Mansoura Engineering Journal, (MEJ), Vol. 39, Issue 2, June 2014

Optimal Tools Selection for Rough Machining of Freeform Cavities Using Cavity Decomposition

اختيار أمثل لعدد القطع للتشغيل الاستقرابى للفجوات ذات الأسطح الحرة باستخدام تقسيم الفجوة

T. T. El-Midany[a], H. E. Elshourbagy[b], M. A. Elkeran[c]
[a] Professor of Production Engineering, Mansoura University
[b]Assistant Professor of Production Engineering, Mansoura University
[c] Demonstrator of Production Engineering, Mansoura University,
melkeran@gmail.com

الملخص

يعتبر اختيار عدد القطع من أهم مراحل تخطيط عمليات الانتاج حيث يؤثر بشكل كبير على زمن وتكلفة التشغيل. لوجود الكثير من الإحتمالات لاختيار عدد القطع، فمن الصعب إيجاد أمثل اختيار لعدد القطع بواسطة المبرمج وخصوصاً مع وجود قيود هندسية معقدة فى حالة الأسطح الحرة. لذلك اتجه بعض الباحثين إلى تحديد أمثل اختيار لعدد القطع عن طريق تقدير حجم المعدن المزال بواسطة كل عدة ويعتبر هذا الاتجاه غير دقيق نتيجة إهمال الاعتبارات التكنولوجية لمسارات عدد القطع (Tool Paths). كذلك اتجه باحثون آخرون إلى تحديد أمثل اختيار لعدد القطع عن طريق حساب مسارات العدد، وهذا الاتجاه يعتبر أدق من السابق. يقدم هذا البحث طريقة جديدة وفعالة لتحديد أمثل اختيار لعدد القطع للتشغيل الإستقرابى للأسطح الحرة اعتماداً على حساب مسارات العدد وتقسيم الفجوة، وتقدم هذه الطريقة وفراً فى الحسابات الهندسية للسطح الحر وأكثر اعتمادية فى الحسابات العددية. و قد تم التحقق من صحة النتائج عن طريق المقارنة ببرنامج تصنيع بواسطة الحاسب (CAM Software) وأظهرت النتائج تحسناً فى وقت الحسابات (CPU Time) عن الطريقة السابقة بنسبة 30 إلى 50 % تقريباً.

Abstract

Tools selection is very important stage in production process-planning and has great impact on the machining time and cost. It is difficult for the part programmer to determine the optimal set of tools that achieve the minimum machining cost. Optimal tools selection is a difficult problem due to large number of tools' combinations and particularly for complex machining features. Some researchers determined the tools sequence by computing each tool accessible volume, however this may not lead to the optimal sequence since they neglect the technological aspects of tool paths. Other research works determined the tools sequence by computing tool paths in more accurate way but it consumed more CPU time. This paper introduces a new and efficient method to find the optimal tools sequence for rough machining of freeform cavities based on tool paths'computationsand cavity decomposition. The newly introduced method saves a lot of complex geometric computations and is more numerically robust than previous methods. The proposed method has been validated against commercial CAM software for correctness and reliability. Computational results show that the proposed method improves the previous method in CPU time by about 30 to 50 %.

Keywords

Freeform cavity machining; Tool selection; Optimization; Voronoi diagram; Toolpath

1. Introduction

A cavity is an important and frequently encountered feature for manufacturing mechanical parts. To meet the advanced customer demands, ergonomics, and sophisticated design approaches, cavities with freeform surfaces are widely adopted especially for aerospace, automotive, and die/mold

Received: 28 April, 2014 - Revised: 8 June, 2014 - Accepted: 15 June, 2014

industries. Recently, CAD/CAM systems can generate efficient machining tool paths for roughing, semi-finishing, and finishing operations. Nevertheless, the responsibility of selecting the appropriate set of tools' diameters and number is left for the user andthis depends mainly on the user's experience and/or guessing. It becomes very difficult for the userto find the optimal selection since there is a wide range of tools combinations besides complex geometrical constraints.

Roughing operation is the most time consuming process of machining and may consume more than 60% of the total machining time [1]. Thus, it is economical to rapidly remove material from the cavity and this may be accomplished by using large tools. Large tools are more rigid and can engage at large depths of cut and high feed rates which result in higher material removal rate, however they may not access the whole machining volume. On theother hand, selecting small tools can guarantee no rest material is left, but at lower material removal rates and greater machining times.

The most effective way for rough machining is to use a combination of large and small tools that yield minimal machining time and cost without leaving unmachined areas. It is essential to compute accessible volume and tool paths for each tool. The first tool accessible volume depends on the tool's diameter and the geometry of part cavity. After the first tool, each subsequent tool should only remove the residual unmachined areas. It is necessary to decompose the part cavity to determine the residual unamchined areas after each tool. The cavity decomposition is a computational bottlenecksince it includes complex geometric operations and prone to numerical inefficiencies. Tool selection is a combinatorial optimization problem in which the objective function is to minimize the machining time and/or cost. From the previous research work, researchers have mainly followed two trends.

The first trend is to approximate the machining time calculations by only considering the volume removed by each tool which may be far from application reality. This is due to neglecting the technological aspects of rapid motion tool paths, tool paths stepfeed, tool paths linking, lead-in/lead-out, etc.Lee et al [2] and Lee and Chang [3] developed methods to find a single tool for rough machining of 3-D cavities via hunting planes. The selected tool is typically the largest tool that can reach the entire pocket without gouging. Bala and Chang [4] have developed a method of selecting exactly two tools to machine a given 2-D pocket, however they did not decompose the pocket. Chen, et al [5] extended Bala and Chang's method for machining 3-D cavities. Veeramani and Gau [6] developed a Voronoi mountain based approach for decomposition of a given 2-D contour into sub-contours for various tools, however this technique includes complex and expensive geometrical operations.

The second trend is to accurately calculate the machining time via exact tool paths generation including rapid motions and technological consideration. Despite the accuracy of machining time calculations, tool paths generation process involves time consuming calculations. D'Souza et al [7] introduced an efficient method of finding the tool sequence with the minimum cost of rough machining of freeform cavities. However, they used a decomposition technique that includes complex operations and prone to numerical inaccuracies. Ahmad et al [8] applied genetic algorithm for selecting tool sequence to machine a 2.5-axis pocket. They applied the same pocket decomposition technique by D'Souza et al [7]. Ramaswamiet al [9] decomposed the pocket boundaries into convex regions and mill each region independently by selecting a sequence of tools based on the accessibility to the region. The generated tool paths will include a lot of rapid motions between feed tool paths.

In this paper, a new method for optimal tools selection for rough machining of freeform cavities is proposed. This method is based on tool paths generation and cavity decomposition. The remainder of this paper is organized as follows. Section 2 delivers the problem definition. Section 3 discusses the tool paths generation.Section 4 introduces the proposed method of cavity decomposition. Section 5 gives details about software implementation. Section 6 shows results of optimal tools selection. Finally, section 7 illustrates the final conclusion.

2. Problem definition

The problem of optimal tools selection is defined as finding the tool sequence from a given set of tools for rough machining of a certain part cavity with/without islands that minimizes the totaltime and/or cost. Tools are arranged in decreasing order of diameters $T = t_1,\ t_2 \ldots t_n,\ (t_1 > t_n)$, where each tool is associated with cutting parameters (width of cut, depth of cut, feed, and speed). This problem is mathematically formulated as the following equations:

$$\arg_{\mathrm{x}} \min\ C(x) \mid x\ \subseteq T, \tag{1}$$

$$C(x) = M_c + T_c, \tag{2}$$

$$M_c = \frac{C_{mc}}{60}\left(\sum_{i=1}^{n}\left(\frac{l_{mc}}{f_{mc}} + \frac{l_{air}}{f_{air}}\right)_i + n\,t_{ch}\right), \tag{3}$$

$$T_c = \sum_{i=1}^{n}\left(\left(\frac{l_{mc}}{f_{mc}}\right)\left(\frac{C_t}{t_{lf}}\right)\right)_i, \tag{4}$$

where $C(x)$ is the total cost for machining the cavity using the tool sequence x, M_c is the machining cost, T_c is the tooling cost, C_{mc} is the machining cost per hour, C_t is the tool cost for tool i, l_{mc} is the cutting toolpath length for tool i, l_{air} is the air (rapid motion) tool path length for tool i, f_{mc} is the selected cutting feed rate for tool i, f_{air} is the machine rapid move feed rate, t_{ch} is the tool change time, t_{lf} is the average tool life. All tools used for roughing are end mill cutters with flat bottoms.

Part cavity is machined via 2.5D slices by a series of hunting planes eventhough it has 3D freeform surfaces. Each hunting plane serves to extract part contours and then accessible area and tool path can be computed. Accessible area fora certain tool at a specific hunting planeis the maximum area which can be totally covered by the tool without gouging. Tool

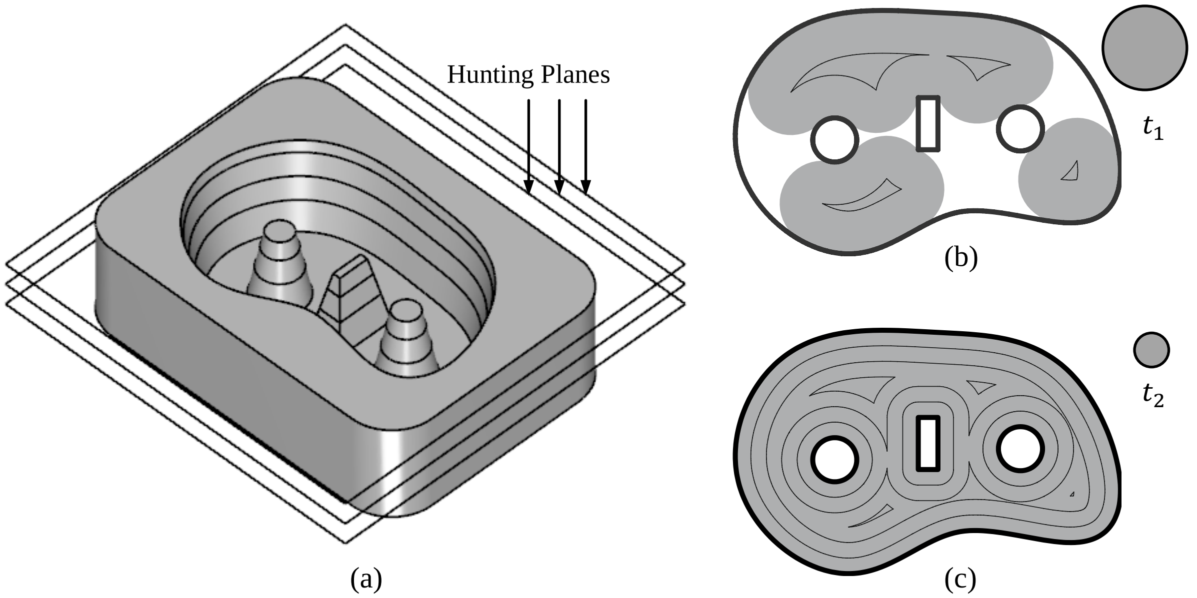


Fig. 1 Free form cavity to be machined. (a) Hunting planes intersect part cavity surfaces to find part contours. (b) Accessible area (shaded) and tool path for large tool t_1. (c) Accessible area (shaded) and tool path for small tool t_2.

path for a certain tool at a specific hunting plane is the path which the tool should trace to cover the accessible area.

Figure 1(a) shows a free form part cavity with three islands where three hunting planes are sketched. Each hunting plane intersects the part cavity surface and yields part contours. Figures 1(b) and 1(c) show part contours at the middle hunting plane along with shaded accessible areas and tool paths. In Fig 1(b), a large tool t_1 isused, however it cannot access all the area of that layer. In Fig. 1(c), a smaller tool t_2 is used and it can access the whole area of that layer without leaving any rest material. Figure 2 shows the basic block diagram for the proposed optimal tools selection algorithm.

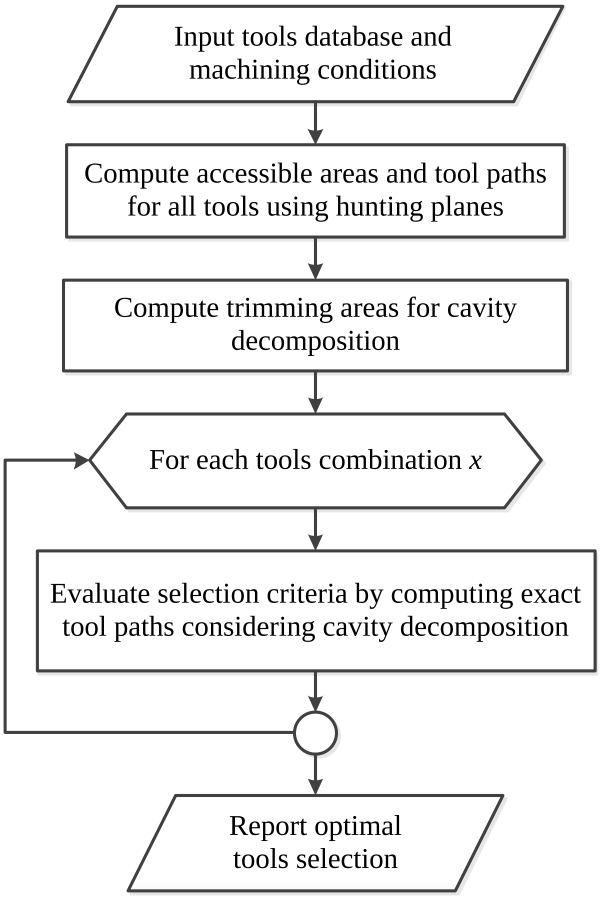


Fig. 2 Optimal tools selection block diagram.

3. Tool path generation

The most fundamental geometric framework for toolpath generation and accessible area computation is offsetting. Contour parallel offset has been studied by many researchers and mainly there are two groups of methods for generating offsets.

The first group is called the *pairwise offsetting* [12-14]. This method is intuitive and simple, however it has time consuming computations especially for repeated offsetting for tool path generation. The second group is called the *Voronoi diagram offsetting*[16-18]. That is known to be more efficient [16-18], however it needs careful implementation due to numerical instability that may happen. In the present work, the Voronoi diagram (VD) method [18] is used for computing offsets and hence accessible areas.

Figure 3 shows Voronoi diagram for part contours of Fig. 1(b). Algorithm 1 shows how to compute the tool path for part contours pc and tool diameter d. Algorithm 2 shows how to compute the accessible area. After generating separated parallel tool path contours at every hunting plane, they must be linked to produce a singly connected tool path for each tool [15].

Algorithm 1. ComputeToolPath(pc, d)

```
$TPC = \emptyset$     // Tool path contours
$r = d/2$
do
    $tp = \text{offset}(pc, -r)$  // using VD
    $TPC = TPC \cup tp$
while ($tp \neq \emptyset$)
return $TPC$
```

Algorithm 2. ComputeAccessibleArea(pc, d)

```
$r = d/2$
$F = \text{offset}(pc, r)$
$AccessibleArea = \text{offset}(F, -r)$
return $AccessibleArea$
```

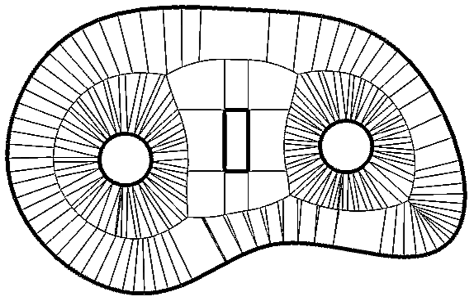


Fig. 3 Voronoi diagram for part contours of Fig. 1(b).

4. Cavity decomposition

Using a set of tools for cavity machining, each tool should remove as much as it can access depending on its diameter and the geometry of cavity surfaces and machining layers. After applying the first tool, every subsequent tool removes only the rest material and so there must be a way to decompose the cavity for using successive tools. Figure 4 shows the rest material which is left after using tool t_1 in Fig. 1(b). The boundary that results from area difference between the layer area and the accessible area of a tool is called open boundaries [7] as shown in Fig. 4. The rest material should be removed by a smaller tool and to determine the tool path for the smaller tool, its accessible area should be calculated. The previous method for cavity decomposition [7, 8] includes expensive geometric operations and prone to numerical inaccuracies.

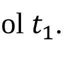


Fig. 4 Rest material (shaded) after using tool t_1.

Algorithm 3 finds the tool path for a smaller tool t_2 which has an accessible area of A_2 after machining a layer using larger tool t_1 which has an accessible area of A_1. Figure 5 shows the stages of Algorithm 3. The previous cavity decomposition method computes three offset operations at lines 2, 3, and 5 besides doing a Boolean difference operation between polygons X and Y at line 4. In Algorithm 3, the Boolean difference operation is critically prone to numerical errors especially when dealing polygons with holes (islands) and/or of multiple components.

Algorithm 3. DecCavity1(A_1, A_2, d)

```
// Traditional cavity decomposition
1. $r = d/2$      // radius of tool $t_2$
2. $X =$ offset($A_1$, $-r$)
3. $Y =$ offset($A_2$, $-r$)
4. $Z = Y - X$
5. $A_{new} =$ offset($Z$, $r$)
6. $TPC =$ ComputeToolPath($A_{new}$, $d$)
7. returnTPC
```

The proposed method for cavity decomposition deals with tool path trimming. The tool path of each tool is trimmed by the accessible area of former tools in the sequence. In order to ensure that the smaller tool begins cutting without vertical plunging into material, the former tools accessible areas are offset by the smaller tool radius.

Algorithm 4 describes the proposed method for cavity decomposition, while Fig. 6 shows the stages of Algorithm 4. Consequently, the proposed method saves a lot of computations since it only computes one offset operation at line 2 and trims the tool path at line 4 of Algorithm 4. Tool path trimming include straightforward segment intersections and point in polygon inclusion tests. The new method is more reliable than the previous method for numerical stability.

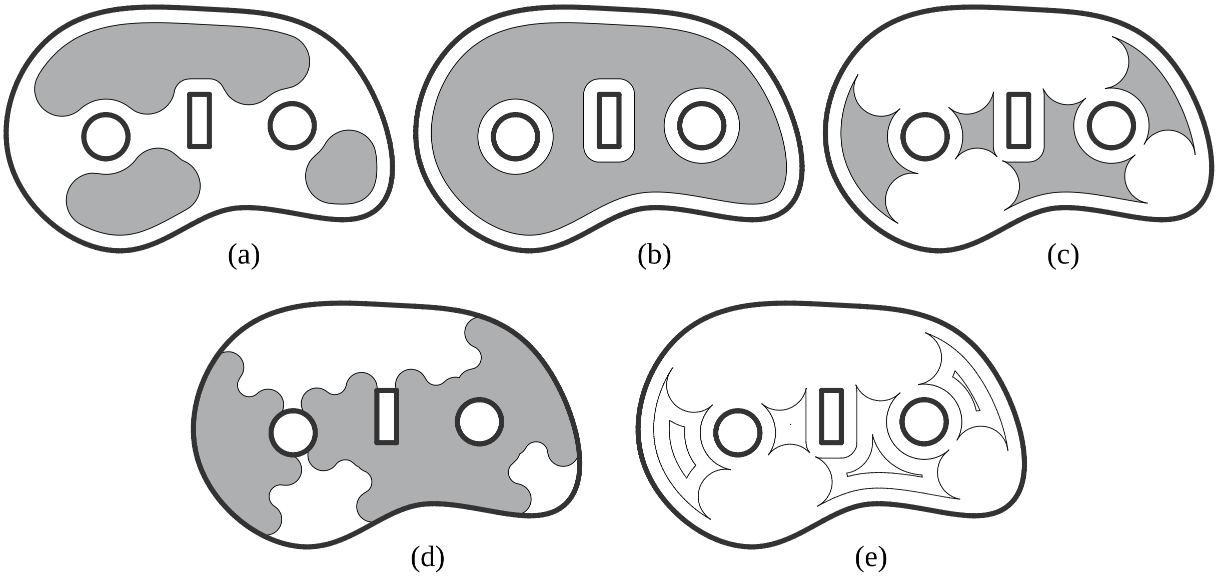


Fig. 5 Previous cavity decomposition. (a) line 2, (b) line 3, (c) line 4, (d) line 5, and (e) line 6 in Algorithm 3.

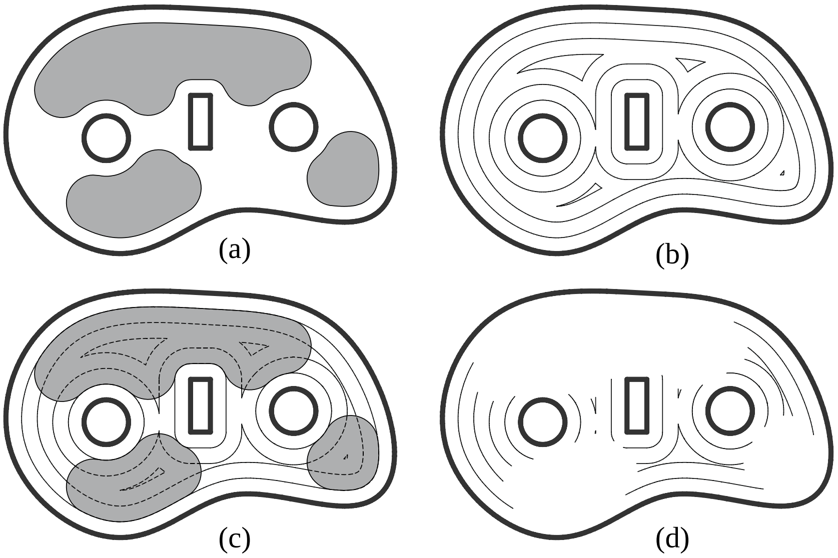


Fig. 6 New cavity decomposition. (a) line 2, (b) line 3, (c) line 4, (d) line 5 in Algorithm 4.

Algorithm 4. DecCavity2(A_1, A_2, d)

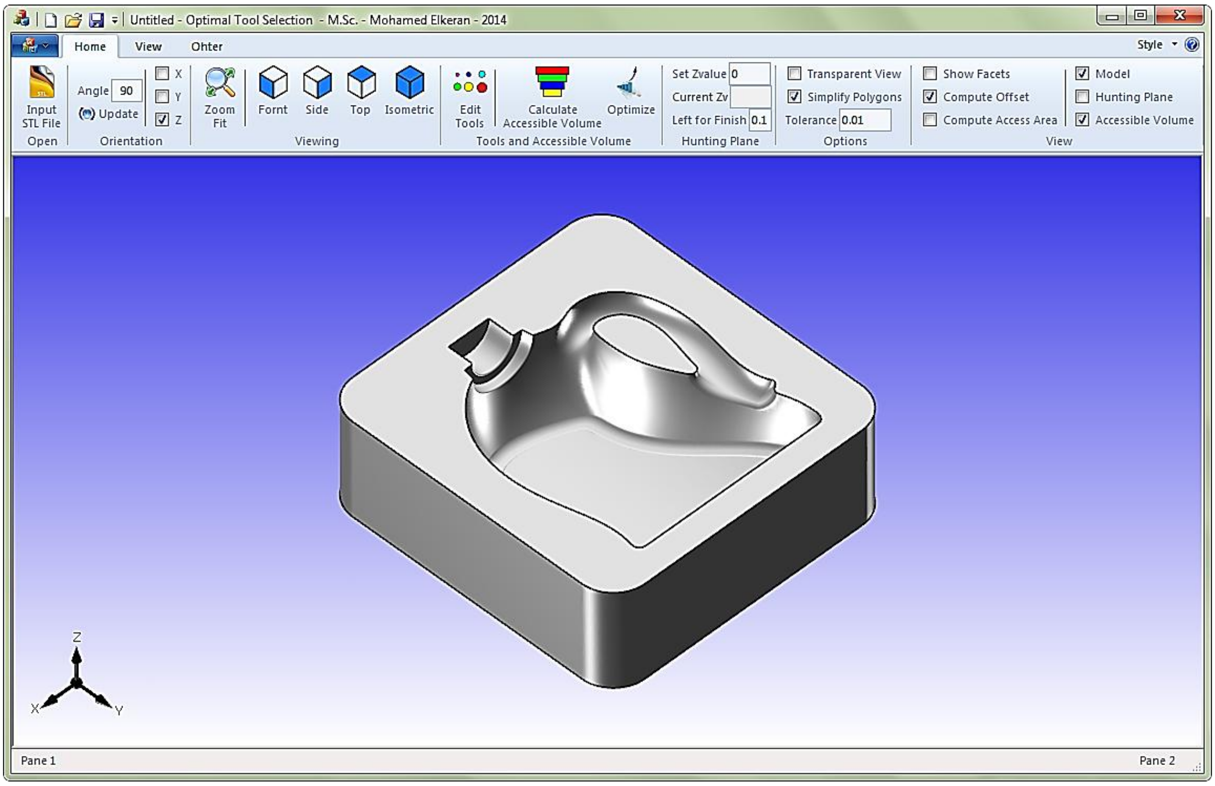
```
// New cavity decomposition
1. $r = d/2$      // radius of tool $t_2$
2. $X = $ offset($A_1$, $-r$)
3. $TPC = $ComputeToolPath($A_2$, $d$)
4. $TTPC = $ TrimTPC($X$, $TPC$)
5. **return**$TTPC$
```

5. Implementation

Through this work, the algorithms of Voronoi diagram, offsetting, geometric intersections, and cavity decomposition are implemented in Visual C++ supported with Open Graphics Library (OpenGL) for part rendering and manipulation. The program has been compiledby Microsoft Visual Studio 2013. The working environment is a PC with core i7 2.2 GHz processor and 6 GB memory.The input file is a part cavity in stereo lithography (STL) format in which the part is represented by a set of triangular facets. Figure 7 shows a snapshot taken from the program Integrated Development Environment (IDE) and a part with freeform cavity.

Figure 8 shows an example of using only one small tool for machining the entire cavity of the part shown in Fig. 7. Figures 9 and 10 show another example ofusing two tools for the same part. The large tool accessible volume and tool path are shown layer by layer inFigs. 9(a-f). The small tool accessible volume and tool path are shown layer by layer in Figs. 10(a-f). In the second example the small tool is the same size as the one used in the first example.

Using only a small tool to machine the whole cavity (Fig. 8) yields a long tool path and thus a low material removal rate. On the other hand, using two tools to machine the whole cavity (Figs. 9 &10), yields a shorter tool path and higher material removal rate. In the two examples the small tools are of the same diameter. Thus, using a set multiple tools will have a great impact on saving machining time and cost. This also will extend the tools’ life since each tool can only remove a portion of cavity material leaving unmachined areas for its subsequent tools.



Untitled - Optimal Tool Selection - M.Sc. - Mohamed Elkeran - 2014

Angle 90

☐ X
☐ Y
☑ Z

Set Zvalue 0
Current Zv
Left for Finish 0.1

☐ Transparent View
☑ Simplify Polygons
Tolerance 0.01

☐ Show Facets
☑ Compute Offset
☐ Compute Access Area

☑ Model
☐ Hunting Plane
☑ Accessible Volume

Fig. 7 A snapshot from the developed software Integrated Development Environment (IDE).

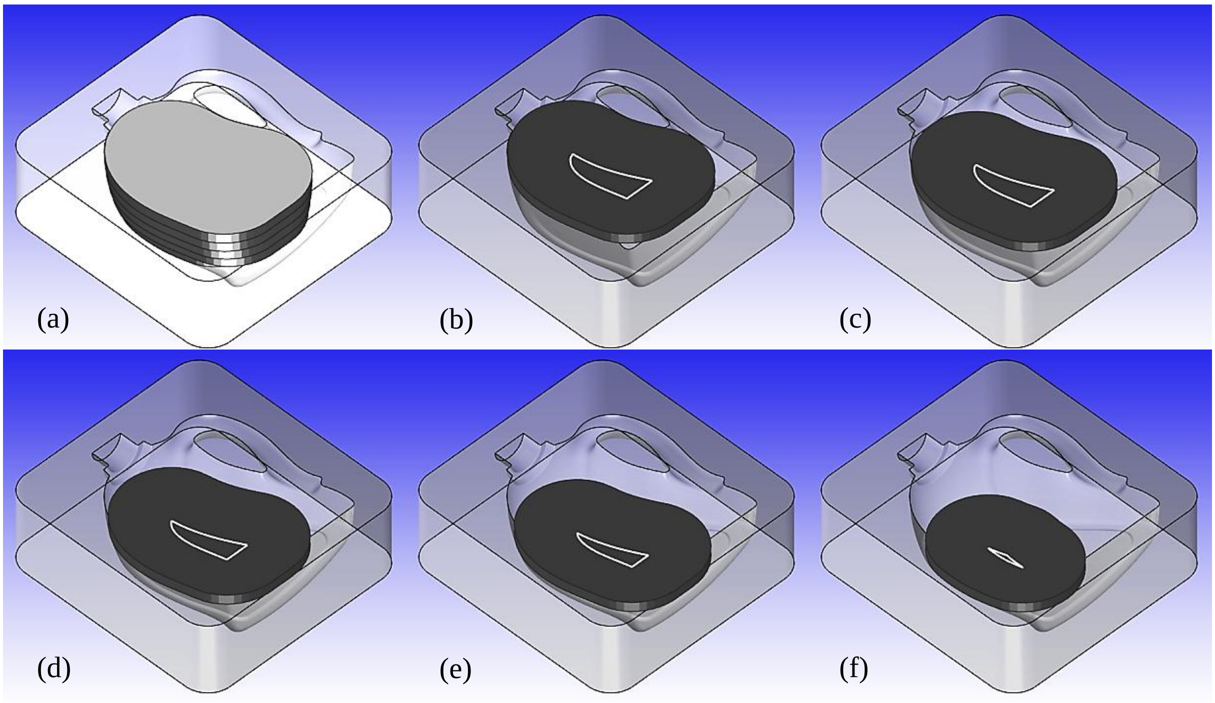


Fig. 9 An example of using two tools machine the entire cavity. *The first tool accessible volume and tool path*.(a) Tool accessible volume. (b) First machining layer.(c) Second machining layer. (d) Third machining layer. (e) Fourth machining layer. (f) Fifth machining layer.

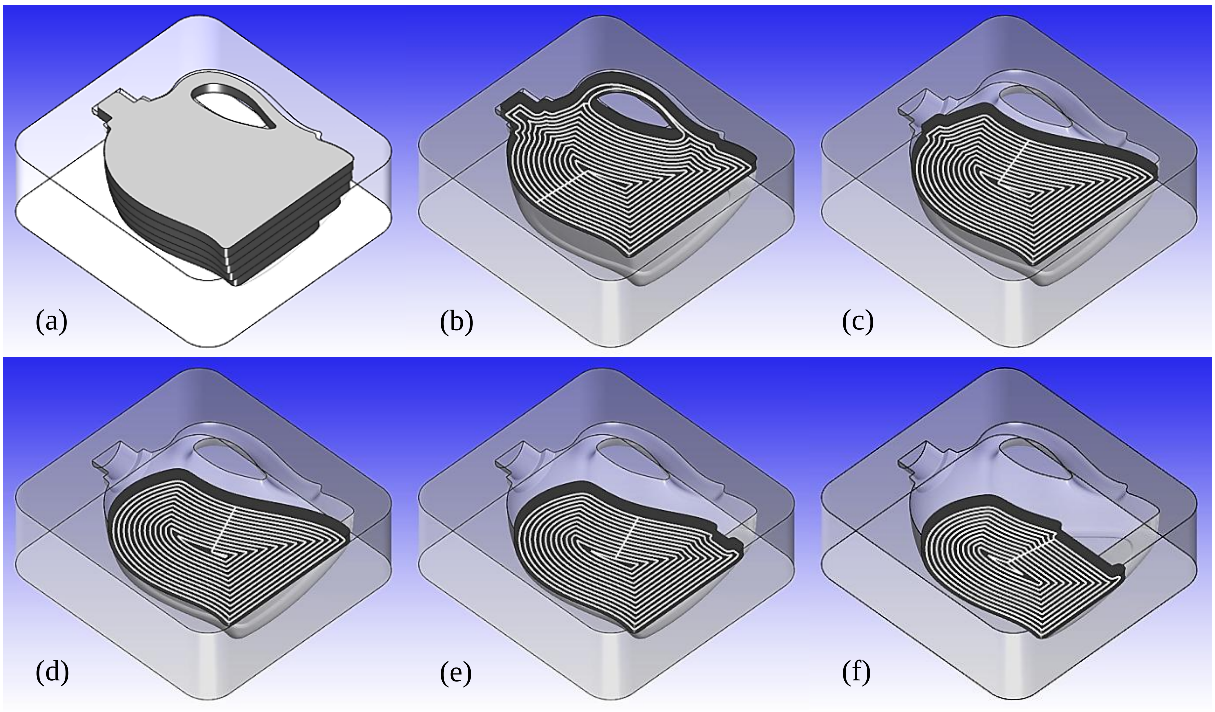


Fig. 8 An example of using only one small tool to machine the entire cavity. Tool path is represented by white contours. (a) Tool accessible volume. (b) First machining layer.(c) Second machining layer. (d) Third machining layer. (e) Fourth machining layer. (f) Fifth machining layer.

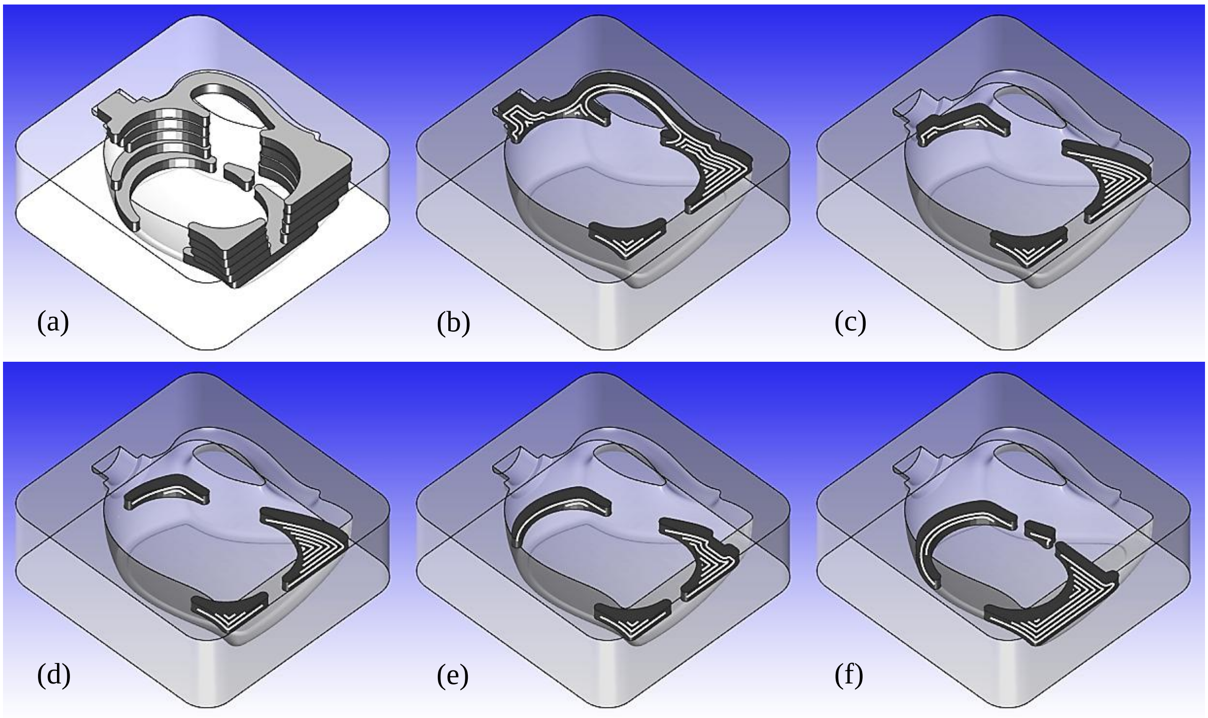


Fig. 10 An example of using two tools machine the entire cavity. *The second tool accessible volume and tool path.* (a) Tool accessible volume. (b) First machining layer.(c) Second machining layer. (d) Third machining layer. (e) Fourth machining layer. (f) Fifth machining layer.

6. Computational results

This section gives details about the application results of the proposed method in this work. The proposed method has been validated for computational accuracy and reliability against commercial software which is CAMWorks [19]. For conducting computational experiments, the selected part material is AISI A2 cold work tool steel. The recommended cutting conditions are selected according to the Sandvik tools manufacturer. Table 1 shows the tools database on which computational experiments are conducted. The machine cost per hour is set to 50$ and the rapid move feed is 30 m/min. The tools database is not constrained to any number of tools and is left for user input.The part used for experiments is the one shown in Fig. 7.

Table 1: Tool database

Tool	Type	No. of teeth	Diameter (mm)	Width of cut (mm)	Depth of cut (mm)	Feed (mm/tooth)	Cutting Speed (m/min)	Price ($)
T1	Indexable insert	7	80	40	4	0.2	80	168
T2	Indexable insert	5	50	25	4	0.2	80	120
T3	Indexable insert	3	36	18	3	0.2	80	72
T4	Indexable insert	3	25	12.5	2.5	0.2	80	72
T5	Solid carbide	4	20	10	2	0.25	60	387
T6	Solid carbide	4	16	8	1.6	0.2	60	250
T7	Solid carbide	4	12	6	1.2	0.08	60	155
T8	Solid carbide	4	10	5	1	0.03	60	120

Table 2 presents validation results ofthe developed software with the CAMWorks software. The sets of tools in Table 2 aresorted according to increasing cost.

CAMWorks software computes the machining time for each set of tools, however it does not compute the total cost of machining and tooling. In order to compare the total cost computations for the developed software and CAMWorks software, the CAMWorks machining time computations is transformed to cost results by using Eqs. (2-4).

The differences in machining time computations between the developed software and CAMWorks software are less than 10 %. This is due to different software implementations and tradeoffs and it is always the case even though between different commercial software products. These differences also are results of how the software controls the input machining parameters which are lead in/out, tool path strategies, tool path stepfeed, tool path linking, rapid motions, rest milling options, etc.

The proposed method of cavity decomposition has been compared with the previous method [7, 8] for CPU time as shown in Table 3. The percentage of improvement of the proposed method ranges from about 30 to 50%. All results including machining time calculations and CPU time measurements are done on the same computer. The last row in Table 3 shows only one tool used for machining the whole cavity. Hence, no cavity decomposition is needed and this explains the no improvement (0%) between the new and previous cavity decomposition methods.

The optimal tools sequence satisfying the minimum machining and tooling cost for the presented example in Fig. 7 has tool diameters of ***80, 50, 36, 25, 20, 10***. Figure 11 shows the shape of rough machining after each tool of the optimal sequence.

Table 2: Machining Time and Cost Validation

Tool Sequence Diameters	MT1* (min)	MT2** (min)	Absolute MT Difference (%)	TC1+ ($)	TC2++ ($)	Absolute TC Difference (%)
80, 50, 36, 25, 20, 10	105	111	5.4	**307**	**325**	5.5
80, 36, 25, 20, 10	104	110	5.5	308	329	6.4
80, 25, 20, 10	103	107	3.7	328	335	2.1
80, 50, 25, 20, 10	106	110	3.6	330	343	3.8
80, 20, 10	107	106	0.9	387	382	1.3
80, 50, 36, 25, 20, 16, 10	116	118	1.7	392	389	0.8
36, 20, 10	138	139	0.7	396	394	0.5
50, 25, 10	148	150	1.3	402	404	0.5
80, 50, 36, 25, 20, 16, 12, 10	136	139	2.2	464	466	0.4
36, 10	200	192	4.0	515	491	4.7
50, 10	219	214	2.3	618	599	3.1
20, 10	182	173	4.9	964	937	2.8
16, 10	417	390	6.5	1901	1808	4.9
12, 10	979	948	3.2	3308	3204	3.1
10	1491	1470	1.4	4225	4165	1.4

* **MT1**: Machining Time for the developed software.

** **MT2**: Machining Time for the CAMWorks software.

\+ **TC1**: Total Cost for the developed software.

++**TC2**: Total Cost for the CAMWorks software.

Table 3: CPU time of the new cavity decomposition technique versus the previous one [7, 8]

Tool Sequence Diameters	New Cavity Decomposition CPU Time (millisecond)	Previous Cavity Decomposition CPU Time (millisecond)	Improvement (%)
80, 50, 36, 25, 20, 10	4388	7049	38
80, 36, 25, 20, 10	3851	6386	40
80, 25, 20, 10	3135	5476	43
80, 50, 25, 20, 10	3358	5648	41
80, 20, 10	2075	3400	39
80, 50, 36, 25, 20, 16, 10	5732	9142	37
36, 20, 10	2216	3935	44
50, 25, 10	1890	3172	40
80, 50, 36, 25, 20, 16, 12, 10	8056	15012	46
36, 10	1276	2112	40
50, 10	1186	2188	46
20, 10	1538	2490	38
16, 10	1698	2688	37
12, 10	1996	3102	36
10	1160	1160	0

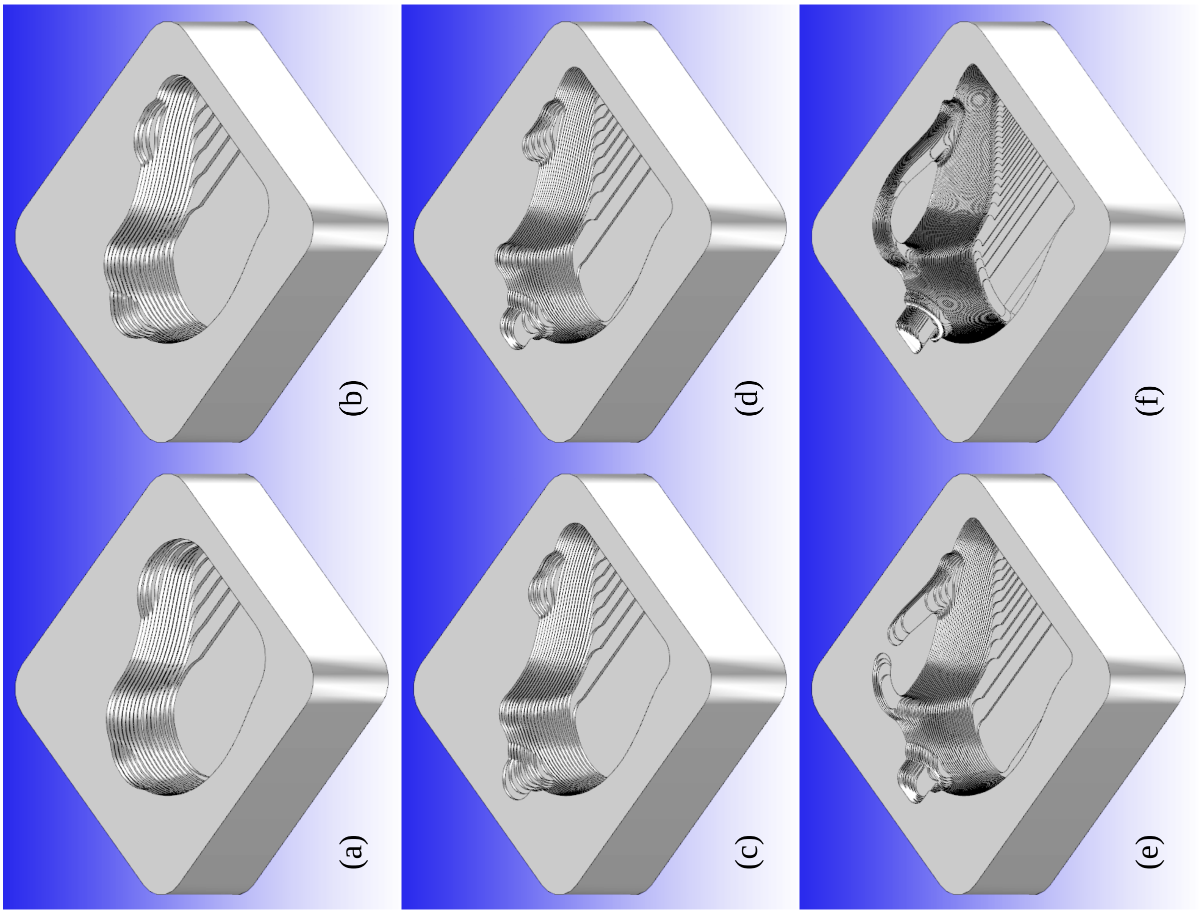


Fig. 11 Shapes of cavity rough machining for the optimal tool sequence. (a) Shape after first tool (d = 80 mm). (b) Shape after second tool (d = 50 mm). (c) Shape after third tool (d = 36 mm). (d) Shape after fourth tool (d = 25 mm). (e) Shape after fifth tool (d = 20 mm). (f) Shape after sixth tool (d = 10 mm).

7. Conclusions

This paper introduces a new method for optimal tools selectionfor rough machining of 3D freeform cavities.This method computes the machining time based on real tool paths generation considering all technological aspects. The cavity decomposition is the most complex and time consuming operation of all computations especially when a large tool library is available. To reduce the geometrical complexity of the problem, a new cavity decomposition technique has been proposed.

Utilizing the proposed cavity decomposition technique with exact tool paths computation, optimal tools selection can be made accurately and fast. An integrated software has been implemented in Visual C++ supported with OpenGL graphics toolkit for part rendering and manipulation. The developed software handles all the geometrical operations as part contours extraction via hunting planes, tool accessible areas computations, tool paths generation, cavity decomposition, and finally automatically determines the optimal tool sequence.

The developed software results have been validated with CAMWorks commercial software for accuracy and reliability.Moreover, the results prove that the new method of cavity decomposition improves the computing time of previous method by about 30 to 50 %.

References

[1] Chen, Z.C., Fu, Q., “An optimal approach to multiple tool selection and their numerical control path generation for aggressive rough machining of pockets with free-form boundaries,”Computer-Aided Design, Vol. 43, No.6, 2011, pp. 651–663.

[2] Lee, Y.S., Choi, B.K., Chang, T.C., “Cut distribution and cutter selection for sculptured surface cavity machining” International Journal of Production Research, Vol. 30, No. 6, 1992, pp. 1447–1470.

[3] Lee, Y.S., Chang, T.C., “Application of computational geometry in optimizing 2.5d and 3d nc surface machining” Computers in Industry, Vol. 26, No. 1, 1995, pp. 41–59.

[4] Bala, M., Chang, T., “Automatic cutter selection and cutter path generation for prismatic parts” International Journal of Production Research, Vol. 29, No. 11, 1991, pp. 2163–2176.

[5] Chen, Y. Lee, Y.S., Fang, S.C. “Optimal cutter selection and machining plane determination for process planning” Journal of Manufacturing Systems, Vol. 17, No. 5, 1998, pp. 371–388.

[6] Veeramani, D., Gau, Y., “Cutter-path generation using multiple cutting-tool sizes for 2-1/2D pocket machining” IIE Transactions, Vol. 32, No.7, 2000, pp. 661–675.

[7] D’Souza, R.M., Sequin, C., Wright, P.K., “Automated tool sequence selection for 3-axis machining of free-form pockets” Computer-Aided Design, Vol. 36, No. 7, 2004, pp. 595–605.

[8] Ahmad, Z., Rahmani, K., D’Souza, R.M., “Applications of genetic algorithms in process planning: tool sequence selection for 2.5-axis pocket machining” Journal of Intelligent Manufacturing, Vol. 21, No. 4, 2010, pp. 461–470.

[9] Ramaswami, H., Shaw R.S., Anand, S., “Selection of optimal set of cutting tools for machining of polygonal pockets with islands” International Journal of Advanced Manufacturing Technology, Vol. 53, No. 9-12, 2011, pp. 963-977.

[10] Zhang, Y., Ge, L., “Selecting optimal set of tool sequences for machining of multiple pockets” International Journal of Advanced Manufacturing Technology, Vol. 42, No. 3–4, 2009, pp. 233–241.

[11] Hansen A., Arbab F., “An algorithm for generating NC tool path for arbitrary shaped pockets with islands” ACM Transactions on Graphics, Vol. 11, No. 2, 1992, pp.152–82.

[12] Choi, B.K., Park, S.C., “A pair-wise offset algorithm for 2D point-sequence curve” Computer-Aided Design, Vol. 31, 1999, pp. 735–45.

[13] Park S.C., Choi B.K., “Uncut free pocketing tool-paths generation using pair-wise offset algorithm” Computer-Aided Design, Vol. 33, 2001, pp. 739–746.

[14] Persson H., “NC machining of arbitrarily shaped pockets” Computer-Aided Design, Vol. 10, 1978, pp. 169–174.

[15] Park S.C, Chung Y.C., “Offset tool-path linking for pocket machining” Computer-Aided Design, Vol. 34, 2002, pp. 299–308.

[16] Held M., Lukács G., AndorL., “Pocket machining based on contour-parallel tool paths generated by means of proximity maps” Computer-Aided Design, Vol.26, 1994, 189–203.

[17] Held. M., “VRONI: An Engineering Approach to the Reliable and Efficient Computation of Voronoi Diagrams of Points and Line Segments” Computational Geometry: Theory and Application, Vol. 18, No. 2, 2001, pp. 95–123.

[18] Held M., Huber, M., “Topology-Oriented Incremental Computation of Voronoi Diagrams of Circular Arcs and Straight-Line Segments” Computer-Aided Design, Vol. 41, No. 5, 2009, pp. 327–338.

[19] http://www.camworks.com/