Int J Adv Manuf Technol (2016) 83:407–420
DOI 10.1007/s00170-015-7506-3

ORIGINAL ARTICLE

An automated CNC programming approach to machining pocket with complex islands and boundaries by using multiple cutters in hybrid tool path patterns

Min Zhou[1] · **Guolei Zheng**[1] · **Zezhong Chevy Chen**[2]

Received: 18 October 2014 / Accepted: 26 June 2015 / Published online: 25 July 2015

Abstract To enhance the efficiency of pocket machining, considerable researches on strategies of tool selection and tool path optimization have been conducted. However, few studies are devoted to tool path planning with consideration of hybrid tool path patterns and multi-cutter selection. This research gap leads to little promotion to industrial application of the available research results. This paper proposes an automated CNC programming approach considering both multiple cutters and hybrid tool path patterns for pockets machining with complex islands and curvilinear boundaries. Firstly, the largest cutter capacity (LCC) of a pocket, which restricts essentially the pocket machining time, is investigated and defined. Secondly, to enlarge the LCC and to generate optimal tool path more easily, the pocket is split into many sub-regions according to the bottleneck lines of its boundary. After subdivision, tool selection and path pattern determination rules for each sub-area are introduced, respectively. Then, several heuristic principles for region recombination are developed to minimize the number of sub-polygons. Besides, based on the soft edges of each sub-area, expansion technique is employed to generate machining bounds for residuals. Finally, to demonstrate the advantages of this approach, two examples are rendered. And, the results show that the presented method performs better than the conventional ways.

Keywords Pocket milling · CNC programming · Polygon decomposition · Multiple tool selection · Tool path planning

Nomenclature

Ω	The connected machining area of a pocket
B_p	The boundary of a pocket
B_i	The boundaries of islands in a pocket
B_Ω	Boundaries of Ω, consisting of B_p and B_i
A_Ω	Area of Ω
P_B_p	Polygon representing B_p
P_B_i	Polygon representing B_i
$\boldsymbol{p}$	Vector
t_m	Machining time
t_{mb}	Machining time before subdivision
t_{ms}	Machining time after subdivision
BnL	Bottleneck line
l_b	Length of BnL
$\overrightarrow{p_i p_{i+1}}$	Vector from start point p_i to endpoint p_{i+1}
$\overset{\frown}{p_{i-1} p_i}$	Circular arc with endpoints p_i and p_{i+1}
MIC	The maximum inscribed circle
R_{MIC}	The radius of a MIC
$\boldsymbol{n}_L$	The direction along the longer edge of a rectangle
D_m	Diameter of the selected main tool for a sub-region
R_c	The acceptable corner radius of a pocket

1 Introduction

One of the most common machining operations in the manufacturing process of aircraft structural part and other mechanical part is pocket milling. Generally, the pocketing process is arranged in two phases named rough machining and finish machining, respectively. In rough machining, most of the materials are removed from the blank using a large tool.

✉ Min Zhou
zhoumin.buaa@139.com

[1] School of Mechanical Engineering and Automation, Beihang University, Beijing, China

[2] Department of Mechanical and Industry Engineering, University of Concordia, Montreal, Quebec, Canada

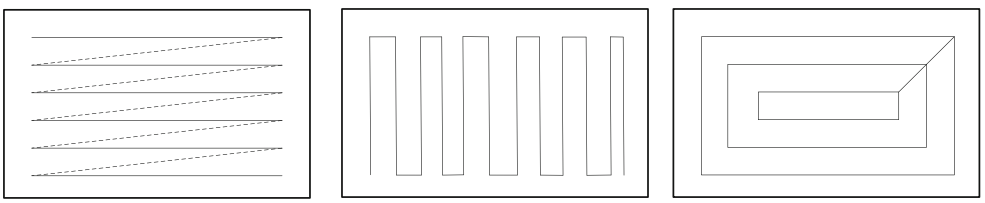
In finish machining, the uncut areas left by the roughing are cleared applying a small tool with acceptable corner radius to ensure machining precision. Therefore, pocket machining efficiency depends largely on the roughing's material removal rate and machining time. Conventionally, in rough machining, only a single tool is employed to generate the cutter path for a pocket. When narrow bottlenecks exist in the pocket, the diameter of the selected tool must be smaller or equal to the bottleneck width to avoid gouging, which may increase the machining time considerably. With the appearance and development of rapid automatical tool change technology, the tool change time is no longer a concern and multiple cutting tools are adopted to promote pocketing efficiency. Large tool can be used for areas with simple geometrical shape to remove most of the materials because of its large material removal rate. Oppositely, cutter with a small diameter can be applied to clear the complex portion like sharp corners and narrow bottlenecks due to its higher accessibility. Therefore, the use of multiple tools of distinct diameters makes it possible to advance the machining efficiency as well as to guarantee the geometric accuracy. Besides, the tool path pattern is another vital factor which influences the machining time greatly. As shown in Fig. 1, direction tool path and contour path pattern are two types of widely used tool path patterns. We can see that for direction tool path, different sweeping direction may lead different tool path lengths. Moreover, the research of Yao and Gupta [1] indicates that depending on the features of a pocket, tool paths generated by a single path pattern may not be efficient for the whole pocket. Thus, studies on multiple tool path patterns also deserve much concern. However, for pocket with complex geometrical shape, it is rather difficult to determine the optimal tool set. Furthermore, even if the tools have been selected and the related machining area for each cutter has been determined, it is still challenging to optimize the tool path for the whole pocket.

1.1 Literature review

In recent decades, a significant amount of researches in the area of machining strategy, such as optimal tool selection and tool path planning, has been reported for 2D pocket machining.

1.1.1 Multiple cutter selection

Most of the works on multiple cutter selection are built on geometrical method with certain optimal algorithm. The approach of Bala and Chang [2] to selecting tools is restricted to two tools. The smaller one's radius is equal to fillet radius at corners, and the larger one is to make sure that the material left by the smaller cutter at each of the convex vertices can be removed by one pass along the boundary of the finishing cutter. Without doubt, this solution would cause poor machining efficiency when a number of narrow bottlenecks or sharp corners exist. In the view of Lee et al. [3], two cutters in roughing can ensure efficient machining. Where, the larger one is employed to the simple areas and the smaller one is for the complex parts like bottlenecks. However, the number of optimal tool combination may exceed two. More than two cutters are determined for rough machining with genetic algorithm [4, 5], dynamic programming approach [6, 9], and other optimization algorithms for optimal tool set selection. The method proposed by You et al. [8] builds on the following hypotheses: for a pocket with no residue, the optimal number of tools is one; for one with local residues, the number may be one or two; and for one with global residues, it is less than four. Without consideration of tool path, Nadjakova and McMains [10] presented an algorithm to generate an optimum sequence of cutter radii for machining a 2D pocket. In their opinions, no restrictions on cutter size or number are preset. Namely, according to their assumption, the number of selected optimum cutters may exceed four. Makhe and Frank [11] subdivided the pocket into smaller sub-polygons based on an approximate polygon subdivision technique to improve tool selection. They selected three tools for each sub-polygon and then determined the final three tools for the entire pocket using a branch and bound approach based on machining time. In addition, to eliminate the number of plunges, they applied a minimum tree spanning algorithm to sequence the sub-polygons to be machined. However, their work is restricted to convex island and ignores the bottlenecks between islands. And, their sequencing approach is only according to the sub-polygons' neighboring relations without considering the selected tools, which may increase the tool change time. Ramaswami et al. [9] provided two methods to decompose the pocket geometry into convex regions and milled each region



(a) zag (b) zigzag (c) contour parallel

Fig. 1 Direction tool path (**a**, **b**) and contour path pattern (**c**)

independently by selecting a sequence of tools based on the accessibilities of various tools to the region. They first selected two cutters for each sub-region independently based on the accessibilities of various tools. Then, they determined the final optimal set of tools with a dynamic approach to minimize the total processing time including tool change time, rapid traverse time, and tool approach and retraction time. According to their algorithm, the number of applied tools for a pocket is always over 5 in their examples, which is a little unreasonable in industry. Besides the approaches for optimal set of tools, to assist and optimize the selection of multiple tools for the machining of complex components, the exact tool accessibility and the coverable area of a given diameter of milling cutter are also concerned by Zhang and Ge [12] and Lim et al. [13].

1.1.2 Minimum bottleneck width computation

No matter whether of using a single tool or multiple tools, the pocket geometry, especially the minimum bottleneck width of the pocket boundary, is a fundamental constraint to cutter selection. Lee and Chang [14] calculated the minimum distance between the island and the pocket boundary as the diameter of the roughing tool when an island is represented by its convex hull. Veeramani and Gau [6, 7] developed the Voronoi mountain of a pocket boundary polygon to calculate the bottleneck width. They also applied a dynamic programming approach for optimal tool radii selection. However, their technique does not serve to open pocket. To solve this problem, Yu et al. [15] revised the domain of a Voronoi mountain to establish the 45° draft of a profile and constructed the soft edge for open pocket to calculate the minimum passage width and to optimize procedure of automatic cutter selection. Lai et al. [16] presented incremental algorithm of the Voronoi diagram technique to calculate the offset distance and minimum passage width for tool path planning. They found all the passage width by enlarging/shrinking border and island contours and detecting the first point that meets the enlarged/shrunk contours. When there is no island in the pocket, monotonous areas are used to find the width. Yao et al. [17, 18] described a geometric algorithm for finding the largest feasible cutter. Their offsetting method is based on dividing pocket into target regions and obstruction regions. The method is also applicable for pocket with open edges. Chen and Zhang [19] determined the largest allowable size for the cutter to move along all the profiles (including NURBS curves) for 2½-axis finish machining without global and local gouging by applying the particle swarm optimization method. Han et al. [20] formulated a mathematical representation for the geometric model of the cutter interfering the impeller and established an optimization model of the cutter size to decide the largest tool for multi-axis roughing free-form surface impeller channel.

1.1.3 Tool path combination

Combined tool path means more than one path pattern or more than one cutting direction are adopted in a pocket. Compared to plentiful studies on tool selection, there are fewer researches on tool path combination. Park and Choi [21] selected the inclination for direction-parallel by finding an angular range belonging to the smallest number of tangent ranges of reflex vertexes. But, it is not optimal for a whole pocket with complex geometry to adopt only one sweeping direction. Vosniakos and Papapanagiotou [22] provided a scheme to machine a convex pocket without island by using hybrid contouring–staircasing pattern with three tools. Contouring with two cutters aims at clearing enough space around the boundary to allow the last one to implement efficient staircasing in the interior. Kim and Choi [23] compared the machining efficiency of three types of directional parallel tool paths (one way path, pure zigzag path, and smooth zigzag path) and contour parallel path by their proposed model which considered the acceleration and deceleration of the CNC machines. And, the smooth zigzag path is revealed as the most efficient tool path pattern by their methodology. Yao and Gupta [1] analyzed various kinds of tool path patterns systematically and discussed several existing heuristics for selecting cutter path patterns in detail. They concluded that depending on the features of a pocket, tool paths generated by a single machining strategy may not be efficient for the whole pocket. Therefore, they described a new cutter path generation algorithm by using different patterns in different regions of the geometry and seamlessly morphing them together. The proposed solutions are superior to those generated by any single pattern for complex pockets. However, this algorithm is only fit to single cutter and pocket without islands. Yao [24] introduced a novel cutter path planning approach to high-speed machining (HSM) which requires few sharp turns by employing a set of modified spiral curves based on the geometry of the 2D region. In his algorithm, the spiral curve centers at line segments. And, the two endpoints of each line segment are the branch points of the 2-D region's medial axis. However, the currently used greedy algorithm in finding the spiral curve segments cannot guarantee the optimal path. Ramaswami et al. [9] calculated the tool paths for each region independently after the pocket being decomposed. The tool paths are classified as main pass machining, triangular corner machining, polygonal corner machining, and final pass. However, tool paths at triangular and polygonal corners after main pass in each sub-region are lack of continuity extraordinarily, which may result in a great deal of lifting and repositioning time. Additionally, the decomposition with split edges produces redundant sharp corners and the adopted staircase milling strategy may cause residual areas (shown in Fig. 2). To satisfy the demand of aggressive roughing of a pocket, Chen and Fu [5] investigated an optimal approach to multiple tool selection and their numerical control path generation based on the medial axis transform of the pocket.

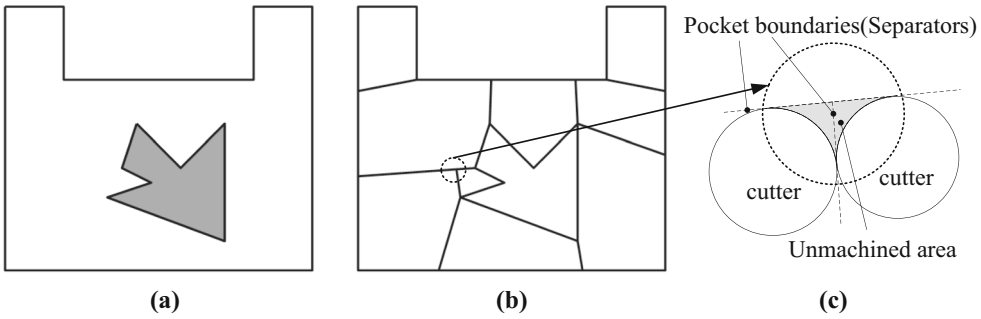


Fig. 2 Decomposition and residual area. **a** Geometry of polygonal pocket with island. **b** Decomposition with split edges. **c** Residual area

1.2 Overview

Though numerous studies have concentrated on tool selection and tool path determination for pocket machining, rather less attention has been paid to planning the tool path while multiple cutters are employed. However, study on tool path arrangement with multiple cutters is very significant for the using of previous research results. Thus, this paper focuses on tool path arrangement with multiple cutters to raise machining efficiency by pocket split and recombination. First, the essential factor restricting the pocket machining time is investigated. And, a new concept of the largest cutter capacity (LCC), a crucial attribute of a pocket, is presented. To enlarge the LCC, the pocket is divided into multiple sub-regions based on bottleneck lines. Due to the subdivision, cutter selection rules are established and different tool path patterns are chosen for sub-regions. Furthermore, to decrease non-cutting time caused by repositioning move and retraction, sub-region recombination is studied. In the process of recombination, soft edge of each sub-polygon is utilized fully. Finally, tool path for the whole pocket is generated by CATIA's numerical control (NC) machining module.

The rest of this paper is organized as follows: The concept of LCC and the technique of pocket division based on bottleneck line are introduced in Sect. 2. Section 3 introduces the principles for tool selection and tool path pattern determination. Based on the selected cutter and tool path pattern, heuristic rules for sub-region combination and expansion are established in Sect. 4. In Sect. 5, implementation process of this approach is given, and two examples are rendered to demonstrate the advantages of this presented approach over the conventional method. The final section is conclusion.

2 Cutter capacity and pocket division

2.1 Cutter capacity

In general, there are abundant factors affecting machining efficiency, such as cutter size, tool path, retraction and approach time, and tool change time if more than one tool is used. However, the selected tool sizes play a decisive role in obtaining short machining time since another important factor tool path length also depends on the selected tools. Suppose only one cutter is used to machine the whole pocket, generally, the expected diameter of the cutter is the largest possible without gouging or interference. However, the largest diameter depends on the geometry of the pocket and the acceptable residue at corner. Therefore, we consider the diameter of the largest tool used to remove all the material as a crucial attribute of a pocket. We call it the LCC of a pocket in this paper.

Definition 1: the largest cutter capacity LCC of a pocket is the diameter of the largest cutter used to remove the whole machining volume of the pocket with consideration of the acceptable corner radius and without gouging.

For example, the R_c of a rectangular-shaped pocket shown in Fig. 3a is 5. So, the LCC is φ10. Another example, the minimum bottleneck width of the pocket shown in Fig. 3b is 8 while its R_c is also 5. Then, the LCC is φ8.

We can see that the factors restricting the LCC of a pocket are the acceptable corner radius and the minimum bottleneck width of Ω. The former is called local restriction while the latter is global restriction in this paper. Local restriction is ignored when LCC is discussed and left to finishing. The following work is an attempt to eliminate global restriction.

Obviously, given $A_{\Omega 1}=A_{\Omega 2}$, if $LCC_1>LCC_2, t_{m1}<t_{m2}$ on condition that $D_1=LCC_1, D_2=LCC_2$ and other cutting parameters are the same, where D is the diameter of the cutter used to remove the whole machining volume of the pocket. Therefore, enlarging the LCC is one of the most effectual strategies to enhance the efficiency. Decomposition of a pocket can make it true with multiple tools. As shown in Fig. 4a, the LCC of the pocket is enlarged to φ30 if its four corners (gray areas in

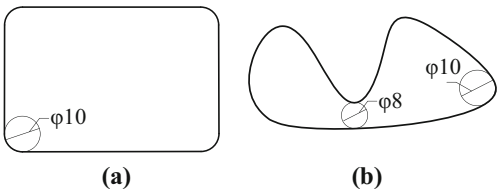


Fig. 3 LCC of pocket. **a** LCC with φ10. **b** LCC with φ8

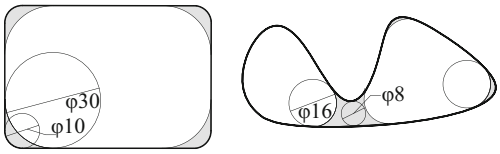


Fig. 4 Illustration of enlarged LCC of pocket. **a** LCC with φ30. **b** LCC with φ16

Fig. 4) are cleared by a smaller tool. And, the LCC of the pocket shown in Fig. 4b is increased to φ16 while its bottleneck area and corners are machined by tool φ8. Suppose a pocket with LCC_0 is partitioned into n sub-areas, in another word, A_Ω is subdivided into $A_{\Omega 1},\ldots, A_{\Omega n}$, and their LCCs satisfy $LCC_1 \le LCC_2 \le,\ldots, \le LCC_n$. When retraction and approach time are pushed aside temporarily, the following reasoning is exactly valid.

$$\because t_{mb} \propto A_\Omega / LCC_0$$

$$t_{ms} \propto A_{\Omega 1}/LCC_1 + A_{\Omega 2}/LCC_2 + \ldots + A_{\Omega n}/LCC_n$$

$$A_\Omega = A_{\Omega 1} + A_{\Omega 2} + \ldots + A_{\Omega n}$$

$$LCC_1 = LCC_0$$

$$LCC_1 \le LCC_2 \le, \ldots, \le LCC_n$$

$$\therefore t_{mb} \propto A_{\Omega 1}/LCC_1 + A_{\Omega 2}/LCC_1 + \ldots$$

$$+ A_{\Omega n}/LCC_1 \ge A_{\Omega 1}/LCC_1 + A_{\Omega 2}/LCC_2 + \ldots$$

$$+ A_{\Omega n}/LCC_n$$

$$\therefore t_{ms} \le t_{mb}.$$

Hence, proper division to the pocket is helpful to achieve high machining efficiency.

2.2 Pocket division based on bottleneck line

Besides the limitation of the acceptable corner radius, LCC of a complicated pocket is rather bounded by bottlenecks of the machining area's border. Consequently, in order to eliminate the global restrictions and enlarge the pocket's LCC, the machining area can be subdivided into several sub-regions on the basis of the bottlenecks. To partition a pocket properly, reflex points are defined and found firstly. Based on the reflex points, bottleneck lines of a pocket are decided, and then, pocket is divided.

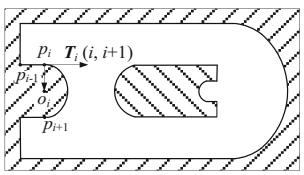


Fig. 5 The tangent vector of p_ip_{i+1} at point p_i

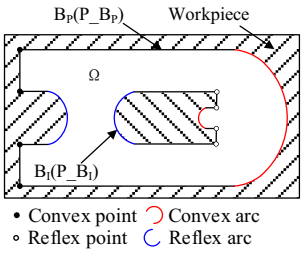


Fig. 6 Illustration of reflex points and reflex arcs on pocket boundaries

2.2.1 Reflex point determination

A point may be reflex for a limitary region and convex for its neighboring region. For generalized polygon including circular arcs, besides the vertices, arcs are also convex or reflex. To determine whether a vertex/an arc is reflex or not, cross product is applied. Suppose serial vertices $p_1, p_2, \ldots, p_i, \ldots, p_n$ are in a counterclockwise sequence on P_B_p and in clockwise direction on P_B_i.

For vertex connecting two line segments, suppose vectors $\overrightarrow{p_{i-1}p_i} \times \overrightarrow{p_ip_{i+1}} = \boldsymbol{p}_i$, based on the right-hand rule,

1. if vector $\boldsymbol{p}_i$ is outside, p_i is a convex point;
2. if vector $\boldsymbol{p}_i$ is inside, p_i is a reflex point;
3. if vector $\boldsymbol{p}_i = \boldsymbol{0}$, p_i is a tangent point.

For vertex connecting two circular arcs or connecting one line segment and a circular arc, vector $\overrightarrow{p_ip_{i+1}}$ in above equation should be replaced by $\boldsymbol{T}_i$ ($\underline{i}$, i+1), the tangent vector of $\overset{\frown}{p_ip_{i+1}}$ at point p_i. And, the direction of $\boldsymbol{T}_i$ ($\underline{i}$, i+1) is in line with the direction of B_Ω (shown in Fig. 5). If p_i connects two arcs, $\overrightarrow{p_{i-1}p_i}$ should also be replaced by $\boldsymbol{T}_i$ (i-1, $\underline{i}$), the tangent vector of $\overset{\frown}{p_ip_{i+1}}$ at point p_i.

For an arc, cross product $\mathbf{T}_i \times \overrightarrow{p_io_i}$ is employed, where o_i is the centre of an arc. If the result of $\mathbf{T}_i \times \overrightarrow{p_io_i}$ is outside, $\overset{\frown}{p_ip_{i+1}}$ is a convex arc. And, if it is inside, $\overset{\frown}{p_ip_{i+1}}$ is a reflex arc. Obviously, if B_p is a circle, the whole circle is a convex. If B_i is a circle, the whole circle is a reflex arc. Figure 6 illustrates reflex points and reflex arcs of a pocket's boundaries.

2.2.2 Pocket division by bottleneck line

Reflex points and reflex arcs can be called as reflex elements collectively. It is the reflex element that determines where the bottleneck is.

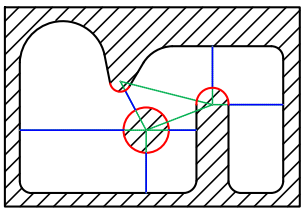


Fig. 7 BnL foundation for arcs

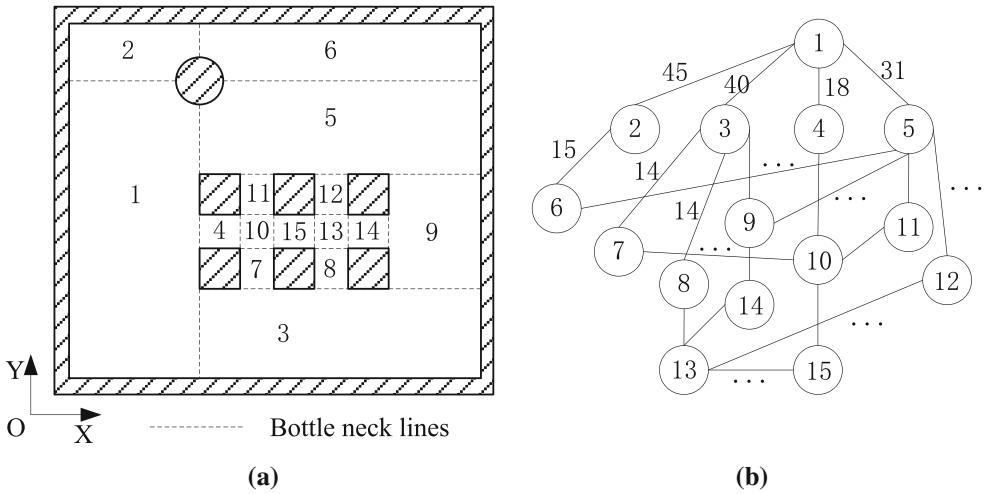


Fig. 8 Sub-regions' numbers and their undirected graph (UGSR). **a** Numbers of sub-areas. **b** UGSR of a divided pocket

Definition 2: bottleneck line Bottleneck line (BnL) is defined as the shortest-distance line segment between the reflex point and B_Ω subject to BnL$\in\Omega$. The length of BnL is represented as l_b.

According to BnL's definition, at least one endpoint of the BnL is a reflex point or on a reflex arc. Therefore, all the BnLs can be recognized based on reflex elements. The key steps of computation for BnL related to a reflex arc are given as follows.

Step 1: Connect the centre of an arc with another reflex point or another arc's centre, and find the shortest distance between the centre and linear edges of B_Ω.

Step 2: Delete line segments not intersecting with the arc and those intersecting with B_Ω more than two points. As is shown in Fig. 7, red lines are arcs, and blue solid lines are valid BnLs. Green lines should be removed.

Step 3: Trim the obtained line segments by the arc and keep the blue segments between the arc and other boundaries as shown in Fig. 7.

Step 4: Suppose $l_{b1}\le l_{b2}\le\ldots\le l_{bk}\le\ldots\le l_{bn}$ are obtained for an circular arc, if $l_{bk}\ge D$, where D is the diameter of the largest tool in the tool bank, remove $\mathrm{BnL}_k,\ldots,\mathrm{BnL}_n$ as they would not restrict the tool selection. Yet, if the arc is a circle, at least two BnLs should be left to construct sub-regions.

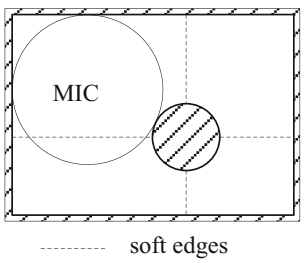


Fig. 9 The MIC of a sub-polygon

The computation for BnL related to reflex points is similar but simpler to decide. So, in this paper, it would not be discussed in detail.

BnLs divide the machining area Ω into numbers of sub-polygons. And, BnLs are called soft edges of each sub-polygon while the edges belonging to P_B_p or P_B_i are considered as hard edges. Soft edge is not a real boundary. So, the cutter is allowed to meet with it.

2.3 Numbering the sub-regions

To further plan the sub-regions and generate shorter tool path, it is necessary to number the sub-regions in sequence.

1. If the sub-area's boundary contains the extreme point with coordinate (x_{min}, y_{min}) of B_p, this sub-region is numbered one.

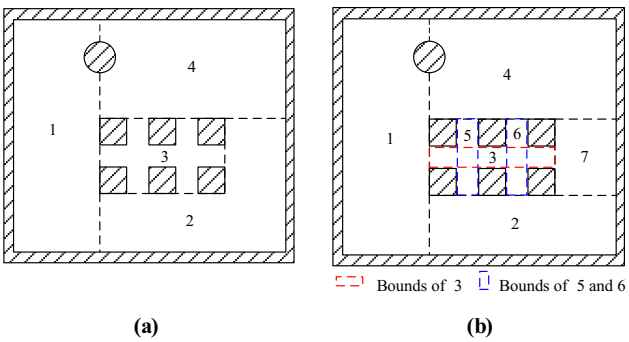


Fig. 10 Illustration of region combination by heuristic rules. **a** Combination by rule 1. **b** Recombination by rules 2 and 3

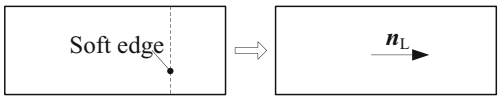


Fig. 11 Combining regions with different cutting direction

2. Search and number its neighboring sub-regions according to its soft edges based on breadth-first algorithm. The sub-polygon with smaller *x*-coordinate edge will be numbered first.
3. When the *x*-coordinates are the same, smaller *y*-coordinate will be considered first.

Figure 8a shows that all the sub-regions of a decomposed pocket are numbered. Then, an undirected graph of the numbered sub-regions (UGSR) can be constructed (shown in Fig. 8b). The nodes are sub-polygons while the edges of the graph are soft edges. The weight values on the edges are the lengths of the soft edges. And, the degree of a node shows the number of its surrounding sub-regions.

3 Tool selection and tool path pattern determination

Discussion about tool selection and tool path determination in this section aims at providing foundations for further sub-region recombination in the next section. First, a tool is selected for each sub-area rested on maximum inscribed circle (MIC) and other rules. Then, tool path pattern is decided according to sub-region's geometry shape.

3.1 Tool selection

To decide the maximum available cutter as the main cutter to remove most of the material, MIC of each sub-polygon is computed first. Based on the MIC, serious rules for cutter selection are given.

3.1.1 Maximum inscribed circle

The MIC of a sub-polygon is allowed to intersect with soft edges and bounded by the hard edges shown in Fig. 9. Furthermore, the center of the MIC should be inside of the convex hull of the sub-polygon. The algebraic description of this question is as follows.

Maximize $R_{\mathrm{MIC}}(x_{\mathrm{MIC}}, y_{\mathrm{MIC}})$

Subject to

$$R_{\mathrm{MIC}} \le \left| \frac{a_i x_{\mathrm{MIC}} + b_i y_{\mathrm{MIC}} + c_i}{\sqrt{a_i^2 + b_i^2}} \right|, i = 0, 1, \ldots, l \quad (1)$$

$$R_{\mathrm{MIC}} \le \sqrt{\left(x_j - x_{\mathrm{MIC}}\right)^2 + \left(y_j - y_{\mathrm{MIC}}\right)^2} - R_j, j = 0, 1, \ldots, m \quad (2)$$

$$R_{\mathrm{MIC}} \le R_k - \sqrt{\left(x_k - x_{\mathrm{MIC}}\right)^2 + \left(y_k - y_{\mathrm{MIC}}\right)^2}, k = 0, 1, \ldots, n \quad (3)$$

$$(x_{\mathrm{MIC}}, y_{\mathrm{MIC}} \in \mathrm{CH(SP)}) \quad (4)$$

where R_{MIC} is the radius of the MIC. $(x_{\mathrm{MIC}}, y_{\mathrm{MIC}})$ is the location of the center of the MIC.

$a_i x + b_i y + c_i = 0$ is the linear equation of the *i*th hard straight edge of the sub-polygon. l is the number of the hard linear edges in the sub-polygon. When all the hard edges are circular arcs, $l=0$.

$(x_j, y_j)$and R_j are the center and radius of the *j*th hard reflex circular arc of the sub-polygon, respectively. m is the number of the hard reflex circular edges in the sub-polygon. If there is no reflex arc as hard edge in the sub-polygon, $m=0$.

$(x_k, y_k)$and R_k are the center and radius of the *k*th hard non-reflex circular arc of the sub-polygon. n is the number of the hard non-reflex circular edge in the sub-polygon. If there is no convex reflex arc as hard edge in the sub-polygon, $n=0$.

CH(SP) is the domain bounded by the convex hull of the sub-polygon.

If the acceptable corner radius is neglected, the diameter of the MIC is the LCC of the sub-region. Thus, the MIC is a key to determine the size of the selected cutter.

3.1.2 Tool selection rules

Suppose n tools, $T_1, T_2, \ldots, T_i, \ldots, T_n$, are selected to machine a given pocket, and their diameters are $D_1, D_2, \ldots, D_i, \ldots, D_n$, respectively, in an increasing sequence. Then, the rules to

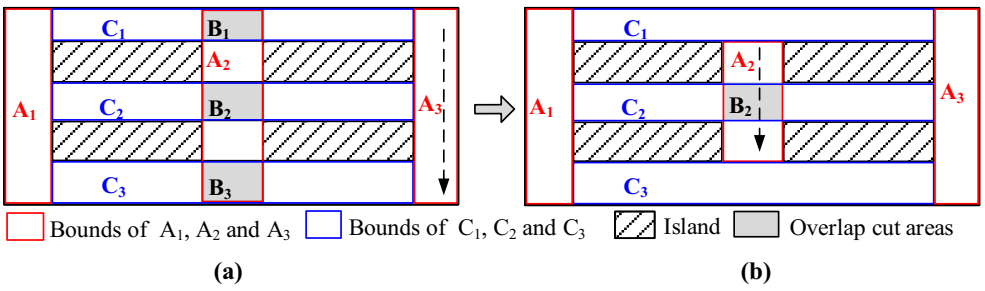


Fig. 12 Illustration of regions overlap cut. **a** Region overlaps. **b** Region subtraction

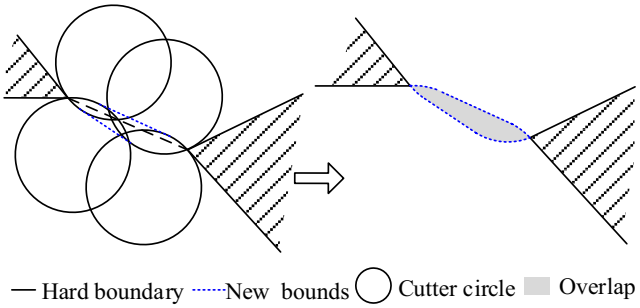


Fig. 13 Illustration of expanding regions

decide the tool for a special sub-area are stated as follows. Where, the first rule is helpful to narrow the list of available tools for a given pocket.

1. $D_1=2R_c$, $D_2\leq\min\{l_{b1}, l_{b2},\ldots, l_{bm}\}$, $D_n\leq\max\{LCC_1, LCC_2,\ldots, LCC_{m+1}\}$, where m is the number of BnLs in the given pocket.
2. The size of the main cutter for each sub-area should be smaller than its LCC.
3. The sub-areas with identical LCC should be cut by the same main cutter. However, though the LCCs are different, it is possible for the relative sub-regions to share one main cutter.
4. If the LCC of the k_{th} sub-area satisfies $D_i\leq LCC_k\leq D_{i+1}$, tool T_i is used in sub-area k.

3.2 Tool path pattern determination

As the staircase shape left by direction tool path results in an uneven surface finish [1] and the unmachined areas are sometimes not only along the hard boundaries, but also along the soft edges [9], direction parallel tool path is not an optimized tool path pattern for the divided pocket. As contour parallel has no cusps with fewer switchbacks in most cases, it is preferred as the main tool path pattern for most sub-regions. However, given that direction parallel involves less computation, zigzag is recommended to use when the sub-polygon is a rectangle. Additionally, by expanding the soft edges of sub-region, it is shown in the following part that the problem discussed above about direction tool path can be avoided efficiently when the sub-polygon is a rectangle. Thus,

```
If (a sub-polygon is a rectangle)
{
 Direction tool path pattern is employed for this sub-area;
  n_L of the sub-polygon is chosen as the cutting direction;
 }
Else
 { Contour tool path pattern is employed for this sub-area;}
```

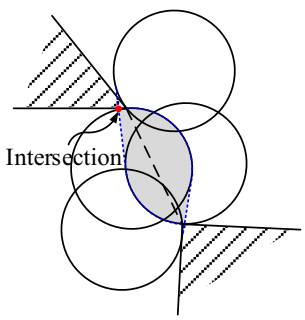


Fig. 14 Tangent line intersect with hard boundary

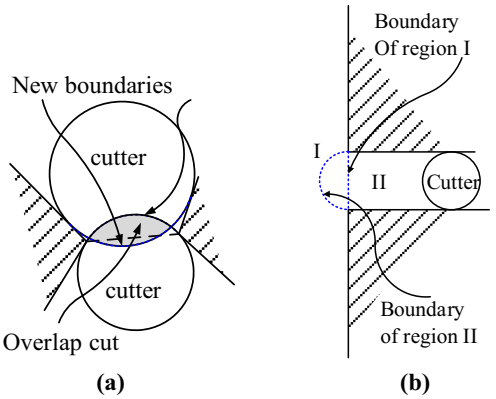


Fig. 15 Arc boundaries for $D_m\geq l_b$. **a** Arc bounds for $D_m>l_b$. **b** Boundary without expansion

4 Sub-region recombination

To obtain shorter path and lessen non-cutting repositioning move and retraction, sub-region recombination for minimizing the number of sub-areas is necessary. First, according to the selected tool and tool path pattern, heuristic principles are explored for sub-region replanning. Then, as soft edge has no limitation to cutter, sub-region is expanded by constructing new bounds at bottlenecks.

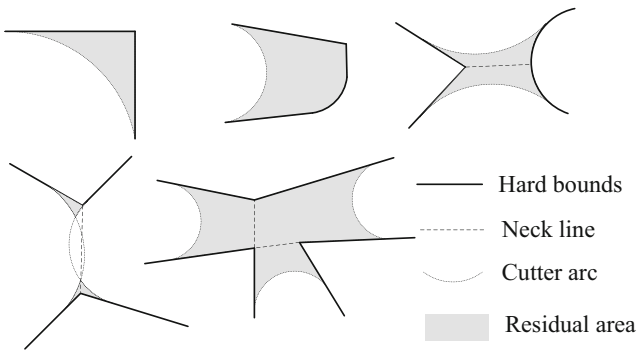


Fig. 16 Various possible uncut areas

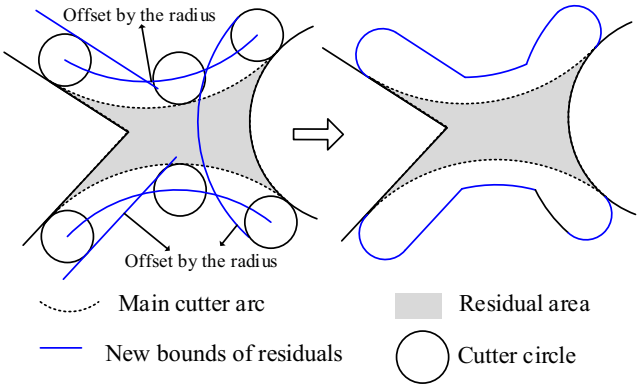


Fig. 17 Determination process of the machining bounds

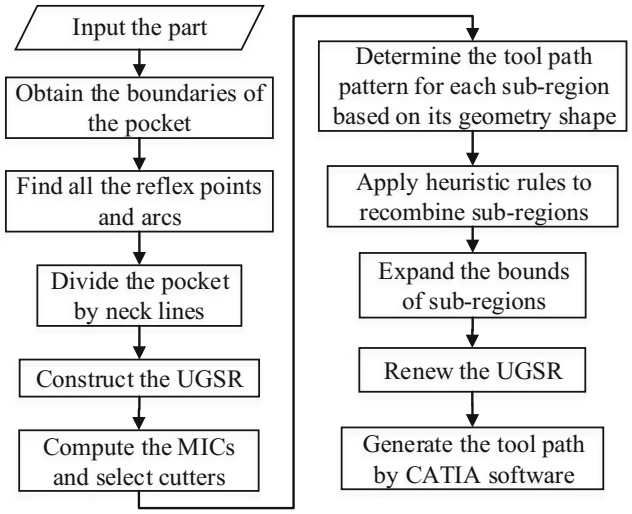


Fig. 19 Implementation process of the approach

4.1 Heuristic rules for sub-region recombination

Redundant sub-areas would raise the number of discontinuous unmachined corners and shorten the length of single tool path. These issues increase idle time spent in approaching and retracting. Whereas, combining adjacent sub-areas probably can efficiently decrease the number of unmachined corners and effectively increase the average length of each tool path. Hence, following heuristic rules are addressed to recombine sub-areas.

Rule 1: Same cutter diameter. For contiguous polygonal areas using the same main cutter, if the weight values on the edges connecting the areas' nodes are smaller than the tool diameter, such areas are combinative. This is a primary principle. Any other recombination principles should be based on it. Illustration of this rule for the pocket presented in Fig. 8 is shown in Fig. 10a.

Rule 2: Identical sweeping direction. For conterminous areas employing zigzag path pattern, if their cutting directions $\boldsymbol{n}_L$ are the same, they can be combined based on rule 1. However, as Fig. 11 depicts, if the common soft edge is the shorter edge of a rectangular sub-polygon and the longer one of its adjoining rectangle, the two rectangular sub-polygons can also be combined as a sub-region. And, the $\boldsymbol{n}_L$ of this combined sub-region coincides with the former sub-polygon.

Rule 3: Continuity of a tool path. To make a tool path consecutive, it is necessary to make overlap cut in some areas. Additionally, some overlap can be avoided by sub-region subtraction as the tool path's continuity would not be affected.

Rule 3.1: Region overlap

Given polygons P_I, P_{II}, and P_{III}, nodes of P_I and P_{II} in UGSR both connect with P_{III}. Suppose the soft edges are BnL_1 between P_I and P_{III}, and BnL_2 between P_{II} and P_{III}, respectively. p_1 and p_2 are endpoints of BnL_1 while p_3

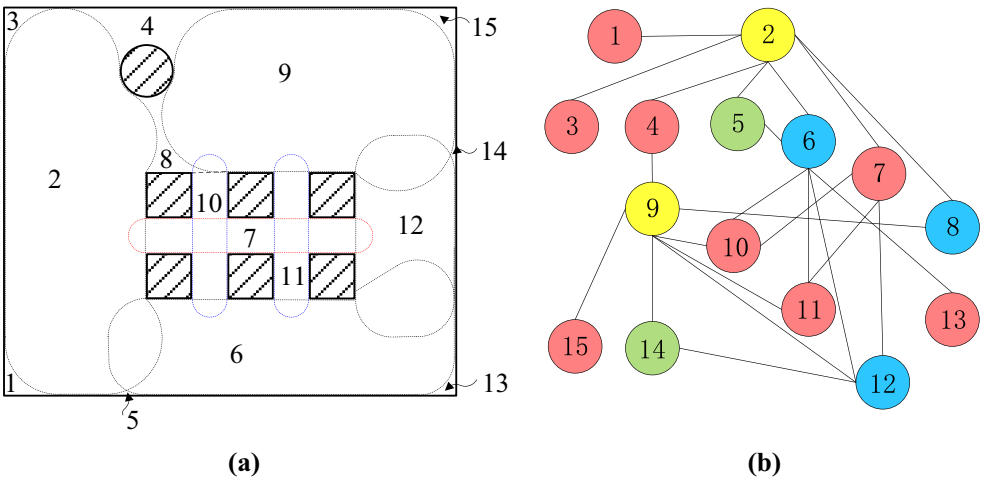
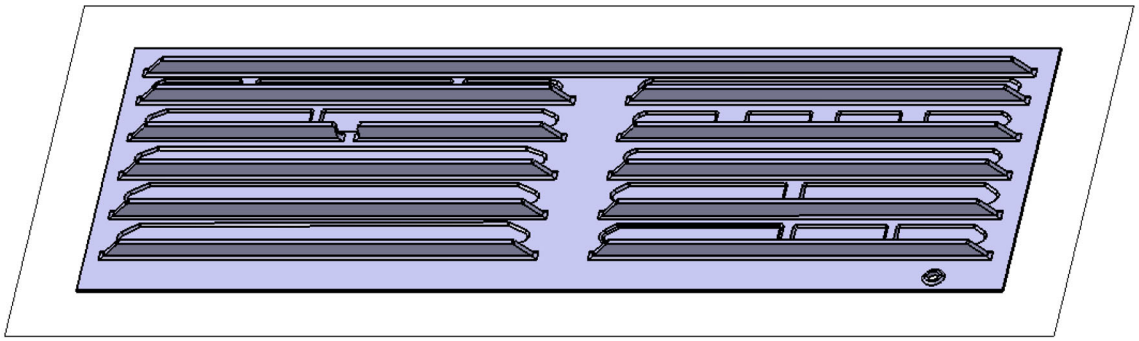
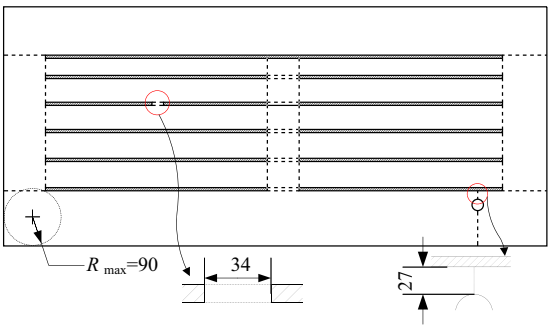


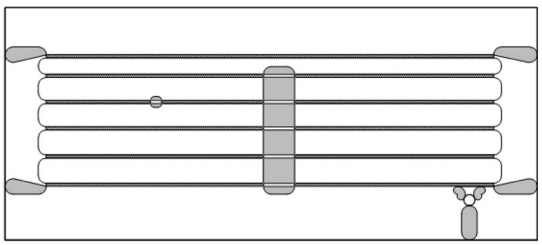
Fig. 18 Final sub-region layout and their undirected graph (UGSR). **a** Final sub-region layout. **b** UGSR of sub-regions



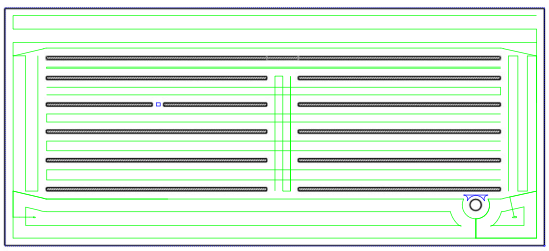
(a)


R max=90
34
27

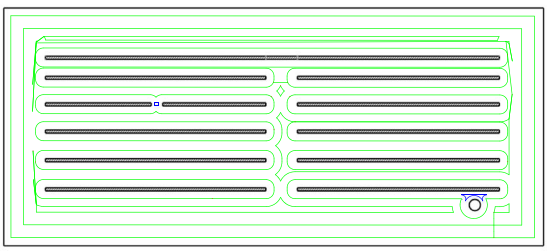
(b)



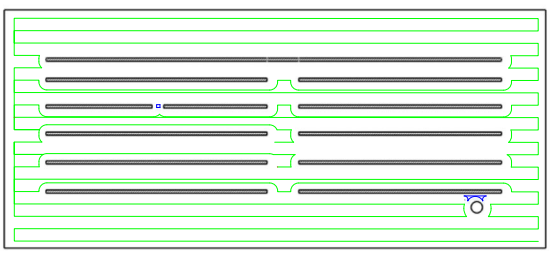
(c)



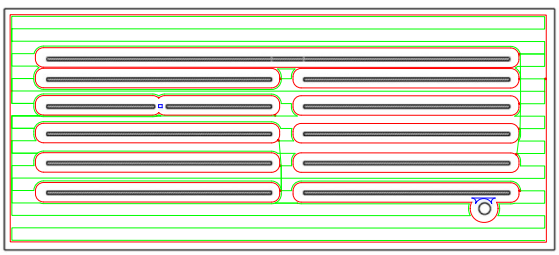
(d)



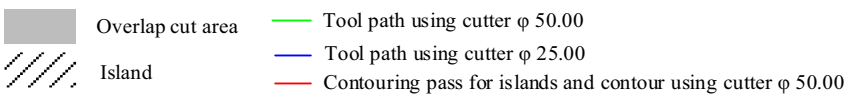
(e)



(f)



(g)


Overlap cut area
Island
Tool path using cutter φ 50.00
Tool path using cutter φ 25.00
Contouring pass for islands and contour using cutter φ 50.00

◀ **Fig. 20** Example 1. **a** The CAD solid model of an aircraft panel with many parallel ribs and its stock's boundary. **b** The intersection of generalized pocket model and the final pocket division result. **c** The expanded bounds of sub-regions and overlap cut areas. **d** Tool path generated by the proposed method. **e** Tool path generated by outward helical pattern using CATIA®. **f** Tool path generated by back and forth pattern using CATIA®. **g** Tool path generated by back and forth with contouring pass pattern using CATIA®

and p_4 are those of BnL_2. If all the following conditions are satisfied, combine P_I, P_{II}, and quadrangle $p_1p_2p_4p_3$ as a new sub-polygon with P_{III} unchanged.

1. P_I and P_{II} are rectangles;
2. $\boldsymbol{n}_{L\_I}//\boldsymbol{n}_{L\_II}$;
3. $l_{b1}\geq D_m$&& $l_{b2}\geq D_m$, D_m is for P_I and P_{II};
4. Quadrangle with vertices p_1, p_2, p_3, p_4 is a parallelogram and $\overrightarrow{p_1p_3}$ $//\boldsymbol{n}_{L\_II}$.
5. In Fig. 12a, the gray areas belong to region A_2 as well as C series regions. And, the sub-areas shown in Fig. 8a are recombined as Fig. 10b illustrates.

Rule 3.2: Region subtraction

If the overlapped area is the end of P_{III}, that is, subtracting the overlap will not split P_{III} into several unconnected sub-areas, then, such area can be removed from P_{III}. As Fig. 12b shows, polygons B_1 and B_2 are subtracted from polygon A_2.

For pockets with many rectangular sub-regions, like aircraft panel with many parallel ribs, principles 2 and 3 may bring about a powerful effect on reducing the number of sub-regions. For others, they may do no good. So, both the two principles are optional. However, rule 1 is a recommended principle for various pockets.

4.2 Expanding the sub-region

Since soft edge has no limitation to cutter, it is made use of to enlarge the bounds of sub-region.

4.2.1 Sub-region expansion

Though the presented subdivision method will not produce residual area like Fig. 2 shows, uncut corners are unavoidably left by the selected cutter in each sub-region. To reduce the unmachined areas, new bounds are constructed for each sub-region using arcs with D_m and the related tangent line at soft edges.

1. $D_m<l_b$

When $D_m<l_b$, new bounds are constructed as follows. First, two selected tool circles being tangent to the hard edges at BnL's endpoints are determined. Then, the tangential line segment of the above circles is used to form the new boundary as Fig. 13 shows. If the tangent line intersects with hard edges, keep the section between the tangent point and the intersection shown in Fig. 14.

2. $D_m\geq l_b$

If $D_m\geq l_b$, arcs with radius of $D_m/2$ are applied as the new bounds as illustrated in Fig. 15. Figure 15a also shows that the new border arc is either tangent to the hard edges or just connecting the two reflex points. If the BnL is collinear with the hard edges which share the same endpoints with the BnL in a sub-polygon, there is no need to enlarge the sub-polygon, which is shown in Fig. 15b.

4.2.2 Residual area expansion

The purpose of enlarging the sub-polygon is to use the main tool to machine as many areas as possible. However, besides the corner residue, unmachined areas along hard boundaries still possibly occur while $D_m>l_b$. The possible uncut material layout is shown in Fig. 16. And, each uncut area can be considered as a new sub-region.

Similarly, soft edge is also utilized when the machining bounds for corner and bottleneck residuals are determined. Figure 17 illustrates the determination process of machining bounds for bottleneck areas. To construct the bounds, soft edges are offset by the radius of the selected tool for the remained area while the hard edges are offset by the diameter. For unmachined corners, profile machining operation is recommended.

After the sub-region recombination, the UGSR needs to be renewed. Figure 18 shows the final sub-region layout and the UGSR. Except remained materials left by main cutters, each sub-region is bounded by dotted lines as shown in Fig. 18a. In the UGSR, the nodes with same color are machined by the same tool. So, if there is an edge connecting the same style nodes, they can be removed in sequence. Small sub-regions like numbers 5 and 14 in Fig. 18b can be cleared by final finishing tool.

5 Implementation and discussion

To validate the superiority of the proposed approach, the proposed algorithm has tried on several parts. In this paper, two examples applying this method and the conventional techniques, respectively, are given for comparison. One of the two parts is an aircraft panel with many parallel ribs while

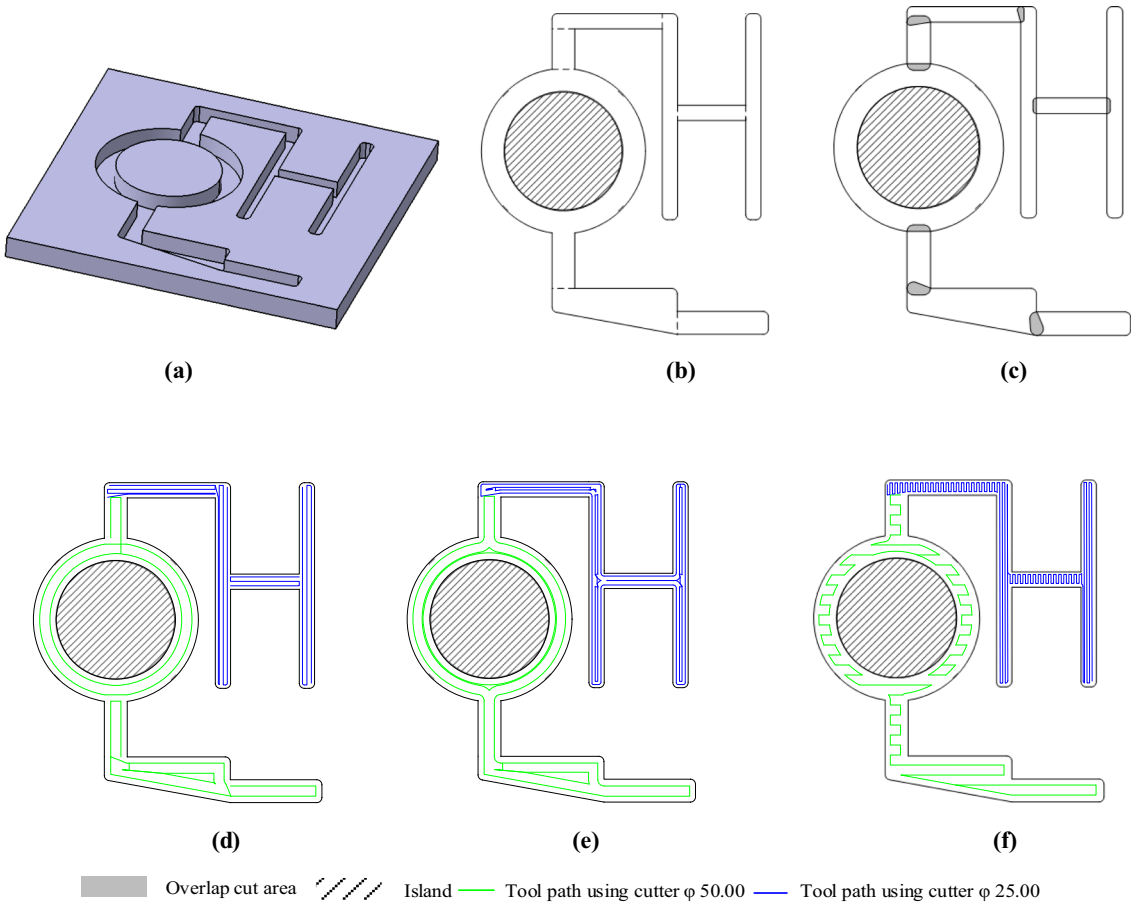


Fig. 21 Example 2. **a** A pocket with an island to be machined. **b** The final division. **c** The expanded bounds of sub-regions and overlap cut areas. **d** Tool path generated by the proposed method. **e** Tool path generated by outward helical pattern using CATIA®. **f** Tool path generated by back and forth pattern

the other is a pocket with an island. As CATIA® is a multi-platform CAD/CAM/CAE commercial software suite which provides processing solution to a wide variety of industries, such as aerospace, we adopt it to generate the tool path for pockets before and after division.

Figure 19 illustrates the implementation process of the proposed approach. In the second step, the contours are obtained by intersecting their generalized pocket model [25]. The part models and the important results are shown in Figs. 20 and 21, including the final pocket split and recombination results, the expanded bounds of sub-regions and overlap cut areas, the tool path generated by the proposed method, the conventional contour, and zigzag tool path. Tables 1 and 2 show the tool path lengths generated by this addressed method and the conventional ways. And, Table 3 gives savings of cutter path using the split and recombination method.

As shown in Fig. 20b, the minimum bottleneck width is 27 mm and the R_{MIC}=90 mm. According to the tool selection rules, cutters with φ50 and φ25 are selected for the pocket roughing. In Fig. 20d, g, the green lines represent the tool path using cutter φ50 while the blue ones illustrate the tool path using cutter φ25. We can see that the tool paths with cutter φ25 are the same for various methods. The only difference is the tool path for cutter φ50. Figure 20d illustrates the tool path generated by the proposed method. Most of the sub-regions are machined in a zigzag pattern with different directions.

Table 1 Results of example 1 with this approach and the conventional ways

Applied method	Tool path length (mm)		Total tool path length (mm)
	φ50	φ25	
Subdivision method	32,400	264	32,664
Contour pattern	34,560	264	34,824
Zigzag pattern	25,421	264	25,685
Zigzag with contouring pass pattern	45,905	264	46,169

Table 2 Results of example 2 with this approach and the conventional ways

Applied method	Tool path length (mm)		Total tool path length (mm)
	φ50	φ25	
Subdivision method	6993	8093	15,086
Contour pattern	8563	9215	17,778
Zigzag pattern	6704	10,396	17,100

Figure 20e plots the tool path in the outward helical pattern in CATIA®, i.e., the conventional contour-parallel pattern. From Fig. 20f, we can see that there are many residuals left while back and forth pattern in CATIA®, i.e., pure zigzag pattern, is employed. However, that is unavoidable when the geometry of the pocket is complex due to the limitation of the algorithm for a zigzag pattern. Therefore, it is necessary to add contouring pass for islands and contour to pure zigzag pattern, as the red lines shown in Fig. 20g, which increases the tool path length a lot as shown in Table 1. Compared to contour pattern, though the introduced approach only saves 6.2 % tool path (shown in Table 3), its tool path is regular while there are many sharp turns and retractions for conventional contour path, which will decrease the machining speed and is not fit for HSM.

Figure 21 gives a relatively simple pocket with only one island as example. Cutters with φ50 and φ25 are also employed to remove most of the materials. In Fig. 21d–f, the green lines represent the tool path using cutter φ50 while the blue ones illustrate the tool path with cutter φ25. Figure 21f plots that different cutting directions are applied for cutters φ50 and φ25, respectively, when using back and forth pattern. Additionally, to decrease the residuals left, the radial depth of cut for cutter φ25 in Fig. 21f is 40 % while 80 % radial depth is adopted for other situations in this example. From Tables 2 and 3, it can be seen that the presented division technique reduces the cutter path lengths more than 10 % in this case compared to the traditional ways without split.

From the figures, tables, and above analyze, the main advantages of the advised split and recombination method for pocket roughing are concluded as follows:

1. The length of the tool path generated by the addressed algorithm is shorter than that of the single type of pattern, as Table 3 illustrates that the split and recombination method even saves as much as 29.2 % cutter path comparing to back and forth with contouring pass pattern using CATIA® when same cutters are employed.
2. The radial depth for each sub-region can be set separately to minimize the tool path length as well as remove all the machining areas. When the geometry of the pocket is complex, especially with many islands, much residual areas would be left if pure zigzag tool path is applied though the accessibility of the selected cutter is satisfied. To reduce or clear these unwanted residuals, shortening the radial depth of cut or adding contouring pass is necessary. However, both of the techniques will highly increase the tool path length throughout the pocket. By contrast, this bad influence to split and recombined pocket declines sharply because the radial depth for each sub-region can be set separately.
3. With the rendered method, the cutter path can be optimized for each split area. As we know, while multiple cutters are adopted in a pocket, tool path can be optimized for each cutter in its accessible areas. For example, directions of the zigzag tool path for cutters φ50 and φ25 in Fig. 21f are different. Nevertheless, with the presented methodology, the cutter path can be further improved for each split area without limitation of the selected cutter. For instance, the orientations of the zigzag tool path for each sub-region of cutter φ25 in Fig. 21d are different according to its longest edge.
4. The introduced cutter path is relatively smooth. When there are a number of islands in the pocket as Fig. 20a, b shows, many sharp turns and retractions appear in conventional contour path, as illustrated in Fig. 20e, which may slow down the machining speed and is not fit for HSM. On the contrary, the tool path generated by the rendered method is smooth comparatively and capable of removing all the machining materials with getting better machining quality (depicted in Fig. 20d).

Table 3 Savings of cutter path using the split and recombination method

Traditional way without split	Example 1 (%)	Example 2 (%)
Compared to contour pattern	6.2	15.1
Compared to zigzag pattern	29.2	11.7

6 Conclusions

To improve the efficiency of machining a pocket with many islands and complex boundaries, an automated CNC programming approach based on cutter capacity, polygon decomposition and recombination is proposed. First, the LCC of a pocket

which restricts the machining efficiency of a pocket radically is investigated and defined. Build on this concept, BnLs of the pocket are adopted as the separators to decompose the pocket into many sub-regions. Then, heuristic rules for sub-region recombination are introduced to efficiently diminish the number of the sub-polygons, as well as to lengthen the average tool path. Finally, hybrid tool path patterns with multiple cutters are applied to the divided pocket.

The algorithm has been implemented and tested. Experimental results show that the presented approach has significantly better performance with respect to both machining quality and efficiency over the existing contour and zigzag path patterns. Our future work is to explore a more efficient tool path pattern for uncut materials at corners and bottleneck areas. Additionally, zigzag is only applied in rectangular sub-polygon in this work. The future work will be devoted to better hybrid path patterns.

Acknowledgments This research was supported by the National Science and Technology Major Project (NSTMP) under Grant No. 2012ZX04010051.

References

1. Yao Z, Gupta SK (2004) Cutter path generation for 2.5D milling by combining multiple different cutter path patterns. Int J Prod Res 42(11):2141–2161
2. Bala M, Chang TC (1991) Automatic cutter selection and optimal cutter path generation for prismatic parts. Int J Prod Res 29(11): 2163–2176
3. Lee K, Kim TJ, Hong SE (1994) Generation of toolpath with selection of proper tools for rough cutting process. Comput Aided Des 26(11):822–831
4. Ahmad Z, Rahmani K, D'souza RM (2010) Applications of genetic algorithms in process planning: tool sequence selection for 2.5-axis pocket machining. J Intell Manuf 21(4):461–470
5. Chen ZC, Fu Q (2011) An optimal approach to multiple tool selection and their numerical control path generation for aggressive rough machining of pockets with free-form boundaries. Comput Aided Des 43(6):651–663
6. Veeramani D, Gau Y-S (1997) Selection of an optimal set of cutting-tool sizes 2.5D pocket machining. Comput Aided Des 29(12):869–877
7. Veeramani D, Gau Y-S (2000) Cutter-path generation using multiple cutting-tool sizes for 2-1/2D pocket machining. IIE Trans 32(7): 661–675
8. You C-F, Sheen B-T, Lin T-K (2007) Selecting optimal tools for arbitrarily shaped pockets. Int J Adv Manuf Technol 32:132–138
9. Ramaswami H, Shaw RS, Anand S (2011) Selection of optimal set of cutting tools for machining of polygonal pockets with islands. Int J Adv Manuf Technol 53(9):963–977
10. Nadjakova I, McMains S (2004) Finding an optimal set of cutter radii for 2D pocket machining. In: Proceedings of 2004 ASME international mechanical engineering congress and RD & D expo. IMECE 62224:2004
11. Makhe A, Frank MC (2010) Polygon subdivision for pocket machining process planning. Comput Ind Eng 58(4):709–716
12. Lim T, Corney J, Clark DER (2000) Exact tool sizing for feature accessibility. Int J Adv Manuf Technol 16:791–802
13. Zhang Y, Ge L (2009) Selecting optimal set of tool sequences for machining of multiple pockets. Int J Adv Manuf Technol 42(3–4): 233–241
14. Lee Y-S, Chang T-C (1995) Application of computational geometry in optimizing 2.5D and 3D NC surface machining. Comput Ind 26(1):41–59
15. Yu F, Zheng G, Rao Y, Du B, Chu H (2010) Algorithms for selecting optimal cutters in pocket machining based on geometric characteristics. Comput Aided Des Comput Graph 22(11):1984–4989, **in Chinese**
16. Lai W, Faddis T, Sorem R (2000) Incremental algorithms for finding the offset distance and minimum passage width in a pocket machining toolpath using the Voronoi technique. J Mater Process Technol 100(1):30–35
17. Yao Z, Gupta SK, Nau DS (2001) A geometric algorithm for finding the largest milling cutter. J Manuf Process 3(1):1–16
18. Yao Z, Gupta SK, Nau DS (2001) A geometric algorithm for selecting optimal set of cutters for multi-part milling. Proceedings of the sixth ACM symposium on Solid modeling and applications. ACM 130–139
19. Chen ZC, Zhang H (2009) Optimal cutter size determination for 2½-axis finish machining of NURBS profile parts. Int J Prod Res 47(22):6279–6293
20. Han FY, Zhang DH, Luo M, Wu BH (2014) Optimal CNC plunge cutter selection and tool path generation for multi-axis roughing free-form surface impeller channel. Int J Adv Manuf Technol 71(9–12):1801–1810
21. Park S, Choi B (2000) Tool-path planning for direction-parallel area milling. Comput Aided Des 32(1):17–25
22. Vosniakos G, Papapanagiotou P (2000) Multiple tool path planning for NC machining of convex pockets without islands. Robot Comput Integr Manuf 16(6):425–435
23. Kim BH, Choi BK (2002) Machining efficiency comparison direction-parallel tool path with contour-parallel tool path. Comput Aided Des 34(2):89–95
24. Yao Z (2006) A novel cutter path planning approach to high speed machining. Comput Aided Des Applic 3(1–4):241–248
25. Yu F, Du B, Ren W, Zheng G, Chu H (2008) Slicing recognition of aircraft integral panel generalized pocket. Chin J Aeronaut 21(6): 585–592