procedure based on the machining time as well as the number of tools. The weight for each of them can be adjusted according to the requirements. In this work, more weight has been given to the machining time. A branch and bound approach is used to solve this optimization problem.

The new approach can be summarized into three important steps –

1) Polygon Subdivision – In this step pocket is subdivided at boundary and island necks.
2) Tool Selection – In this step with the initial tools, tool filtering is done to obtain final tools, and
3) Polygon Sequencing – In this step, correct sequencing of the sub-polygons to avoid multiple plunges is done.

The following two chapters will explain this method in more detail.

CHAPTER 4 – POLYGON SUBDIVISION APPROACH

This chapter presents a method for subdividing a pocket into several distinct regions. There are two particular places where subdivision may be appropriate, island necks and boundary necks. The chapter presents two separate approaches for each type of division.

The pocket boundary and islands are defined by freeform curves. Knots are inserted in the control polygon of these curves to obtain a better representation.

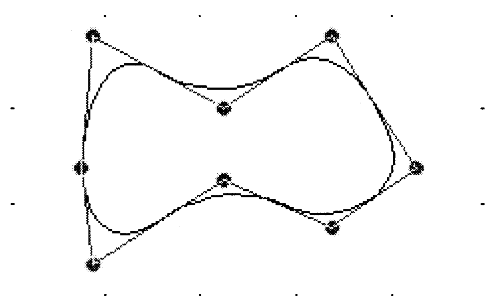

Figure 4.1a: Initial Control Polygon

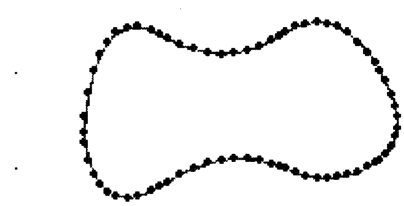

Figure 4.1b: Final Control Polygon

Figure 4.1a shows the initial control polygon of the boundary curve and figure 4.1b shows the control polygon after knot insertion. As shown in the figures the final control polygon is a good representation of the actual curve and can be used for further calculations.

4.1 Subdivision at the island necks

The presence of an island leads to the formation of at least two necks. The purpose of the subdivision at the island necks is to avoid the consideration of these necks from the initial tool selection process. If these regions are present, their width will limit the tool diameter that can be used for the adjoining sub-polygons. An effective way to avoid these necks from affecting the initial tool selection process would be to divide the polygon at their location, in order to create smaller sub-polygons that have these necks only as a part of their boundary.

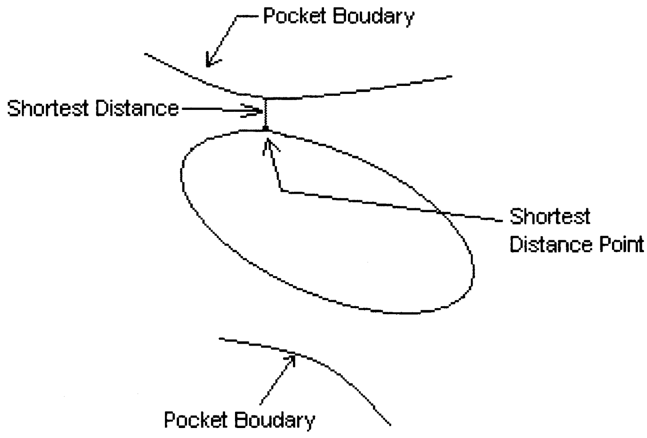

Figure 4.2: Shortest Distance Point

The shortest distance between the island and the pocket boundary is the narrowest region created by the presence of that island. Thus, these regions become the first choice for the subdivision. For this subdivision, the shortest distance island and the boundary curve is measured by checking the distances between the pocket boundary and island points. The point on the island corresponding to this shortest distance is referred as the *shortest distance point*. Figure 4.2 shows a shortest distance point.

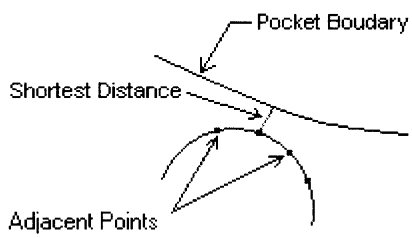

Figure 4.3: Adjacent points to shortest distance point

To find the second neck corresponding to the same island, we need to divide the island into two equal parts otherwise the other shortest distance point can come on adjacent point leading to the formation of very small sub-polygon. Figure 4.3 show the adjacent points of a shortest distance point.

The first step towards this region definition is to find the longest line on the island. The longest line can be defined as the maximum distance between the island control points, i.e., diameter of the island.

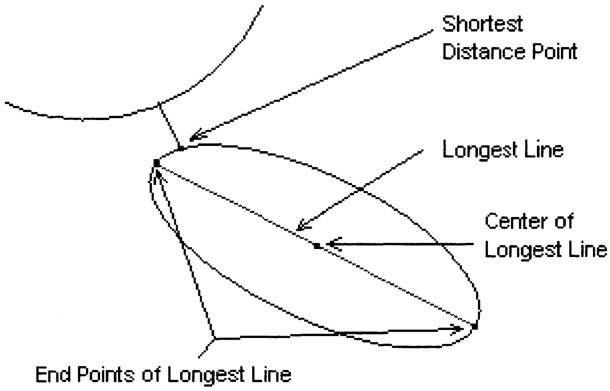

Figure 4.4: Longest Line

The center of the longest line serves as a reference point through which the line dividing the island into two parts will pass. To define the feasible area for the second neck, it has been decided to exclude 180°, i.e., half of the island containing the shortest distance point. The line dividing the island into two parts is referred as the Division Line.

The distance from the shortest distance point to the center end points of the longest line is calculated. The subdivision of the island is based on these distances.

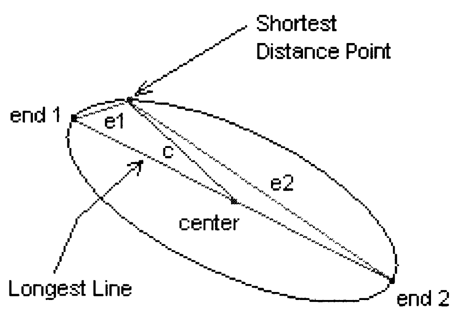

Figure 4.5: Distance Calculation

The variables e1 and e2 represent the distances from the end points of the longest line and c is the distance from the center of the longest line to the Shortest Distance Point. If e1 or e2 is smaller than c, the perpendicular to the longest line through center serves as the division line.

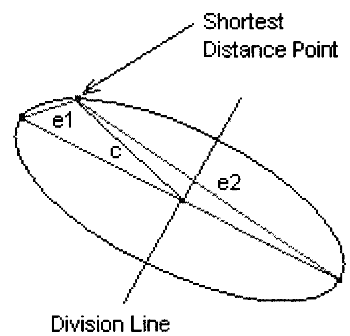

Figure 4.6: Division line when shortest point is nearer to an end of longest line

On the contrary, when distance c is less than e1 and e2, then the longest line serves as the division line.

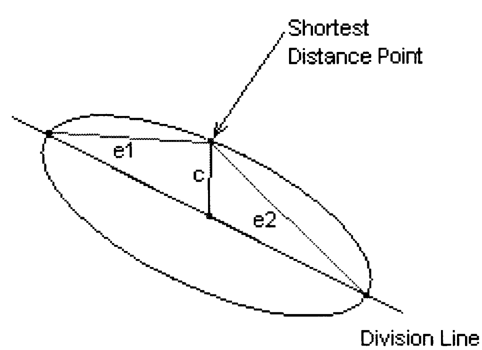

Figure 4.7: Division line when shortest point is nearer to center of longest line

The part having shortest distance point on its side is referred as the side1 of the island, while the part on the other side of the division line is referred as the side 2 of the island. The shortest distance between the side 2 of the island and the pocket boundary gives the second neck on the island. These points are joined by a straight line to obtain the second subdivision line on the island.

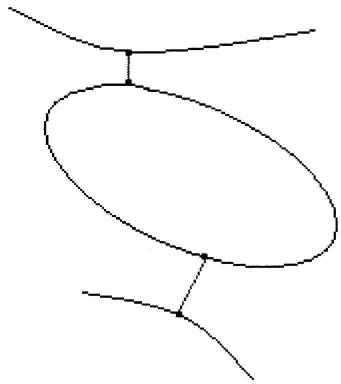

Figure 4.8: Island Subdivision

4.2 Subdivision at the boundary necks

The boundary necks being the narrow regions formed by the pocket boundary, cause a restriction of candidate tool sizes and formation of multiple centers of the tool path. This section describes the methodology of the subdivision process at the boundary necks.

The information from the initial control polygon is used to determine the convex and non-convex parts of the boundary curve. This classification of the boundary curve into two types of parts is necessary, as we have noticed that the boundary necks would mostly be present on the non-convex parts. The points at which the control polygon changes its side on the curve can be used as the characterization points. If the control polygon is outside the curve, that part of the curve is convex otherwise it is non-convex. The control points that are inside the actual boundary curve need to be selected first to characterize the boundary curve.

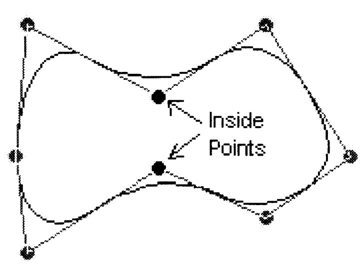

Figure 4.9: Control Points inside the boundary curve

Figure 4.9 illustrates the points that are inside the curve. The control points that are inside the actual curve would have a non-convex region beside them. These inside control points are joined by a straight line to the preceding and succeeding control points. Intersection points are obtained where these lines cross the boundary curve.

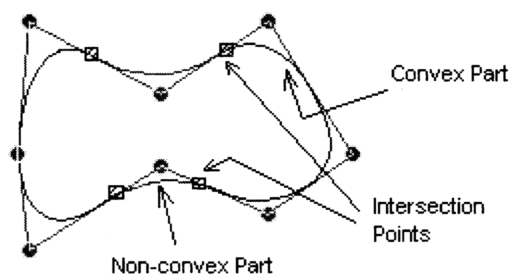

Figure 4.10: Convex and Non-Convex Parts

These intersection points are important as they determine the end points of the convex and non-convex regions. If there are multiple intersection points on any of the above control polygon lines, the intersection point farthest from the inside point is selected as that would be the actual point where the control polygon changes its side on the curve. Since the non-convex regions are the regions of main concern with respect to finding boundary necks, they will be further analyzed in the following sections.

In the boundary neck detection process, the skeleton of the pocket performs a very important role of verifying the presence of a neck.

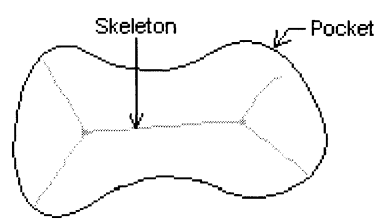

Figure 4.11: Skeleton of Pocket

The arms of the skeleton have to be separated at the skeleton intersection points since each of these will be used separately when determining the presence of neck.

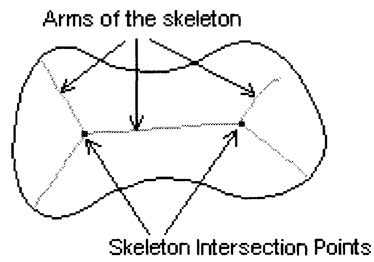

Figure 4.12: Intersection Points and Arms of skeleton

A pixel-level operation and object detection process in the image morphing technique has been used to obtain the intersection points and the arms of the skeleton. As two non-convex regions form a neck in the pocket, each of the non-convex regions has to be checked for the presence of a neck with every other non-convex region. The process of neck detection is explained in the following section.

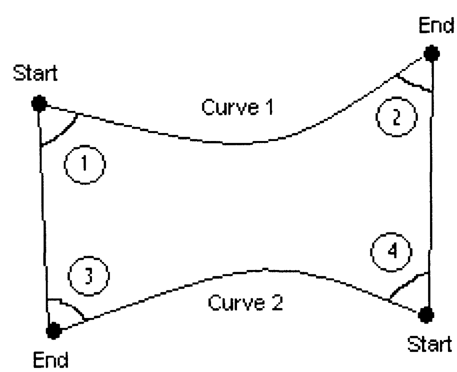

Figure 4.13: Angles between the two Non-Convex curves

The ends of the two non-convex curves under consideration are joined to form a polygonal structure as shown in Figure 4.13. This leads to the formation of four angles between the curves. The determination of these angles between the two non-convex curves under consideration is an important step in the neck detection process. The four angles are measured at the locations shown in the Figure 4.13. The four angles can be defined as:

Angle 1 – Angle between curve 1 and line between 'Start point of curve 1' and 'End point of curve 2'.

Angle 2 – Angle between curve 1 and line between 'End point of curve 1' and 'Start point of curve 2'.

Angle 3 – Angle between curve 2 and line between 'Start point of curve 1' and 'End point of curve 2'.

Angle 4 – Angle between curve 2 and line between 'End point of curve 1' and 'Start point of curve 2'.

Although the angles can roughly indicate the presence of a neck, the combination of their values is required for the confirmation of an initial neck. The condition for confirmation of the initial neck is that the sum of the angles on any line joining the ends of the curves should be less than 180°. This means that the sum of Angle 1 and Angle 3 and Angle 2 and Angle 4 in figure 4.13, both should be less than 180° for the presence of an initial valid neck.

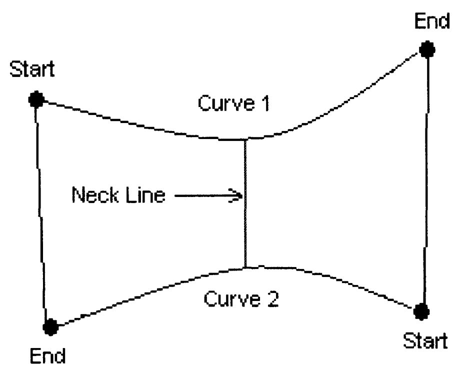

Figure 4.14: Neck Line between Curve 1 and Curve 2

All the valid non-convex curves are further analyzed to confirm the presence of necks. The shortest distance between these curves is calculated and joined by a straight line. These lines are called the Neck lines. The combinations of the curves that fail to represent a valid neck at this stage will be excluded from further analysis.

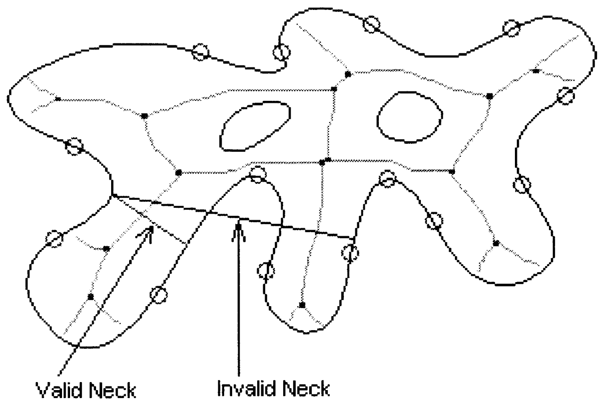

Figure 4.15: Valid and Invalid Neck

The skeleton of the pocket has been used to verify the validity of the neck lines. Each skeleton arm is a representative of one sub-polygon. So, the skeleton arms can be used to check if the neck line is cross into more than one sub-polygon. The neck lines crossing into more than one sub-polygon are removed. To check this condition, the number of intersections of the neck line with the arms of the skeleton is checked. If any of the neck line intersects more than one skeleton arm, it must cross into more than one part of the pocket and thus is invalid. Figure 4.15 illustrates a valid and invalid neck. This validity check removes most of the invalid necks from the selection; however, there may still be some invalid necks present, which must be filtered before the final neck line list determined.

After the neck lines having more than one skeleton arms intersection are removed, there is a possibility that there will be some neck line that intersects an island or the intersection point of the skeleton.

The neck lines intersecting with the islands ignore the presence of the island so they are removed. It is assumed that the skeleton intersection points have all the skeleton arms present, so the neck lines intersecting with the intersection points of the skeleton are also removed from further consideration.

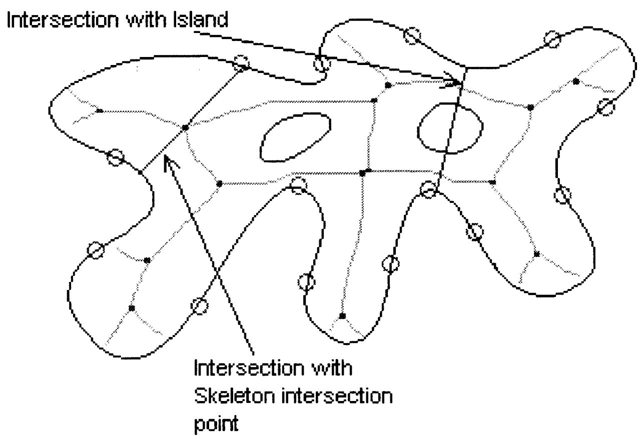

Figure 4.16: Neck lines filtered because of intersection with island and intersection point of the skeleton

This filtering process completes the boundary neck subdivision. The neck lines in the selection after this filtering represent the final necks in the pocket. These neck lines divide the pocket into the sub-polygons. These sub-polygons are used for the initial tool selection process. The next chapter will present the details of the tool selection process.

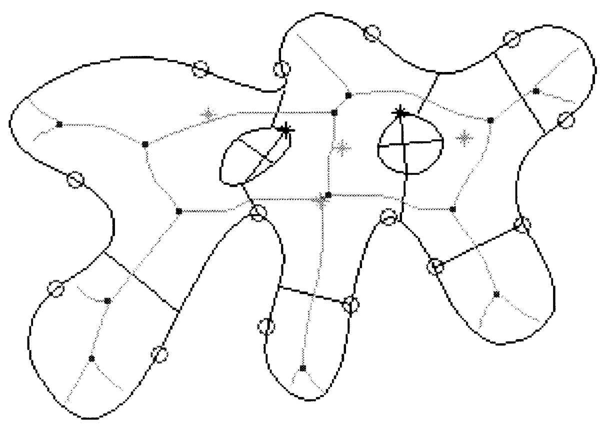

Figure 4.17: Final Subdivision of the pocket

CHAPTER 5 –TOOL SELECTION AND TOOL SEQUENCING

This chapter presents a methodology for tool selection based on the machining time and number of tools. The tool selection is a three step process, initial tool selection, tool refining, and final tool selection. The initial tool selection determines tool sizes for each sub-polygon while the refining process eliminates reoccurring tools. The final tool selection determines the suitable tools to machine the entire pocket.

5.1 Initial Tool Selection

The initial tool selection process uses the sub-polygons of the pocket, obtained from polygon subdivision, to determine the initial tool sizes. Each sub-polygon is analyzed separately for three different types of tools, namely, the *largest tool*, *smallest tool* and the *neck tool*. A separate method is designed to determine each of these tools. The largest tool and smallest tool methods are based on the voronoi points of the pocket while the neck tool method is based on neck diameter.

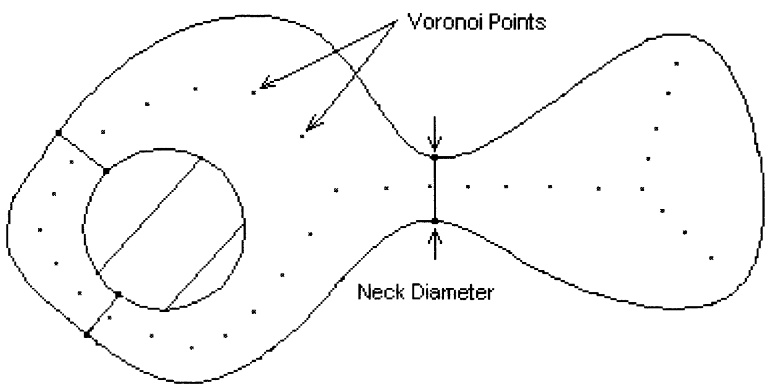

Figure 5.1: Voronoi Points and Neck Diameter

The voronoi points represent the medial axis or centerline of the pocket so they can be used towards the determination of the center of the largest and smallest circle that can be fit in the sub-polygon.

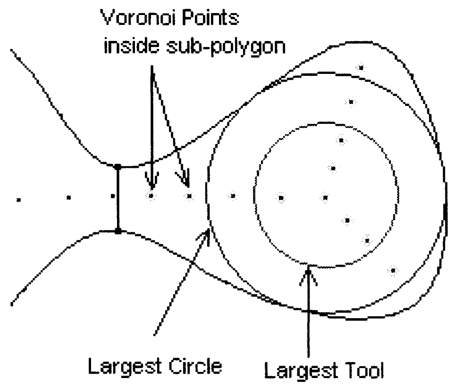

Figure 5.2: Largest Circle and Largest Tool

The process of determining the largest tool uses the largest circle that can be fit in the sub-polygon under consideration. The diameter of the largest tool to be used for the sub-polygon is equal to 0.7 times the diameter of largest circle. The multiplication factor 0.7 is selected somewhat arbitrarily; however, the idea is to reduce the size of the largest circle to obtain a tool that can be used to machine and not only plunge. The voronoi points that are present inside the sub-polygon are obtained. The maximum distance of each of the voronoi point from the sub-polygon boundary points is calculated. The point having the maximum distance from the boundary is selected as the center of the largest circle. The radius of the largest circle is the maximum distance of the center point from the boundary points. The multiplication of this circle diameter by multiplication factor of 0.7 represents the largest tool for the sub-polygon.

The smallest tool is used for machining the material that cannot be machined by the roughing operation with the larger tools. The size of the smallest tool is small enough to machine all small curves and areas of the sub-polygon. The method designed for determination of the smallest tool is based on the voronoi points of the pocket. The islands are assumed to be convex in the current version of this method and hence ignored from the consideration when calculating the smallest tool.

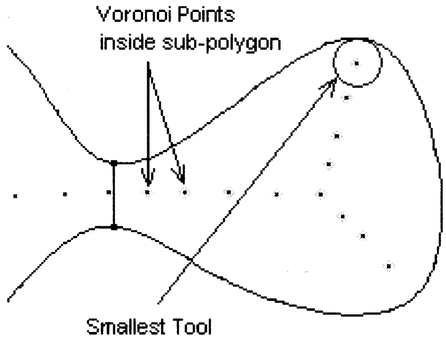

Figure 5.3: Smallest Tool

The voronoi points inside the sub-polygon under consideration are obtained. The minimum distance of each of the voronoi point from the sub-polygon boundary points is calculated. The point having the smallest minimum distance is selected as the center of the smallest tool. The minimum distance of this point from the boundary is the radius of the smallest tool.

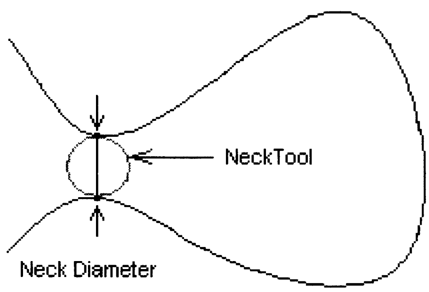

Figure 5.4: Neck Tool

Neck tools are the tools having the diameter equal to the necks in the sub-polygon. They are primarily used so that the necks present in the sub-polygon can be machined. The other important purpose of the neck tools is to connect the adjoining sub-polygons while machining.

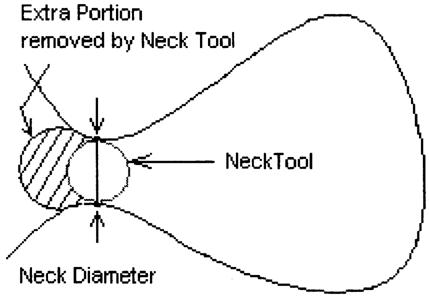

Figure 5.5: Extra portion machined by neck tool

If necks tools were not utilized, each sub-polygon would require plunging before machining. The neck tools remove some extra portion of the adjoining sub-polygon so that the roughing tool can be placed into the next sub-polygon. The use of the same technique to join all the sub-polygons reduces the number of plunge milling operations to one.

Since the tool diameters obtained using these methods will likely not be standard size tools, all the tools are standardized, for example, to 1/16 of an inch. When choosing a standard tool size, the next smaller tool diameter will be used.

5.2 Refining the tool selection for the sub-polygon

The initial tool selection chooses three types of tools for each sub-polygon. The number of tools obtained for each pocket may be large if there are many necks. The refining process eliminates the unnecessary tools from consideration. After the refining process, each sub-polygon will have only two tools associated with it, a roughing tool and a finishing tool.

The refining process compares all the neck tools with the smallest tool and the largest tool. First, the neck tools having diameter larger than the largest tool are eliminated. Then, if a neck tool has a diameter smaller than the smallest tool, it replaces the smallest tool. All other neck tools are ignored henceforth. Each sub-polygon now has only largest and smallest tool associated with it and they are used as roughing and finishing tools respectively.

5.3 Machining Time Calculation

The machining time calculations in this research are based on the area. The area machinable by a tool is used for the roughing and finishing time. The formula derived for the calculation of the machining time can be represented as:

$M_t = (A_m / A_t) * t_d / f$ .... (I)

Where

M_t = Machining time

A_m = Area machinable by the tool

A_t = Area of the tool

t_d = Tool diameter

f = Feed Rate of the tool

The area machinable by the roughening tool is based on the offset generation method.

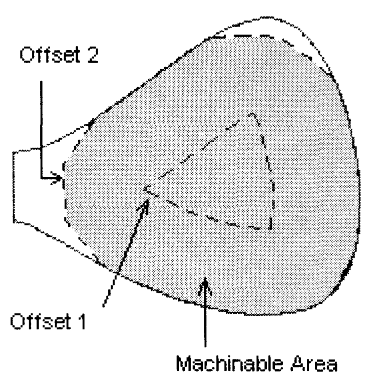

Figure 5.6: Offsets and machinable area

In the offset method, the sub-polygon boundary is first offset by the tool diameter inside the polygon as represented by 'Offset1' in Figure 5.6. 'Offset1' is then offset again by the tool diameter but in reverse direction, which is represented by 'Offset 2'. The area inside the 'Offset 2' represents the machinable area of the roughening tool.

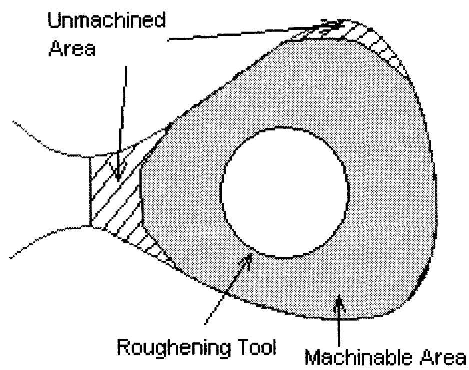

Figure 5.7: Machinable area for finishing tool

The machinable area for the finishing tool is equal to the unmachined area of the roughing, tool as illustrated in Figure 5.7.

The feed rate of the tool is very important towards the calculation of machining time. The feed rate calculations in this research are based on the following equations [22]. The tools are assumed to be made of High Speed Steel.

$f = N * F_t * C_{s\,(rpm)}$ .... (II)

Where

f = feed rate of the tool

N = Number of teeth on the cutter

F_t = Feed rate / tooth

C_s = Cutting Speed in RPM

The calculation of number of teeth is based on the general industry accepted formula [22] for HSS cutters, which is

$N = 19.4 * \sqrt{R} - 5.9$ .... (III)

Where

N = Number teeth required

R = Radius of the cutter

5.4 Final Tool Selection

The two tools per sub-polygon can obtain good machining times for individual sub-polygons but the machining time of the pocket as a whole may not be optimal. To achieve an optimal solution for the pocket, a branch and bound technique is used. In this technique, all the roughing and finishing tools in the pocket are checked for a combination to obtain the optimal time. The number of tools is varied from one to the maximum available tools. Figure 5.8 illustrates a branch and bound tree for three tools.

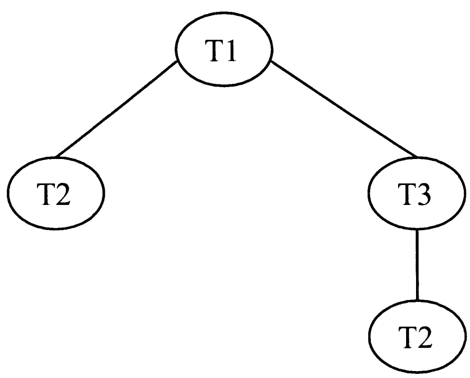

Figure 5.9: Branch and Bound Tree

T1, T2, T3 are the three tools with T1 being smallest and T3 being largest.

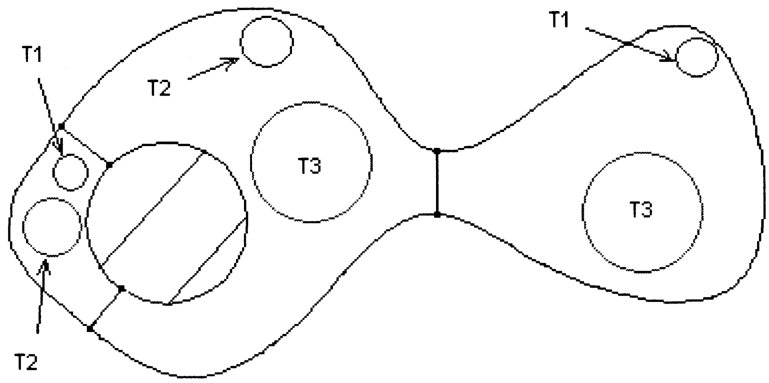

Figure 5.10: Tools for the pocket

It can be noticed in Figure 5.10 that the smallest tool will always be present in the solution, as it is required to machine the smallest boundaries present in the pocket. Depending on this condition, the options for selection would be

No. of tools to be used	Options
1	T1
2	T2 – T1 or T3 – T1
3	T3 – T2 – T1

The width-wise search is preferred over depth-wise search, as it allows evaluating all possible conditions for a given number of tools at the same time. The time for every combination of tools is obtained. The final number of tools is the combination leading to the minimum time. There is a possibility that increasing the number to obtain optimal time is not preferable in some cases. To increase the flexibility of the algorithm, a weight can be specified on the relative importance of minimizing either the time or the number of tools. When the weights are specified, the final decision is based on weighted time, which can be calculated as follows.

$$\text{Weighted Time} = (\text{Machining Time} * \text{Weight}_{\text{Machining Time Minimization}}) + (\text{No. of Tools} * \text{Weight}_{\text{No. of Tools Minimization}})$$

$$\text{Where } \text{Weight}_{\text{Machining Time Minimization}} + \text{Weight}_{\text{No. of Tools Minimization}} = 1$$

The option with the minimum weighted factor would be chosen as the solution.

When numbers of combinations are less, full factorial search can be used for optimal combination instead of branch and bound method.

5.5 Sub-polygon Sequencing

One of the important objectives of this research was to prevent the plunging of the tool at all the tool centers. The sequencing of the sub-polygons in proper order is very important to satisfy this objective. An effective sequence can minimize the number of plunging. The *minimum spanning tree* is a very effective technique that can be used to minimize the length of the path while covering all the nodes. In this research, Sollins algorithm for minimum tree spanning is used with some modifications [21].

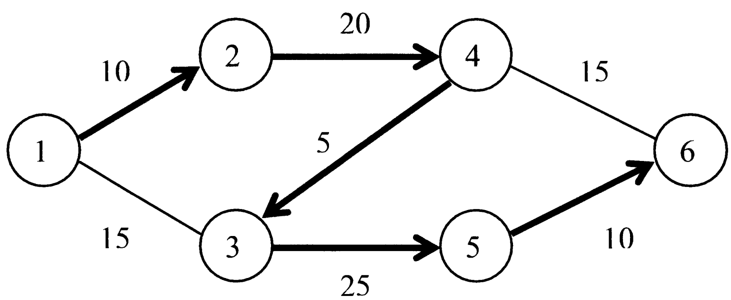

Figure 5.11: Sollins algorithm example

As illustrated in Figure 5.11, Sollins algorithm finds the path in such a way that all the nodes are covered in minimum possible time.

In this research, the length between the sub-polygon centers is used as the search criterion. In Sollins algorithm there is no requirement of bi-directional arcs but in this research, it is possible that a tool has to return to the previous node to continue. For example, if the search is started at node '1' in Figure 5.12, it will stop at '4' and would not be able to move further as '5' would have been already machined. To accommodate such conditions, bi-directional arcs are allowed.

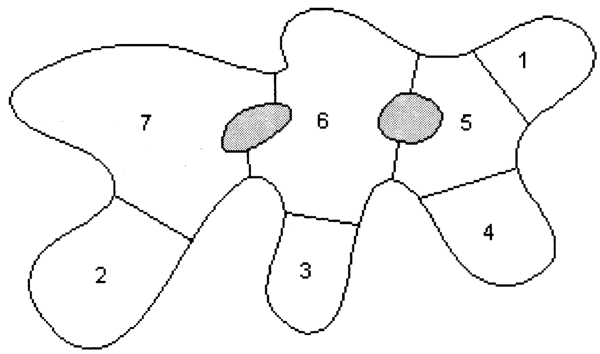

Figure 5.12: Polygon Sequencing

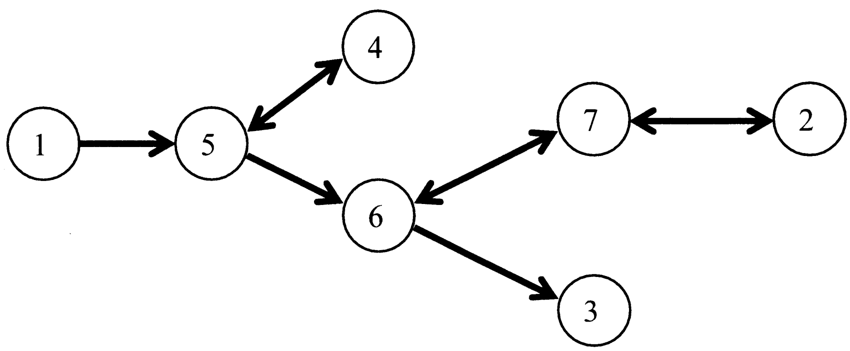

Figure 5.13: Spanning Tree for the pocket in Figure 5.12

The tree spanning sequence in this case would be 1 – 5 – 4 – 5 – 6 – 7 – 2 – 7 – 6 – 3. Although the spanning tree sequence looks very large and repetitive, it would not be the machining sequence as revisited sub-polygons would have been already machined. The actual machining sequence in this case would be 1 – 5 – 4 – 6 – 7 – 2 – 3. When moving through the previously machined areas, it is assumed that the tool can move at rapid speeds and not cutting feed rate. The selected optimal number of tools will follow the tree spanning sequence to produce the optimal machining time under given constraints. The best solution is presented as the results.

The next chapter will present various example pockets with the results obtained from the implementation of this methodology. The results will also be compared to the results for a single tool strategy to compare the effectiveness of this method.

CHAPTER 6 – IMPLEMENTATION

Several pockets with different parameters were analyzed to obtain the optimal combination of tools and tool sequence. This chapter illustrates different pockets with the steps involved in this analysis. This research methodology is implemented in MatLab.

The calculation can be based on the combination of the machining time and number of tools. The weight on each parameter can be specified as the input depending on preferences. For this analysis, all the examples except Example 6.2.3.2 have the entire weighting factor on minimizing machining time with no weight on minimizing the number of tools. Example 6.2.3.2 has 75% weight on minimization of machining time and 25% weight on minimization of number of tools.

The software has been made flexible to accept the control points in clockwise as well as counter-clockwise direction. The image of the pocket required for the skeleton generation has been automatically generated. The only input required for the execution of the software is the control points of the pocket boundary and islands, which can be easily obtained from AutoCAD, and also the depth of the pocket.

6.1 Reading the output of the software

The results begin with the maximum number of tools found in the analysis. The algorithm evaluates all possible combination of tools under the given constraints to find the best solution.

```
Maximum No of Tools can be used =
   3
```

Machining time using the single tool approach assuming each tool center is plunged is presented in the first section. The plunging at each tool center before machining as suggested by Hansen and Arbab [1] has been used for the single tool approach machining time.

```
================================
Machining Time using Single Tool Approach
================================

Tool_Diameter_Used =

   0.2500

Machining_Time =

   1.8525
```

It can be read as the tool of diameter 0.25 inch will be used for the machining and machining time will be 1.8525 minutes.

The next section presents the results of the sub-division approach.

```
================================
Machining Times using Subdivision Approach
================================
```

The results of the sub-division approach present the best option available for each value of the number of tools. For example, after evaluating all the possible combination of three tools, the best results were obtained with the following combination. The procedure to read this data is also explained below.

```
*********************************
Number_of_Tools_Used =

   3

Tool_Diameters =

  0.5000    0.3750    0.2500

Machining_Time =

  1.0039
*********************************
```

This section indicates that the three tools used for this analysis have diameters of 0.5 inch, 0.375 inch and 0.25 inch. The tools are used in that order. The best machining time can be obtained with three tools is 1.0039 minutes. This data is presented in the output so that if the number of tools obtained in the best solution is not preferable, the user can choose the appropriate values. The details of tool usage follow these main results. They are arranged in the order of tool usage. The "Tool data" section presents the purpose of the tool, i.e., what sub-polygons it will rough and finish.

```
------ Tool Data Start -------
Selected_Tool =
  0.5000
Rough_Milled_Sub_polygons =
   1   2   3   4
Finish_Milled_Sub_Polygons =
   1
------ Tool Data Finish ------
```

For example, the details presented in the above section can be read as the tool with 0.5 inch diameter would be used for roughing sub-polygons 1, 2, 3 and 4 as well as finishing sub-polygon 1.

The "Best Option" section follows the detail results. This section presents the best possible option under given conditions. In the research examples presented, it would be the best time, as no weight on minimizing the number of tools has been specified. The addition of the weight on number of tools would evaluate the results on the combination of machining time and number of tools. This section has all the details about the number of tools and tool usage for convenience. The "Best Option" section also has one more important section presenting the machining sequence.

```
*********************************
Polygon_Machining_Sequence =
   1   2
*********************************
```

The sequence to be followed while machining the sub-polygons is presented in this section. This section follows the tool usage details in the "best Option" section.

6.2 Example pockets

6.2.1 Pocket with no islands

Figure 6.1 illustrates a pocket with no islands in it. When there is no island present in the pocket, the analysis skips the steps of island neck detection. The boundary necks are the only form of necks detected. The sub-division is completely based on the boundary necks in this case.

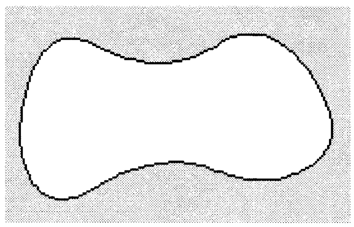

Figure 6.1: Pocket with no islands

Figure 6.2 illustrates the pocket after analysis and final set of tools selected.

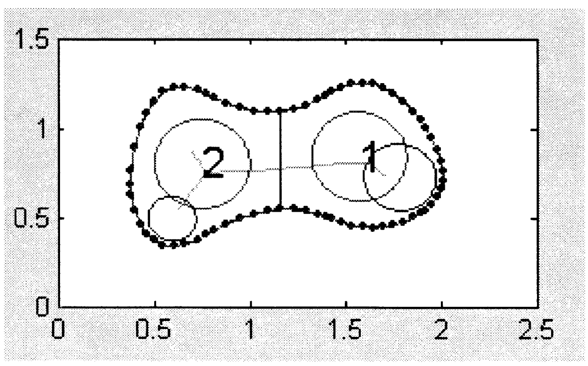

Figure 6.2: Analyzed pocket with no islands

The numbers inside the pocket represent the number of the sub-polygon. There are two types of circles in each sub-polygon representing the tool sizes. The large circles having sub-polygon number at their center are roughing tools while the small circles near the boundary are finishing tools. The line dividing the sub-polygons can also be seen.

The output results of the analysis (output of the research software on MatLab) for this example have been presented below as a sample. Tables 1 and 2 summarize the results for this example.

```
Maximum_No_of_Tools_can_be_used =

    3

==================================
Machining Time using Single Tool Approach
==================================

Tool_Diameter_Used =

   0.2500

Machining_Time =

   1.8525

==================================
Machining Times using Subdivision Approach
==================================
.........................................
*************************************
Number_of_Tools_Used =

    1

Tool_Diameters =

   0.2500

Machining_Time =

   1.7691
*************************************
-----------------------------
Details About Tool Usage
-----------------------------
------- Tool Data Start -------

Selected_Tool =

   0.2500

Rough_Milled_Sub_polygons =

    1    2

Finish_Milled_Sub_Polygons =

    1    2
------- Tool Data Finish ------
-----------------------------
.........................................
*************************************
Number_of_Tools_Used =

    2

Tool_Diameters =

   0.5000    0.2500

Machining_Time =

   1.0336
*************************************
-----------------------------
Details About Tool Usage
-----------------------------
------- Tool Data Start -------

Selected_Tool =

   0.5000

Rough_Milled_Sub_polygons =

    1    2
------- Tool Data Finish ------
------- Tool Data Start -------

Selected_Tool =

   0.2500

Finish_Milled_Sub_Polygons =

    1    2
------- Tool Data Finish ------
-----------------------------
.........................................
*************************************
Number_of_Tools_Used =

    3

Tool_Diameters =

   0.5000    0.3750    0.2500

Machining_Time =

   1.0039
*************************************
-----------------------------
Details About Tool Usage
-----------------------------
------- Tool Data Start -------

Selected_Tool =

   0.5000

Rough_Milled_Sub_polygons =

    1    2
------- Tool Data Finish ------
```

```
------- Tool Data Start --------

Selected_Tool =

  0.3750

Finish_Milled_Sub_Polygons =

   1
------- Tool Data Finish ------
------- Tool Data Start --------

Selected_Tool =

  0.2500

Finish_Milled_Sub_Polygons =

   2
------- Tool Data Finish ------
------------------------------
.............................................
############# Best Option ############

Number_of_Tools_Used =

   3

Tool_Diameters =

  0.5000   0.3750   0.2500

Machining_Time =

  1.0039
************************************
--------------------------
Details About Tool Usage
--------------------------
------- Tool Data Start --------

Selected_Tool =

  0.5000

Rough_Milled_Sub_polygons =

   1   2
------- Tool Data Finish ------
------- Tool Data Start --------

Selected_Tool =

  0.3750

Finish_Milled_Sub_Polygons =

   1
------- Tool Data Finish ------
------- Tool Data Start --------

Selected_Tool =

  0.2500

Finish_Milled_Sub_Polygons =

   2
------- Tool Data Finish ------
************************************
Polygon_Machining_Sequence =

   1   2
************************************
######################################
```

Simplified Results

Table 1 - Single Tool Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time
1	0.2500	1 2	1 2	1.8525

Table 2 - Sub-division Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time
1	0.2500	1 2	1 2	1.7691
2	0.5000 0.2500	1 2	 1 2	1.0336
3	0.5000 0.3750 0.2500	1 2	 1 2	1.0039

Polygon_Machining_Sequence = 1 2

No. of tools used = 3

% Saving in machining time = 45.81%

6.2.2 Pocket with one island

This section presents a pocket with one island inside it, as shown in figure 6.3.

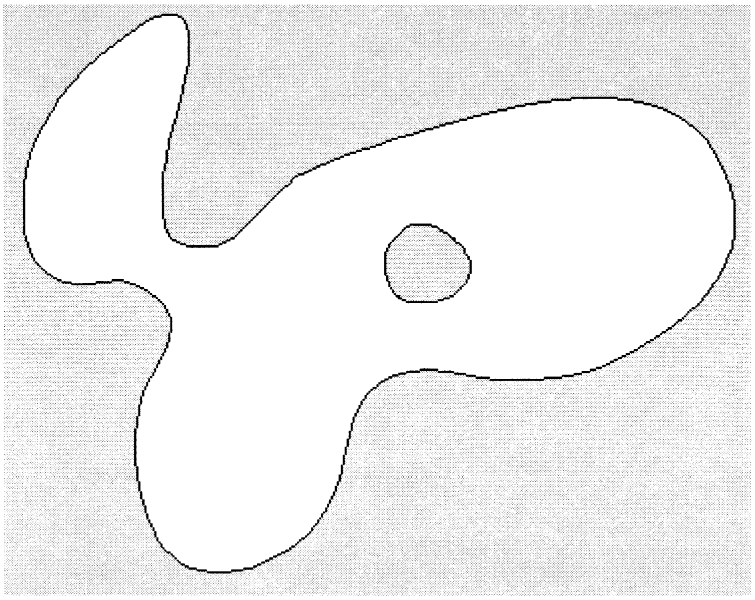

Figure 6.3: Pocket with one island

The presence of an island would require the algorithm to consider boundary and island necks during the sub-division process.

Simplified Results

Table 3 - Single Tool Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time
1	0.1250	1 2	1 2	15.9278

Table 4 - Sub-division Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time
1	0.1250	1 2 3 4	1 2 3 4	15.6778
2	0.5000 0.1250	1 2 3 4 	1 2 3 4	5.5488
3	0.5000 0.2500 0.1250	1 2 3 4 	1 2 3 4	4.8823
4	0.6250 0.5000 0.2500 0.1250	1 2 3 4 	1 2 3 4	4.2562
5	0.8750 0.6250 0.5000 0.2500 0.1250	2 1 3 4 	 1 2 3 4	4.4595

Polygon_Machining_Sequence = 1 3 4 2

No. of tools used = 4

% Saving in machining time = 72.85%

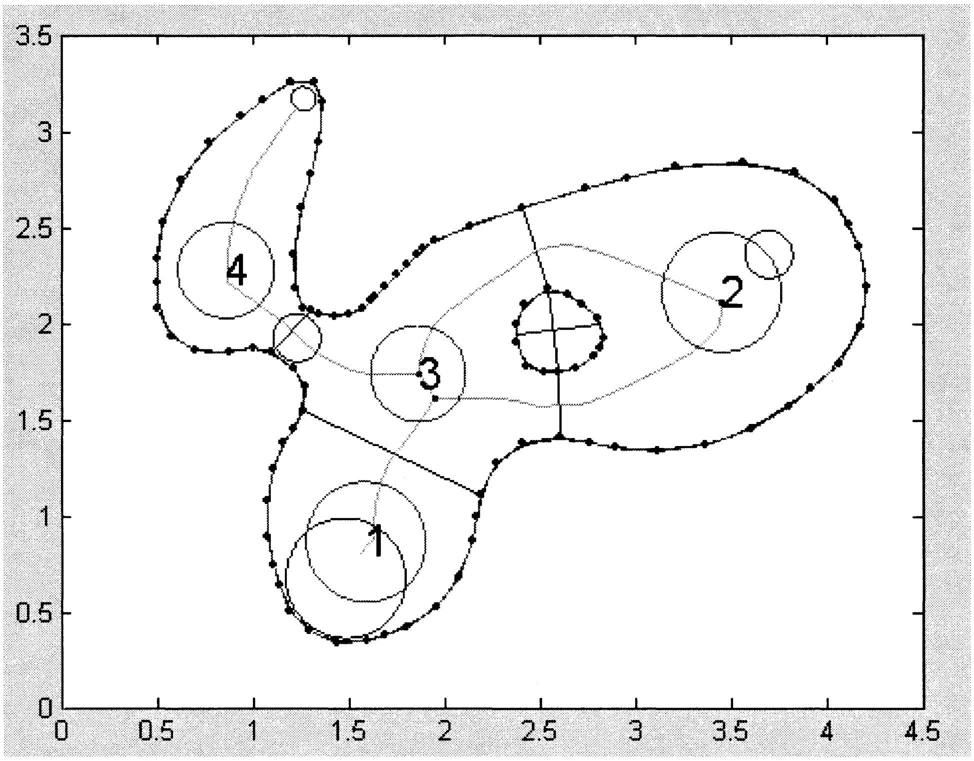

Figure 6.4: Analyzed pocket with one island

6.2.3 Pockets with two islands

There are three examples for pockets with two islands. Three examples have been chosen because of their unique features.

6.2.3.1 Simple pocket with two islands

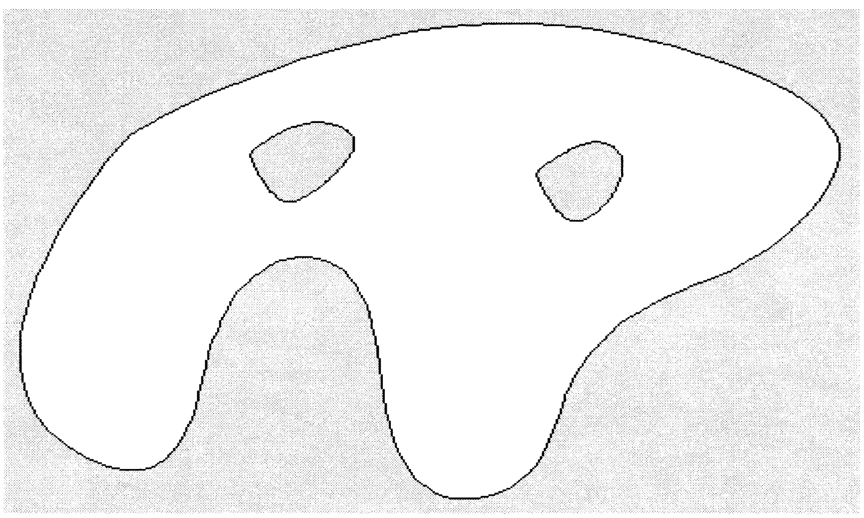

Figure 6.5: Simple pocket with two islands

This is an example of simple pocket with two islands. The analyzed pocket is illustrated in Figure 6.6.

Simplified Results

Table 5 - Single Tool Approach

Number of Tools Used	**Tool Diameter**	**Roughing Sub-polygons**	**Finishing Sub-polygons**	**Total Machining Time**
1	0.2500	1 2 3	1 2 3	11.8802

Table 6 - Sub-division Approach

Number of Tools Used	**Tool Diameter**	**Roughing Sub-polygons**	**Finishing Sub-polygons**	**Total Machining Time**
1	0.2500	1 2 3	1 2 3	11.7136
2	0.6250 0.2500	1 2 3	1 2 3	5.3305
3	0.8750 0.6250 0.2500	2 1 3	 1 2 3	5.6118

Polygon_Machining_Sequence = 1 2 3

No. of tools used = 2

% Saving in machining time = 55.13%

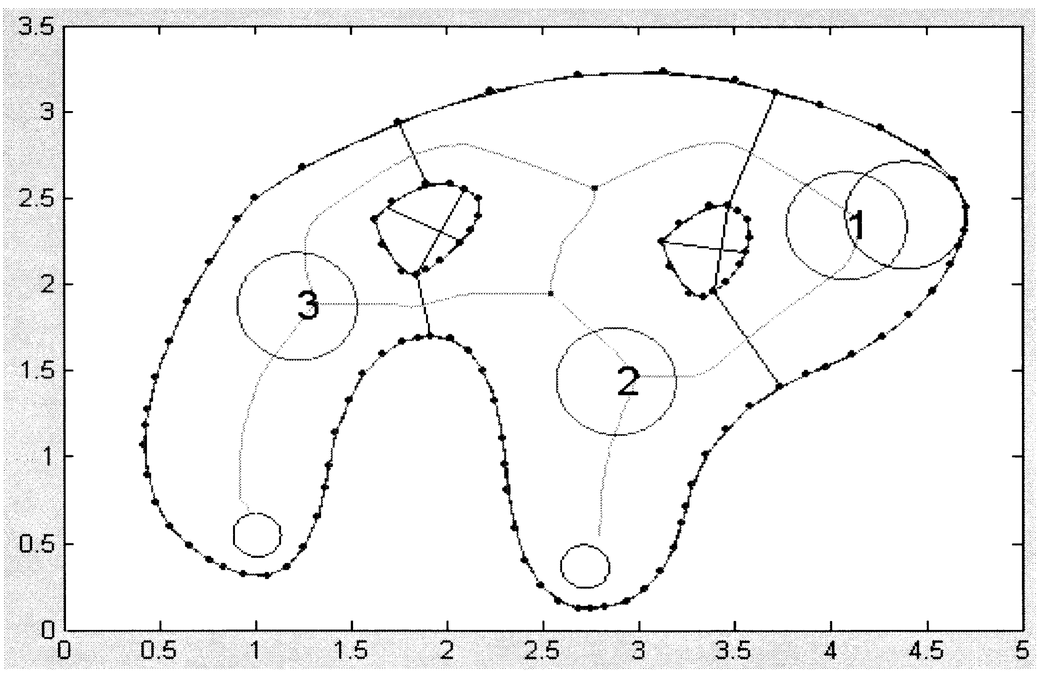

Figure 6.6: Analyzed simple pocket with two islands

6.2.3.2 Control Points specified in clockwise direction

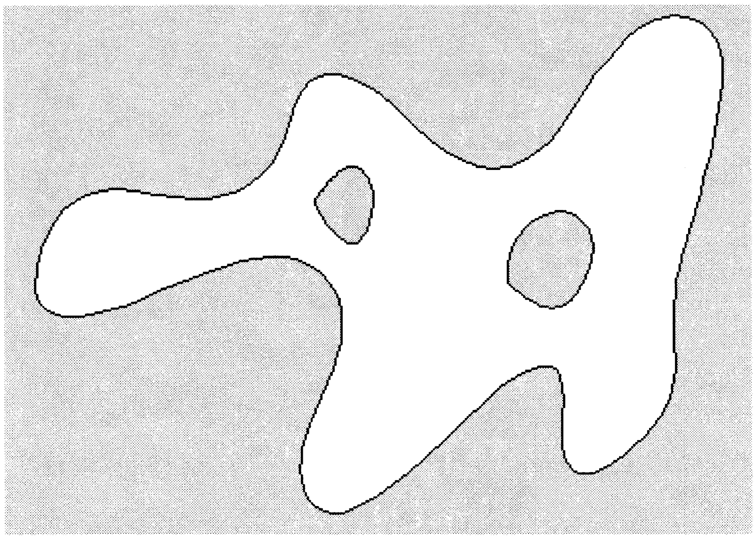

Figure 6.7: Pocket with Control Points in clockwise direction

The control points for this pocket have been specified in clockwise direction. The analyzed pocket is illustrated in Figure 6.8. In this example, the weighting factor for minimizing machining time is 0.75 whereas minimizing number of tools has a weight of 0.25.

Simplified Results

Weighting Factors

i) Minimizing Machining Time = 0.75

ii) Minimizing Number of Tools = 0.25

Table 7 - Single Tool Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time	Weighted Factor
1	0.1250	1 2 3 4	1 2 3 4	12.6086	9.7064

Table 8 - Sub-division Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time	Weighted Factor
1	0.1250	1 2 3 4	1 2 3 4	12.3586	9.5189
2	0.5000	1 2		10.2980	8.2235
	0.1250	3 4	1 2 3 4		
3	0.5000	1 2		10.1705	8.3778
	0.3750	4			
	0.1250	3	1 2 3 4		

Polygon_Machining_Sequence = 1 2 3 4

No. of tools used = 2 (Based on Weighted Factor)

% Saving in machining time = 16.68%

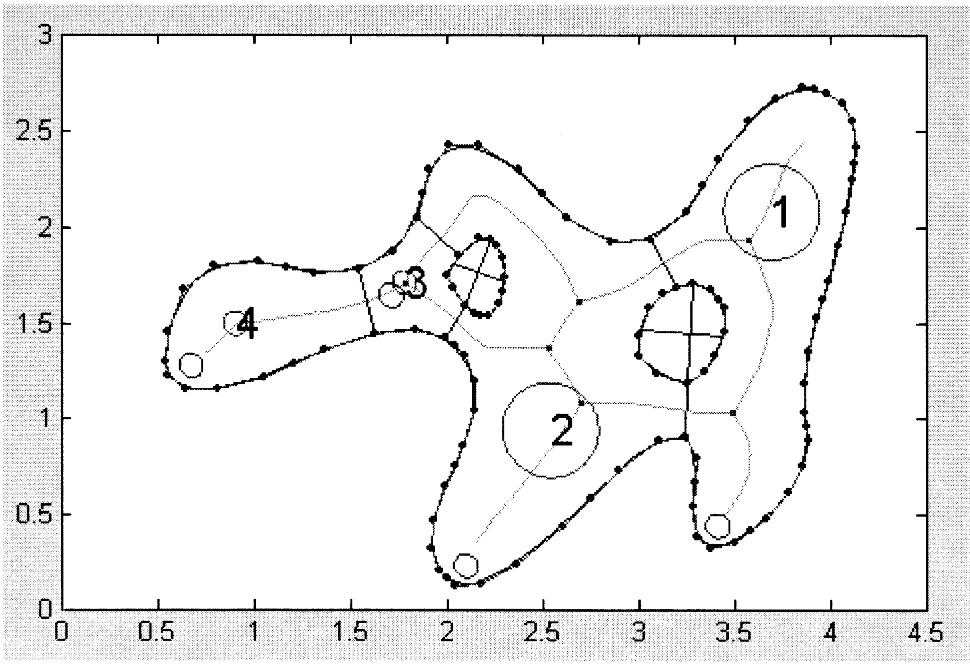

Figure 6.8: Analyzed pocket with clockwise control points

6.2.3.3 Pocket with many sub-polygons

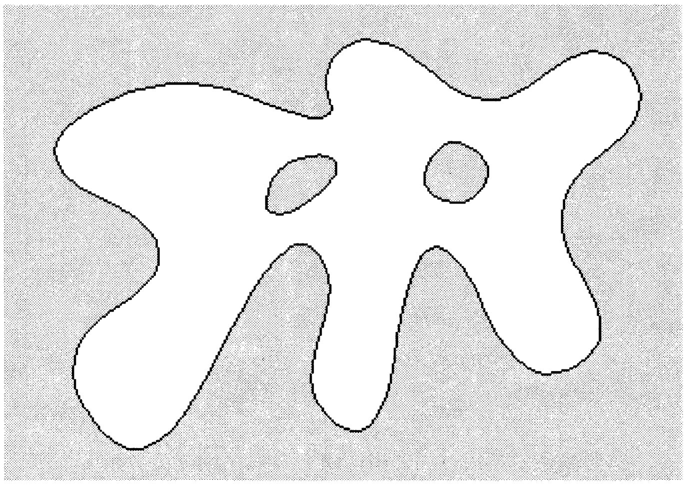

Figure 6.9: Pocket with many sub-polygons

This is a pocket with comparatively more sub-polygons. This example has been chosen specifically to present the results of the polygon sequencing, which is based on modified Sollins algorithm for minimum tree spanning. The analyzed pocket is presented below followed by results.

Simplified Results

Table 9 - Single Tool Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time
1	0.1250	1 2 3 4 5 6 7	1 2 3 4 5 6 7	11.3053

Table 10 - Sub-division Approach

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time
1	0.1250	1 2 3 4 5 6 7	1 2 3 4 5 6 7	10.8053
2	0.2500 0.1250	1 2 3 4 5 6 7	1 2 4 3 5 6 7	5.9621
3	0.3750 0.2500 0.1250	2 6 7 1 3 4 5	 1 2 4 3 5 6 7	5.1318
4	0.5000 0.3750 0.2500 0.1250	7 2 6 1 3 4 5	 1 2 4 3 5 6 7	4.9733

Polygon_Machining_Sequence = **1 5 4 6 7 2 3**

No. of tools used = 4

% Saving in machining time = 56%

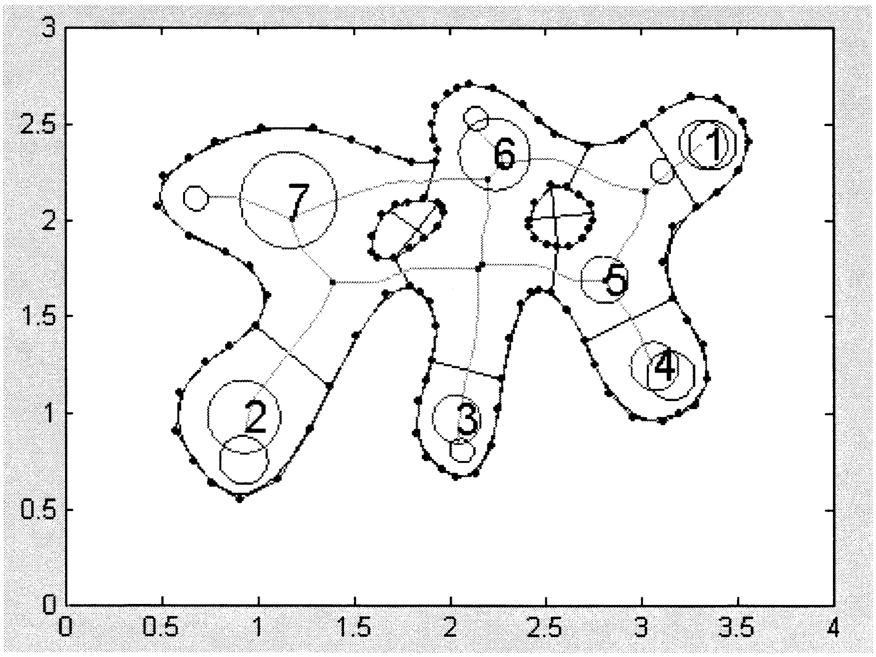

Figure 6.10: Analyzed pocket with many sub-polygons

6.3 Comparison with Multi-tool method

This section shows the comparison of this research methodology with a multi-tool methodology presented by Yang et al. [23]. The method by Yang determines the effective area for the selected tools and designs a zigzag path in that area.

Figure 6.11 shows the example chosen for the comparison. The results of the implementation of the method by Yang are presented below.

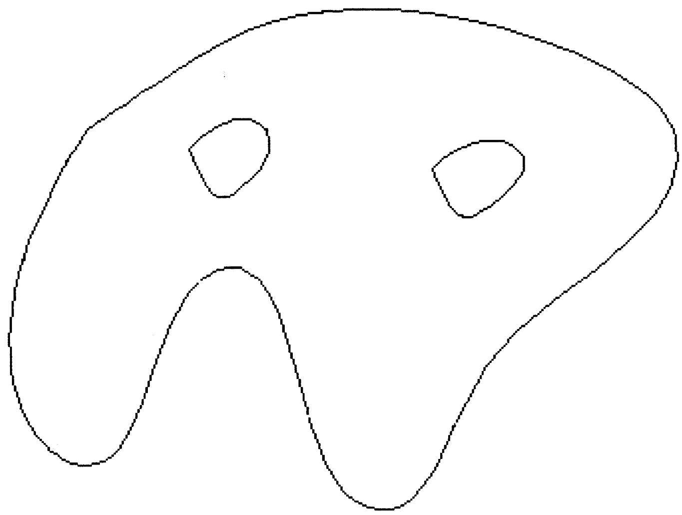

Figure 6.11: Multi-tool method example

The largest tool that can fit inside the pocket is 1.25 inches whereas the largest tool that can machine the entire pocket is 0.25 inches. So, the tools selected in this method are 0.25, 0.50, 0.75 and 1.00 inch. These tools are selected as suggested in the methodology.

Figure 6.12 shows an example of effective areas when selected tools are 1.00 and 0.25 inch.

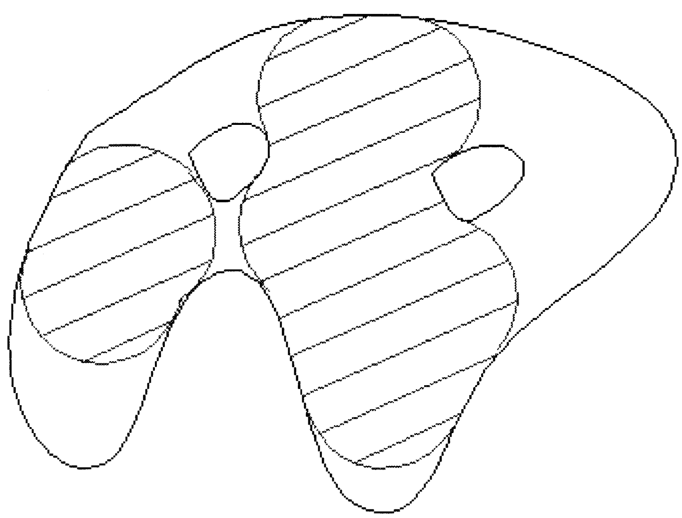

Figure 6.12: Effective areas of tool

The hatched part of the pocket shows the effective area of a 1 inch tool. The rest of the part of the pocket is machined using a 0.25" tool. The machining time is calculated using a zigzag path in the effective areas of the tools.

Similarly, the effective area for each tool in the combination and its zigzag path is used for the calculation of the machining time.

The approximate machining times with this method are

Table 11: Machining times using Yang's method

Tools	Machining Time (Minutes)
0.50, 0.25	8.937 + 1.010 = 9.947
0.75, 0.25	4.075 + 4.122 = 8.227
1.00, 0.25	1.756 + 8.669 = 10.425
0.75, 0.50, 0.25	4.075 + 1.006 + 1.221 = 6.302
1.00, 0.75, 0.25	1.756 + 1.492 + 4.973 = 8.221
1.00, 0.50, 0.25	1.756 + 4.311 + 1.010 = 7.077

The best option with this method is to use the 0.75, 0.5 and 0.25 inch tools. The optimal machining time is 6.302 minutes.

Next, the results of implementing the methodology proposed in this thesis are presented below.

Table 12: Machining times by subdivision method

Number of Tools Used	Tool Diameter	Roughing Sub-polygons	Finishing Sub-polygons	Total Machining Time (Minutes)
1	0.2500	1 2 3	1 2 3	11.7136
2	0.6250 0.2500	1 2 3 	1 2 3	5.3305
3	0.8750 0.6250 0.2500	2 1 3 	 1 2 3	5.6118

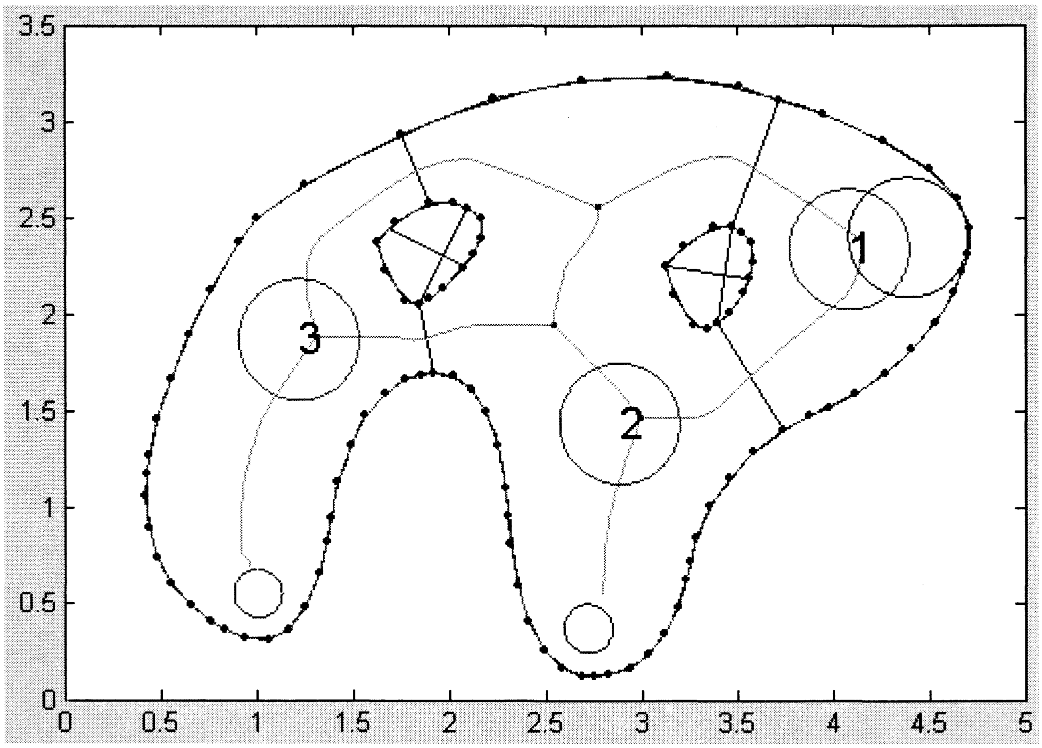

Figure 6.13: The subdivision resultant pocket

According to the new methodology, the best combination of the tools is 0.625 and 0.25 inches. The optimal machining time is 5.3305 minutes, which is 15.4% less than the machining time obtained using the methodology presented by Yang. The number of tools is also less than the compared method.

This comparison suggests that the results obtained by the new methodology are better than the method proposed by Yang. The methodology by Yang requires multiple plunges when the selected tool has disjointed effective areas as in Figure 6.12. The new methodology requires only one plunge.

This chapter presented the results obtained for different examples of the pockets. The software designed from the algorithm accepts the control points and depth of pocket as the inputs to obtain the tool sizing and sequencing data. The next chapter will present the conclusions based on the results as well as the future direction of the work.

CHAPTER 7 – CONCLUSIONS AND FUTURE WORK

The polygon subdivision approach for pocket machining is a powerful method for optimal tool selection and process planning. A traditional single tool approach is not very economical for machining time especially with the advent of rapid tool changers. The results of this research suggest that the polygon subdivision and use of multiple tools can significantly improve the machining time. The percentage saving in the machining time when compared to the single tool approach can be as high as 75%.

The approach of this thesis has many advantages and is very flexible. The input parameters are control points and the depth of the pocket. The control points can be easily obtained from any CAD software and depth of pocket is always available as a design parameter. The algorithm has a provision for changing the weight on the machining time and number of tools according to the requirements of the conditions. The polygon machining sequence, which is generated as one of the outputs, would eliminate the requirement of more than one plunging. This polygon machining sequence can be followed using any desired toolpath design procedure.

This research has covered all the main parts of this methodology of polygon subdivision, tool selection and tool sequencing. In polygon subdivision, the pocket is analyzed for the best positions for the neck for efficient tool selection. The tool selection procedure determines all possible options of the tools and selects the best option. The last step, i.e., the tool sequencing process, determines the best sequence of the polygons to machine.

Although the current approach generates data for most of the pockets there are some opportunities for improvement. The current neck detection algorithm is based on determination of the narrow regions formed by the two non-convex regions. There is a possibility of neck formation by combination of convex and non-convex regions as illustrated in Figure 7.1.

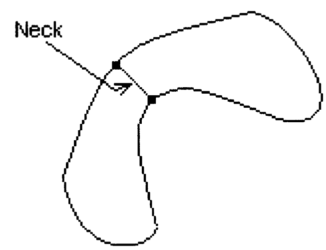

Figure 7.1: Neck between convex and non-convex regions

In the future, the approach can be extended to determine the necks formed by such combination of convex and non-convex regions. The present algorithm calculates the machining time based on area. Although the area based calculation is good approximation, they may not be very accurate. The next version of the system should use better machining time calculation algorithm. The current system assumes that the bottom of the pocket is a linear 2D plane. It is expected that the future algorithm would accommodate the pockets with non-linear 3D bottom surfaces. The future algorithm is also expected to subdivide the non-convex islands. Lastly, this algorithm can be implemented in a better CAD environment to make it more user friendly. Also the implementation of the algorithm in a CAD environment will make it easier to generate automatic instructions for the CNC machine.

REFERENCES

1. Hansen, A., Arbab, F., An for algorithm arbitrarily for generating NC tool paths shaped pockets with islands, ACM Transactions on Graphics, Vol. 11, No. 2, April 1992, p 152-182
2. Persson H., NC machining of arbitrary shaped pockets. Computer Aided Design, 1978, 10(3), p 169-174
3. Chuang, S. H.; Lin, W.S. Tool path generation for pockets with free form curves using Bezier convex hulls, The International Journal of Advance Manufacturing Technology, Vol. 13, n 2, 1997, p 109-115
4. Jeong, J., Kim, K., Generation of tool paths for machining free form pockets with islands using distance maps, The International Journal of Advance Manufacturing Technology (1999) Vol. 15, p 311–316
5. Jeong, J., Kim, K., Generating tool paths for free-form pocket machining using z-buffer-based Voronoi diagrams, The International Journal of Advance Manufacturing Technology (1999) Vol. 15, p 182–187
6. Held, M., Lukacs, G., Andor, L., Pocket machining based on contour-parallel tool path generated by means of proximity maps, Computer Aided Design, Vol. 26, n 3, Mar, 1994, p 189-203
7. Held, M., Geometry-based investigation of the tool path generation for zigzag pocket machining, Visual Computer, Vol. 7, n 5-6, Sep, 1991, p 296-308
8. Veeramani, D., Gau, Y., Cutter-path generation using multiple cutting-tool sizes for 2-1/2D pocket Machining, IIE Transactions (2000) Vol. 32, p 661-675
9. Hinduja, S., Roaydi, A., Philimis, P., Barrow G., Determination of optimal cutter diameter for machining 2 ½ O pockets, International Journal of Machine Tools & Manufacture, Vol. 41, (2001), p 687-702

10. Lim, T., Corney, J., Ritchie, J. M., Clark, D. E. R., Optimising automatic tool selection for 2 ½ D components, Proceedings of DETC'00: ASME 2000 Design Technical Conference and Computers and Information in Engineering Conference, 2000

11. Lim, T., Corney, J., Clark, D. E. R., Exact tool sizing for feature accessibility, The International Journal of Advance Manufacturing Technology (2000) Vol. 16, p 791 – 802

12. Manuel, D., Liang, M., Kolahan, F., Dynamic offsetting approach to tool path generation for machining convex polygons, Computers and Industrial Engineering., Vol. 31, No. ½, p 135 – 138, 1996

13. Tiller, W., Hanson, E. G., Offsets of two-dimensional profiles, EG IEEE Computer Graphics and Applications, Vol. 4, p 36-46, 1984

14. Bohm, W., Generating the Bezier points of B-spline curves and surfaces, Computer Aided Design. Vol. 13, p 365-366. Nov. 1981

15. Boehm, W., Inserting new knots into B-spline curves. Computer-Aided Design. Vol. 12, p 199–201, 1980

16. Frank S. H. C., C. Z. Kao, One-sided arc approximation of B-spline curves for interference-free offsetting, Computer-Aided Design, Vol. 31 (1999), p 111-118

17. Ge, Y., Fitzpatrick, M. J., On the generation of skeletons from discrete euclidean distance maps, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, no. 11, Nov. 1996

18. Liu, Q., A recursive algorithm for triangulation of arbitrary polygon based on BSP tree, IAPRS, Vol. XXXIV, Part 2, Commission II, Xi'an, Aug.20-23, 2002

19. Zelinski, P., Curvilinear tool paths for pocket machining, MMS Online, http://www.mmsonline.com/articles/0702rt1.html, July 2002

20. Arya, S., Cheng, S. W., Mount, D. M., Approximation algorithms for multiple-tool milling, Proceedings of the fourteenth annual symposium on computational geometry, 1998, p 297-306

21. Ahuja, R. K., Magnanti, T. L., Orlin, J. B., Network Flows, Theory, Algorithms and Applications, Prentice Hall

22. Walsh, Ronald A., Handbook of machining and metalworking calculations, McGraw-Hill, 2001

23. Yang Z., Joneja A., Zhu S., Recognizing generalized pockets for optimizing machining time in process planning – Part 2., International Journal of Production Research , 2001, Vol. 39, No 16, p 3601-3621