J Intell Manuf (2006) 17:363–372
DOI 10.1007/s10845-005-0009-6

Selecting an optimal tool sequence for 2.5D pocket machining while considering tool holder collisions

Roshan M. D'Souza

Received: February 2005 / Accepted: September 2005

Abstract Milling using a sequence of tools has become very attractive with the advent of rapid tool change mechanisms in modern CNC machines. However, the commercial CAM systems used to generate G&M code rely on experienced process planners to select a good tool sequence. When a tool sequence is selected and tool paths are generated, NC-verify systems are used to check the tool paths for tool holder collisions. If tool holder collisions are detected, the part has to be re-planned ab-initio. In this paper, we describe a method to select an optimal tool sequence by formulating the problem under certain assumptions as the shortest path solution to a single source directed acyclic graph. Also described is a method to incorporate tool holder solution in the context of selecting an optimal tool sequence. Examples have been worked out to illustrate the workings of the algorithm.

Keywords 2.5-Axis machining · Tool sequence selection · Process-planning

Nomenclature

$f(p,h)$	Removal volume represented by a planar bottom face $p$and a depth h
t_m	End milling cutter with diameter d_m and length l_m
$A_m(f)$	Nominal accessible area of a tool t_m within the face p of feature f without taking into account tool holder.
$D_{mn}(f)$	Clean up or decomposed area of a tool t_n without considering the tool holder within the face $p$of feature f after the larger tool t_m has machined to the extent of $A_m(f)$
$A'_m(f)$	Accessible area of a tool t_m within the face p of feature f while considering tool holders
$D'_{mn}(f)$	Decomposed area of a tool t_m within the face p of feature f while considering tool holders
Offset(p,d)	Function to create a planar offset of a face p through a signed distance d

R. M. D'Souza
Department of Mechanical Engineering-Engineering Mechanics,
Michigan Technological University,
1400 Townsend Drive,
Houghton, MI 49931, USA
e-mail: rmdsouza@mtu.edu

Introduction

Process planning for milling consists of three main tasks. The first identifies removal volumes/machining features and various access directions for machining them (Gao & Shah, 1998; Regli, 1995). The second clusters them into setups based on the feasibility of machining these removal volumes in a particular direction and clamping the stock (Echave & Shah, 1999; Kannan & Wright, 2001). The final task consists of selecting appropriate tool sequences.

Current state of the art process planning systems (SURFCAM, 2002; MASTERCAM, 2002) allow users to select two or more tools for machining the pockets. The actual tool sequence selection is left to the human process planner. The process of time or cost optimization is one of trial and error where complete process planning has to be done in order to validate the plan and calculate costs using NC-Verify systems. Tool holder collision is another serious issue. NC-verify systems can be used to detect tool holder collisions. If tool holder collisions are detected, the only solution is to eliminate the offending tools and re-plan ab-initio.

The issue of selecting tool sequences has been addressed by several researchers (Arya, Cheng, & Mount, 1001.1; Bala & Chang, 1991; Chen, Lee, & Fang, 1998; Joo & Cho, XXXX; Kunwoo, Kim, & Hong, 1994; Lee & Chang, 1995; Lim, Corney, Ritchie, & Clark, 2000; Veeramani & Gau, 1997; Yao, Gupta, & Nau, 2001). However, none of these researchers have incorporated tool holder collision handling in their algorithms.

In this paper, a systematic method has been developed to select an optimal tool sequence. The problem of selecting an optimal tool sequence has been formulated as the shortest path solution to a single-source, single-sink directed acyclic graph under certain assumptions. The nodes in the graph represent the shape of the removal volume after the tool in the node is done machining to the extent of its accessible area at each depth of cut. The edges represent the cost of machining. Methods have been described to find accessible areas in presence of stock boundary open-edges, and finding decomposed sub-pockets. Re-interpretation of the assumption used in the formulation of the problem results in an elegant method to handle tool holder collisions.

Problem statement

The objective of this research is to find the cheapest tool sequence T_{opt} to machine a feature/pocket $f(p, h)$ given a candidate set of tools $T = \{t_1, t_2, \ldots, t_n\}$ with diameters $\{d_1 > d_2 \cdots > d_n\}$. Also given is the intermediate stock I. The cheapest tool sequence must be such that no tool $t_m \in T_{\mathrm{opt}}$ causes tool holder collision with the intermediate stock while machining the region assigned to it.

Tool sequence selection formulation

In this section, we present the basic graph algorithm formulation of the tool sequence selection problem. The material presented is reproduced from an earlier publication by the author (D'Souza, Wright, & Séquin, 2001) for the sake of completeness.

Given a set of cutting tools and a 2.5D feature, the first step in tool sequence selection is the determination of the region that each tool can machine in the feature without gouging.

This region is represented as an area called accessible area that the tool will traverse at each depth of cut in the pocket. The next step is the determination of feature decomposition. Given two tools of different diameters, the decomposed area is the area that the smaller tool traverses at each depth of cut for clean up machining after the larger tool is done machining whatever it can reach in the pocket. Given the accessible areas of various tools, and the decomposed areas of all possible tool pairs, the problem of selecting the optimal tool sequence is then formulated as the shortest path search in a single source, single sink, directed acyclic graph.

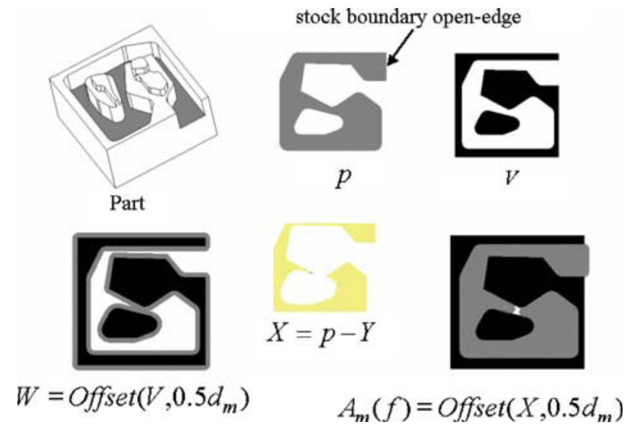

Fig. 1 Accessible area

Accessible area

Accessible area $A_m(f)$ in feature f for a tool t_m is defined as the area within the pocket face p that the tool can reach without gouging. This is the area that the tool traverses at every depth of cut to machine whatever it can in the feature. Accessible areas must effectively cover stock boundary open-edges for complete machining. Stock boundary open-edges are edges shared by p and a section of the intermediate stock I at the same level as p. The procedure in algorithm 1 is followed to calculate $A_m(f)$ given p and d_m, the diameter of the tool t_m. Figure 1 illustrates an example. Note that even though the tool traverses over the area $A_m(f)$, the material removal within the feature is only to the extent of $A_m(f) \cap p$.

Algorithm 1: *Accessible Area calculation without considering tool holder collisions*

PROCEDURE ACCESSIBLE_AREA (f, t_m, I)
  $U \leftarrow$ section of I at the level of p
  $V \leftarrow U - p$
  $W \leftarrow Offset\ (V, 0.5d_m)$
  $X \leftarrow p - W$
  if $(X == NULL)$ /* tool cannot enter
    p without gouging */
    $A_m(f) \leftarrow NULL$
  else
    $A_m(f) \leftarrow Offset\ (X, 0.5d_m)$
  endif

An important property of the accessible areas results from the fact that tools of smaller diameter can reach a larger area in the pocket as compared to tools of larger diameter. In other words,

$$\forall t_m, t_n : d_m > d_n \Leftrightarrow A_m(f) \cap p \subseteq A_n(f) \cap p. \quad (1)$$

Decomposed area

Consider a case where two tools $t_m, t_n : d_m > d_n$ are used to machine a pocket f. Tool t_m will machine to the extent of its accessible area $A_m(f) \cap p$ at each depth of cut while t_n will machine the remaining material to the extent of the area given by $\{A_n(f) - A_m(f)\} \cap p$ at each depth of cut. Since the contour $\{A_n(f) - A_m(f)\} \cap p$ has sharp corners, using it to generate tool paths for t_n will result in incomplete machining at these sharp corners. This area has to be extended suitably such that complete machining takes place. This extended area $D_{mn}(f)$ is called the decomposed sub-pocket. The procedure in algorithm 2 is used to calculate the decomposed sub-pocket. Figure 2 shows an example.
Note that the following relations hold:

$$\forall t_m, t_n : d_m > d_n \Leftrightarrow A_n(f) = A_m(f) \cup D_{mn}(f). \quad (2)$$

Algorithm 2: *Finding Decomposed Area Without Considering Tool Holder Collisions*

PROCEDURE DECOMPOSED_AREA ($A_m(f)$, $A_n(f), t_n$)
 $B_{mn} \leftarrow Offset(A_m(f), 0.5d_n)$
 $B_{nn} \leftarrow Offset(A_n(f), 0.5d_n)$
 $C \leftarrow B_{nn} - B_{mn}$
 if ($C == NULL$) /* larger tool can machine whatever smaller tool can reach */
  $D_{mn}(f) \leftarrow NULL$
 else
  $D_{mn}(f) \leftarrow Offset(C, 0.5d_n)$
 end if

Problem formulation

Lemma: *"The shape of the left over area at each depth of cut after a particular tool is done machining is independent of any larger tool used before it, provided each tool machines to the extent of its accessible area within the pocket."*

Proof: Consider a pocket $f(p, h)$ and tools $t_m, t_n : d_m > d_n$. The shape of the left over area at each depth of cut S_n after tool t_n is done machining to the extent of its accessible area $A_n(f)$ is given by:

$$S_n = p - \{A_n(f) \cap p\}. \quad (3)$$

Suppose tool t_m is used before tool t_n, t_m will remove material to the extent $A_m(f) \cap p$ within the pocket. Tool t_n will machine whatever is left, to the extent of $A_n(f) \cap p$. Therefore, the shape of the left over area at each depth of cut S_{mn} after both tools are done machining is given by:

$$\begin{aligned} S_{mn} &= p - \{A_m(f) \cap p\} - \{A_n(f) \cap p\} \\ &= p - \{\{A_m(f) \cap p\} \cup \{A_n(f) \cap p\}\}. \end{aligned} \quad (4)$$

Using Eq. 1, we can rewrite this as:

$$S_{mn} = p - \{A_n(f) \cap p\} = S_n. \quad (5)$$

This is because $\{A_m(f) \cap p\} \cup \{A_n(f) \cap p\} = \{A_n(f) \cap p\}$ since $\{A_m(f) \cap p\} \subseteq \{A_n(f) \cap p\}$.

Now consider a sequence of tools $t_1 \rightarrow t_2 \rightarrow \cdots t_k \cdots \rightarrow t_n$, with diameters $d_1 > d_2 > \cdots d_k \cdots > d_n$ The shape of the left over area after all of these tools are done machining to the extent of their respective accessible areas is given by:

$$S_{1,2\ldots n} = p - \cup_{k=1}^{n}\{A_k(f) \cap p\}. \quad (6)$$

However, we know that:

$$\{A_k(f) \cap p\} \subseteq \{A_l(f) \cap p\}, \forall d_k > d_l. \quad (7)$$

Hence Eq. 6 reduces to:

$$S_{1,2\ldots n} = S_{2,3\ldots n} = S_{3,4\ldots n} \ldots S_n. \quad (8)$$

Hence the proof.

Figure 3 shows an example. In the first case (Fig. 3b), tools of diameter 0.4″ is used before 0.25″. In the second case (Fig. 3c), 0.5″ is used before 0.25″. However, the shape after 0.25″ is done machining in both cases is the same.

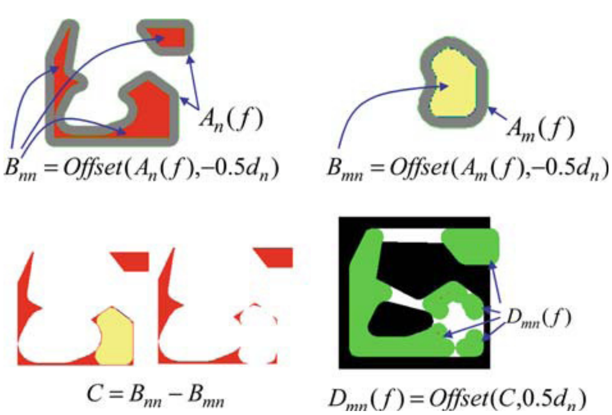

Fig. 2 Decomposed area

Graph algorithm

Consider a set of tools $T = \{t_1, t_2 \ldots t_n\}$ and a feature $f(p, h)$ such that

$$\begin{aligned} &d_1 > d_2 \cdots > d_n, \\ &A_1(f), A_2(f), \ldots, A_{(n-1)}(f) \neq \phi, \\ &A_n(f) \neq \phi. \end{aligned} \quad (9)$$

This means that each of the tools can reach at least some area within p at each depth of cut, and the smallest tool t_n can reach everywhere in p without gouging. All possible tool sequences consist of all combinations of tool sequences using one tool,

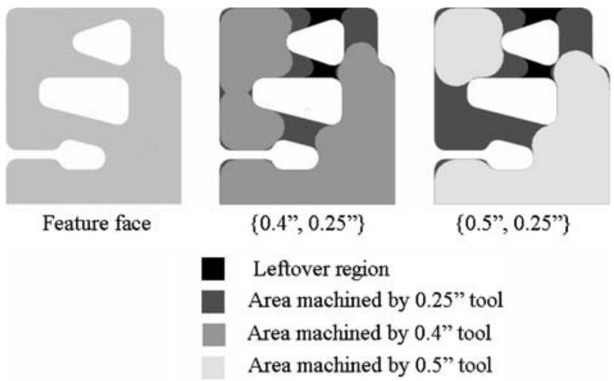

Fig. 3 Independence of shape

two tools, up to sequences using n tools. Since only smallest tool t_n can reach everywhere in the pocket, the number of tool sequences consisting of one tool is 1. For two-tool tool sequences, other than t_n, we can select one tool from the remaining $(n-1)$ tools. Therefore, the total number of sequences is ${}^{(n-1)}C_1$. The total number of tool sequences consisting of $k+1$ tools is given by ${}^{(n-1)}C_k$. The total number of tool sequences consisting of $1, 2, \ldots k \ldots n$ tools is given by:

$$
\begin{aligned}
&{}^{(n-1)}C_1 +{}^{(n-1)} C_2 + \cdots\cdots \\
&{}^{(n-1)}C_k \cdots +{}^{(n-1)} C_{(n-1)} = 2^{(n-1)}.
\end{aligned} \tag{10}
$$

All possible tool sequences may be represented as a set of directed linked lists. The nodes in the linked lists represent the shape of the pocket after the tool named in the node is done machining. The edges represent the cost of machining for the tool named in the tail node of the edge after all preceding tools in the list are done machining, assuming each tool machines to the extent of its accessible area.

If we use just one tool, the tool that will be used is t_n. The number of edges (and corresponding tool pairs) evaluated will be 1. Let us consider all possible tool sequences which use one other tool in addition to the final tool t_n. The number of such sequences is $(n-1)$. The number of edges that must be evaluated is $2(n-1)$. One edge connects the larger tool to the start node. The other connects this tool to t_n. Now consider tool sequences having three tools. The number of such sequences is ${}^{(n-1)}C_2$. The number of edges in each of these tool sequences is $3\left({}^{(n-1)}C_2\right)$. In each of the 3-tool tool sequences, part of the sequence consisting of the edges leading into the largest tool would have already been evaluated while evaluating all possible 2-tool tool sequences. If this information is reused, only $2\left({}^{(n-1)}C_2\right)$ edges will have to be evaluated. If we consider all possible 4-tool tool sequences, part of the sequence consisting of the first two largest tool would have been evaluated while evaluating all possible 3-tool tool sequences. Therefore, we will have to evaluate $2\left({}^{(n-1)}C_3\right)$ edges. Clearly, total number of edges we need to evaluate is thus given by:

$$
\begin{aligned}
&1 + 2\left({}^{(n-1)}C_2\right) + 2\left({}^{(n-1)}C_3\right) \cdots 2\left({}^{(n-1)}C_k\right) \cdots \\
&\quad +2\left({}^{(n-1)}C_{(n-1)}\right) = 2^{(n)} - 1.
\end{aligned} \tag{11}
$$

Clearly, this becomes intractable even for a modest number of available tools. This is because the evaluation of the weight of an edge involves several boolean operations and 2D offset operation for pocket decomposition and generation of tool paths. For example, a set of 10 available tools will result in evaluation of 1023 edges.

Using the lemma in "Problem formulation", all nodes with the same tool named in them can be collapsed into a single node. This is because each of these nodes essentially represents the same shape. This converts the set of directed linked list into a single-source, single-sink, directed acyclic graph. The number of edges to be evaluated reduces to $0.5n(n+1)$. Figure 4 shows this reduction in complexity for four tools. Moreover, every edge in the graph can be evaluated independent of which path it occurs. For example the edge $e_{\{2\to4\}}$ can be independently evaluated whether it is part of the tool sequence $0 \to t_2 \to t_4$ or is part of $0 \to t_1 \to t_2 \to t_4$. The cheapest tool sequence is the shortest path in the graph given by Djikstra's algorithm (D'Souza, Wright, & Séquin, 2001). The weight associated with the edge W_{ij}, which is the total cost of machining is given by:

$$
W_{ij} = \left(\frac{l_m}{F_m} + \frac{l_a}{F_r} + T_{\text{ch}}\right)\frac{h}{60} + \frac{l_m \times C_{\text{tool}}}{F_m \times T_{\text{life}}}. \tag{12}
$$

where, l_m is the tool path length for machining, F_m is the machining feed rate, l_a is the air-path length, F_r is the rapid traverse rate, h the hourly overhead rate, C_{tool} the cost of buying and installing a new cutting tool in the machine tool, T_{tool} is the tool life for the prescribed cutting parameters. Equation 12 accounts not only for machine use, but also tool wear. Alternatively, if machining time is to be optimized, the weight is given by:

$$
W_{ij} = \left(\frac{l_m}{F_m} + \frac{l_a}{F_r} + T_{\text{ch}}\right). \tag{13}
$$

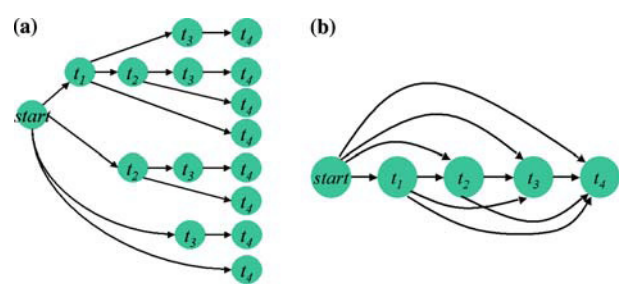

Fig. 4 Graph representation of tool sequences (**a**) All possible tool sequences (**b**) Reduced tool sequence graph

Tool sequence selection in presence of tool holder collisions

The analysis presented in the preceding sections assumes that each of the tools in the optimal sequence can machine the entire region assigned to it without tool holder collisions. For example, if $T_{\mathrm{opt}} = \{t_1, t_4, t_6\}$ for some feature $f(p, h)$ the tools t_1, t_4, t_6 would machine areas up to $A_1(f)$, $A_4(f)$, $A_6(f)$, respectively, at each depth of cut. Suppose there exists a region $r \subset A_1(f)$ such that the tool holder of t_1 interferes with the intermediate stock I when the tool machines any part of r then this tool sequence is invalid.

One might argue that letting each tool machine only the regions where tool holder collision does not occur will solve the problem. However, this approach will not let us simplify the problem as in sub-section “Graph algorithm”. This is because the shape of the stock after a particular tool is done machining will not be independent of tools used before it. Hence the number of tool sequence graph edges to be evaluated will be equal to $2^n - 1$.

The proof of the above assertion is illustrated with the help of an example. Consider a set of two tools $t_1, t_2 : d_1 > d_2$ and a pocket $f(p, h)$. Let there be an area r such that $r \subset \{A_2(f) \cap p\}, r \subset \{A_1(f) \cap p\}$. Note the fact that $\{A_1(f) \cap p\} \subset \{A_2(f) \cap p\}$. Let there be tool holder collision everywhere in r for t_2.

If the approach explained above were to be used, the shape of the leftover area at each depth of S'_2 after only t_2 is done machining is given by:

$$S'_2 = p - (\{A_2(f) \cap p\} - r)\,. \tag{14}$$

If instead t_1 was used first, and then t_2 was used, the shape of the pocket S'_{12} would be:

$$\begin{aligned} S'_{12} &= p - \{A_1(f) \cap p\} - (\{A_2(f) \cap p\} - r) \\ &= p - \{A_1(f) \cap p\} \cup r - \{A_2(f) \cap p\} \\ &= p - \{A_1(f) \cap p\} - \{A_2(f) \cap p\}. \end{aligned} \tag{15}$$

This is because $r \subset \{A_1(f) \cap p\}$. Using equation 1, this further reduces as:

$$S'_{12} = p - \{A_2(f) \cap p\}. \tag{16}$$

Clearly, $S'_{12} \neq S'_2$. In other words, the shape of the leftover area after a tool is done machining is dependent on the larger tools used before it, if the smaller tool was used only in regions where it did not have tool holder collisions.

Reinterpreting assumption

Consider a situation where there exist regions $r_1, r_2 : r_1 \cup r_2 = A_2(f)$. Instead of using t_2 to machine the entire region $A_2(f)$, we will use some other tool t_x to machine r_1 and t_2 to machine r_2. It is assumed here that r_1 is entirely accessible to t_x. The shape of the pocket S''_2 after both these tools are done machining is given by:

$$\begin{aligned} S''_2 &= p - \{r_1 \cap p\} - \{r_2 \cap p\} \\ &= p - \{(r_1 \cup r_2) \cap p\} \\ &= p - \{A_2(f) \cap p\}. \end{aligned} \tag{17}$$

Now consider the case where two tools $t_1, t_2 : d_1 > d_2$are used successively. Therefore, t_1 will machine to the extent of $A_1(f)$. Normally, the second tool t_2 would machine to the extent of $D_{12}(f)$. Let there be two regions r'_1, r'_2such that $r'_1 \subset r_1$ and $r'_2 \subset r_2$, and $r'_1 \cup r'_2 = D_{12}(f)$ Instead of using t_2 to machine the entire region $D_{12}(f)$, we will use t_x to machine r'_1 and tool t_2 to machine r'_2. The shape of the pocket S''_{12} after these three tools are done machining is given by:

$$\begin{aligned} S''_{12} &= p - \{A_1(f) \cap p\} - \{r'_1 \cap p\} - \{r'_2 \cap p\} \\ &= p - \{A_1(f) \cap p\} - \left(\{r'_1 \cup r'_2\} \cap p\right) \\ &= p - \{A_1(f) \cap p\} - \{D_{12}(f) \cap p\} \\ &= p - (\{A_1(f) \cup D_{12}(f)\} \cap p)\,. \end{aligned} \tag{18}$$

Using Eq. 2:

$$S''_{12} = p - \{A_2(f) \cap p\} = S''_2 = S_{12} = S_2. \tag{19}$$

It is clear that the shape of the leftover area is not affected by *which tools* are used. It is affected only by *what is machined*. In the above analysis, area was machined to the extent of $A_2(f)$ Instead of using t_2 alone, some portion of $A_2(f)$ was machined by some other tool t_x that could access this region. This analysis leads to an elegant method to handle tool holder collisions.

In the example described above, consider a case where t_2 has tool holder collision in region r_1. Instead of using t_2 to machine the entire accessible area $A_2(f)$, t_2 is used to machine r_2 and t_x is used to machine r_1. Similarly, if t_1 was to be the first tool, instead of using t_2 to machine the entire region $D_{12}(f)$, t_2 would be used to machine r'_2 and t_x would be used to machine r'_1. By doing so, the independence of shape lemma is still valid, thus allowing the reduction in complexity of the problem as shown in sub-section “Graph algorithm”. However, the weights of the edges $e_{\{\mathrm{start}\to 2\}}$ and $e_{\{1\to 2\}}$ change as the regions associated with these edges are machined by multiple tools. The tool t_x is called the *surrogate tool*. The following sections will describe algorithms for finding accessible areas, and decomposed sub-pockets when there are tool holder collisions.

Tool nomenclature

In this research the tools that will be used for milling are general purpose end milling cutters. The tool holder is represented in terms of the bounding cylinders of the tool holder

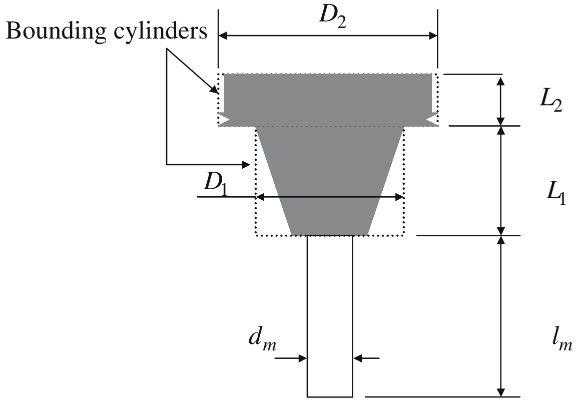

Fig. 5 Cutting tool nomenclature

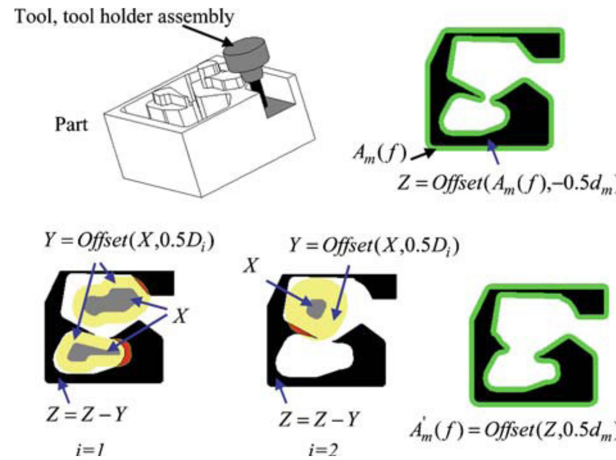

Fig. 6 Calculating collision free accessible area

geometry. Figure 5 shows a the schematic diagram of the tool, tool holder assembly with the associated dimensions

Finding tool holder collision free accessible area

Since, the parts considered here are 2.5D, the cross-section of the removal volume monotonically decreases with depth. The removal volume is divided into a number of pockets with precedence constraints. Precedence constraints occur because the pockets may be nested. We assume that no part of a pocket is machined until all the other pockets which have precedence are *completely machined*. For example, if the removal volume is divided into three pockets with precedence constraints given by $f_1 \rightarrow f_2 \rightarrow f_3$, no part of f_3 is machined before completely machining f_1, f_2.

Algorithm 3: *Finding Tool Holder Collision Free Accessible Area*

PROCEDURE COLLISION_FREE_ACCESSIBLE_AREA (f, I, t_m)
$A_m(f) \leftarrow$ ACCESSIBLE_AREA(f, I, t_m)
$Z \leftarrow Offset(A_m(f), -0.5d_m)$
$n \leftarrow$ number of tool holder cylinders
for $i = 1$ **to** $i < n$ **do**
  $D_i \leftarrow$ diameter of the ith tool holder cylinder
  $h_i \leftarrow$ distance of the bottom of the ith cylinder from the pocket bottom
  $X \leftarrow$ section of I at a distance h_i from the pocket bottom
  $Y \leftarrow Offset(X, 0.5D_i)$
  $Z \leftarrow Z - Y$
end for
$A'_m(f) \leftarrow Offset(Z, 0.5d_m)$

Consider a pocket $f(p, h)$ and the associated intermediate stock I. Since, the cross-section of the removal volume will monotonically decrease with depth, tool holder collision is most likely to occur when the tool is machining the bottom of the pocket. If the tool has no tool holder collision at this level, then it is safe to assume that there are no tool holder collisions anywhere along the depth of the pocket.

The first step is to generate the intermediate stock I. The intermediate stock is the shape of the part just after the pocket in consideration has been completely machined. The problem that we address in this section is to find the collision free accessible area of a tool within the pocket p. As mentioned before, this area is calculated when the tool tip is in contact with p. Algorithm 3 illustrates the procedure to calculate collision free accessible area. Figure. 6 shows an example.

If the nominal accessible area $A_m(f)$ is different from that of the collision free accessible area $A'_m(f)$, then tool holder collision exists for tool t_m. A surrogate tool has to be selected to “help” t_m to machine to the extent of its nominal accessible area.

Selecting surrogate tools

In order for any tool to be used as a surrogate tool, the region where tool holder collisions occur should be accessible to the surrogate tool. Also, the surrogate tool should not have tool holder collisions while machining that region.

The area that has tool holder collisions for tool t_m is given by:

$$Y = A_m(f) - A'_m(f). \tag{20}$$

In general, the area Y can be composed of many disconnected areas. In other words, $Y = y_1 \cup y_2 \ldots y_r$. Not all of these areas may be accessible by a single surrogate tool. A separate problem that has to be solved is to find an optimal subsequence of surrogate tools to machine Y. In this research we select a single surrogate tool to machine each of the regions y_r.

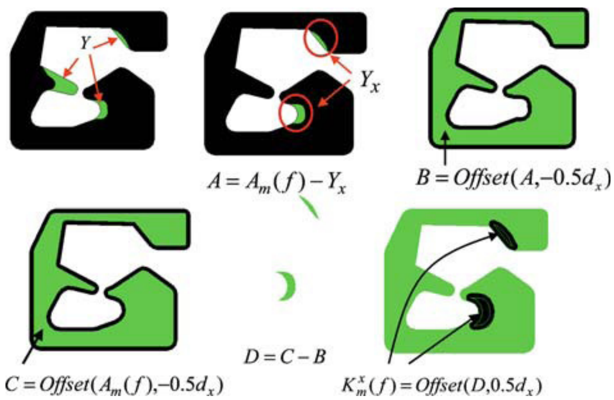

Fig. 7 Surrogate tool area calculation

A tool t_x can be a surrogate tool if and only if its collision free accessible area $A_x^{'}(f)$ covers the area y_r. In other words:

$$y_r \subset A_x^{'}(f). \tag{21}$$

The area y_r has sharp corners and open edges and will have to be extended for t_x to cover the open edges and sharp corners for complete machining. The following procedure is used to calculate the extended area. (Fig. 7)

Algorithm 4: *Calculating Surrogate Tool Machining Area*

PROCEDURE SURROGATE_TOOL_AREA($Y, A_m(f), t_x$)
Split Y into its constituent lumps $y_1, y_2 \dots y_r$
for i =0 **to** $i < r$
  if $(y_r - A_x(f) == NULL)$
  /* surrogate tool can machine the lump without tool holder collision*/
    $Y_x = Y_x \cup y_r$
  end if
end for
if $(Y_x \neq NULL)$
  $A = A_m(f) - Y_x$
  $B = Offset(A, -0.5d_x)$
  $C = Offset(A_m(f), -0.5d_x)$
  $D = C - B$
  $K_m^x = Offset(D, 0.5d_x)$
end if

Decomposed area in presence of tool holder collisions

Calculating the weights of the edges in the tool sequence graph requires the computation of decomposed area at each depth of cut. However, the decomposition in this case should take care of tool holder collisions.

Consider two tools $t_m, t_n : d_m > d_n$ and a pocket $f(p, h)$. Let $A_n^{'}(f)$ be the collision free accessible area of tool . Let $K_n^x(f)$ be region where surrogate tool t_x machines.

Suppose we use the larger tool t_m first and then use t_n for clean up, the decomposed area $D_{mn}^{'}(f)$ free of tool holder collisions that t_n machines is given by:

$$D_{mn}^{'}(f) = \text{DECOMPOSE\_AREA}\ (A_n^{'}(f), A_m(f), d_m)$$

The decomposed area that the surrogate tool machines is given by:

$$K_{mn}^{x}(f) = \text{DECOMPOSE\_AREA}\ (K_n^x(f), A_m(f), d_x)$$

Once the accessible areas for each tool and the decomposed areas for all possible tool pairs are calculated, tool paths are generated to calculate costs. The graph formulation developed earlier can now be used to find the optimal collision free tool sequence.

Results

The algorithms developed in this research have been implemented in a prototype system. The ACIS solid modeling geometric kernel was used for the Boolean and geometric operations. The tool set and the associated cutting parameters are as shown in Table 1. Table 2 shows the tool holder dimensions of each tool. We have assumed that each tool holder can be approximated using two bounding cylinders. However, it is possible to add more bounding cylinders to better approximate the tool holder. The number of possible tool sequences using this tool set is 255. In this exercise we account for total time only and not total cost.

Figure 8 shows the part and the removal volume for which tool sequence selection was performed using algorithms

Table 1 Tool database

Tool	d_m(in)	Woc (in)	Doc (in)	Feed (in/min)	Speed (rpm)
t_1	0.125	0.0625	0.06	15	15277
t_2	0.2	0.1	0.095	16	8200
t_3	0.25	0.125	0.12	18	7638
t_4	0.3125	0.15625	0.15	20	6111
t_5	0.375	0.1875	0.18	21	5092
t_6	0.4	0.2	0.195	21.5	4000
t_7	0.45	0.225	0.22	22.0	4210
t_8	0.5	0.25	0.24	23.0	3819

Table 2 Tool holder data

Tool	d_m (in)	l_m (in)	D_1 (in)	L_1 (in)	D_2 (in)	L_2 (in)
t_1	0.125	0.4	0.285	0.6	0.4	1.0
t_2	0.2	0.4	0.285	0.6	0.4	1.0
t_3	0.25	0.5	0.45	0.7	0.7	1.1
t_4	0.3125	0.6	0.45	0.8	0.7	1.2
t_5	0.375	0.5	0.6	0.7	0.8	1.15
t_6	0.4	0.6	0.8	0.8	0.8	1.2
t_7	0.45	0.5	0.75	0.7	1.0	1.1
t_8	0.5	0.5	0.75	0.7	1.0	1.1

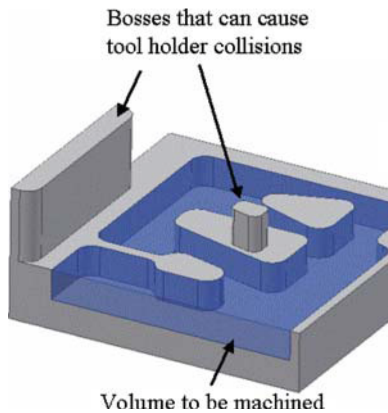

Fig. 8 Part and removal volume

Table 3 Optimal collision free tool sequence

Tool Dia(in)	Total Time (min)
0.5	0.992
0.2	2.713
0.125	0.683

developed in this research. The bosses can potentially cause tool holder collisions. Table 3 shows the resulting tool sequence with the associated times. We have assumed a 5 s tool change time for the automatic tool changer in the milling machine. Figure 9(a) shows the areas machined at each depth by the first tool t_8 (0.5″) in the tool sequence. Notice that there are regions where a surrogate tool has to be used. Figure 9(b) shows the swept volume of the tool holder if the tool were to be used to machine the nominal accessible area. Tool holder interference with the part exists in the region shown. Therefore, a surrogate tool, in this case $t_2(0.2'')$ was used to machine the region where tool holder collision occurs. Figure 9(c) shows the areas where $t_2(0.2'')$ machines at every depth of cut for the first clean up operation. Figure 9(d) shows the areas where t_1 (0.125″) machines at each depth of cut for the second and final clean up.

For comparison, we planned the same feature without the bosses in the part that cause tool holder collisions. Figure 10 shows the areas machined by each tool in the optimal tool sequence at each depth of cut. Table 4 shows the results of optimal tool sequence selection.

Conclusions

In this paper, we have presented a method to automatically handle tool holder collisions while selecting tool sequences for 2.5D machining. We have also developed geometric algorithms for calculating accessible areas and decomposed areas which take into account tool holder collisions. The concept of a *surrogate tool* has been introduced to handle tool holder collisions. Methods have also been developed to select appropriate surrogate tools.

The concepts developed in this research are also applicable to free-from machining. Typically, ball end mills are used to machine free-form surfaces. For a given surface, and any

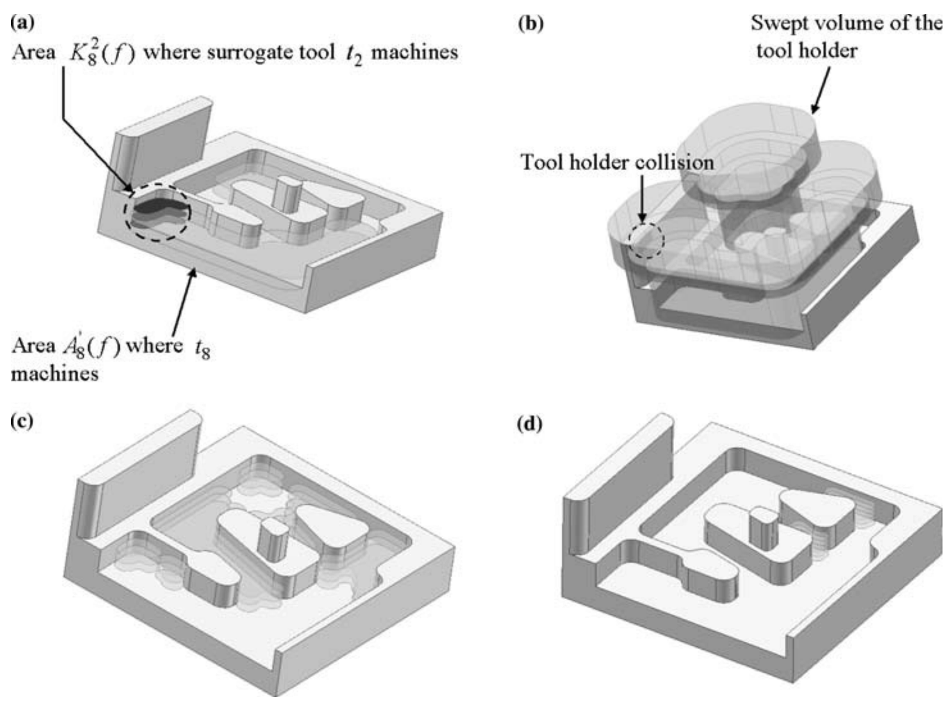

Fig. 9 Results of the optimal collision free tool sequence selection. **(a)** Collision free Accessible Area for 0.5″ dia tool and Area Machined by Surrogate Tool 0.2″ dia **(b)** Tool Holder Collision if 0.5″ tool Machined its Nominal Accessible Area **(c)** Decomposed Area for 0.2″ tool for the First Clean Up Operation **(d)** Decomposed Area for 0.125″ for the Second And Final Clean Up Operation

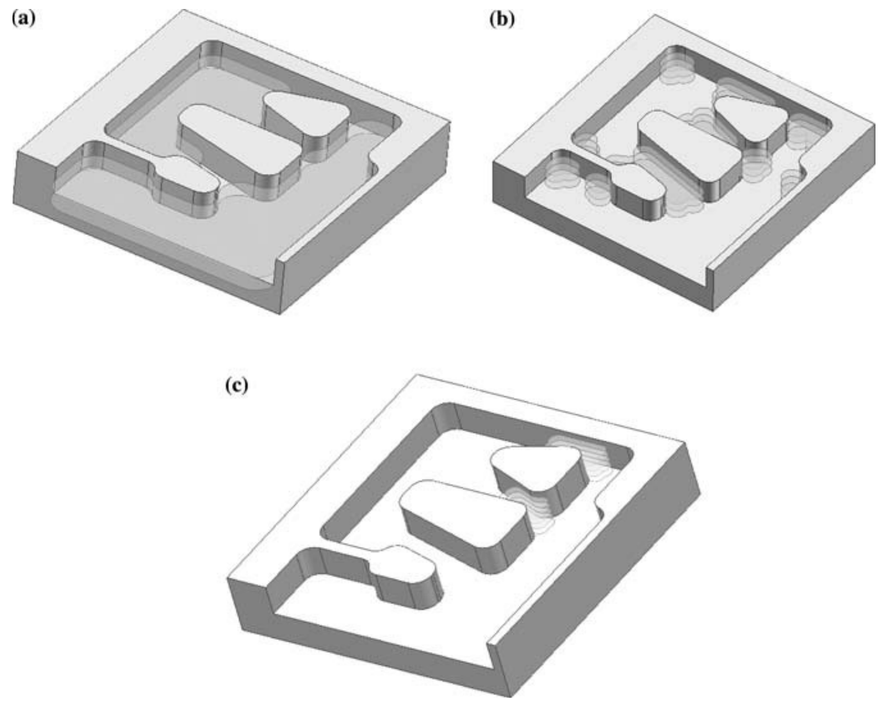

Fig. 10 Results of optimal tool sequence selection without considering tool holder collisions. (**a**) Area machined by 0.375″ Tool (**b**) Area machined by 0.2″ tool for first clean up operation (**c**) Area machined by 0.125″ tool for final clean up operation

two ball end cutters $t_m, t_n : d_m > d_n$, the accessible patch of t_m on the surface is a subset of the accessible patch of t_n. Therefore, the independence of shape assumption can be used to formulate the tool sequence selection problem as a graph problem. Consequently, tool holder collision handling that is described in the previous sections holds. However, the geometric algorithms needed for calculating the accessible patch and surface decomposition which are the free-form analogues of accessible area and decomposed area, respectively, are different.

Currently, we assume that every disconnected region in Y which has tool holder collisions will be machined by a single surrogate tool. However this may not be the optimal strategy. A better solution is to perhaps solve a tool sequence graph for a set of candidate surrogate tools to find the best sequence of surrogate tools to machine Y. For the graph solution to work, the accessible area of a smaller surrogate tool within Y should be a sub-set of the accessible area of a smaller tool (independence of shape lemma). This may not necessarily hold since the smaller tools may have tool holder collisions in some region within Y. Thus, there will be areas within Y that the larger tool can reach but not the smaller tool. The independence of shape assumption which is essential for the graph algorithm is therefore not valid.

Table 4 Optimal tool sequence without considering tool holder collisions

Tool Dia (in)	Total Time (min)
0.375	1.296
0.2	1.750
0.125	0.683

References

Arya, S., Cheng, S.W., & Mount, D.M. (2001). Approximate algorithm for multiple-tool milling. *International Journal of Computational Geometry and Applications, 11*(3), 339–372.

Bala, M., & Chang, T.C. (1991). Automatic cutter selection and cutter path generation for prismatic parts. *International Journal of Production Research, 29*(11), 2163–2176.

Chen, Y., Lee, Y.S., & Fang, S.C. (1998). Optimal cutter selection and machining plane determination for process planning. *Journal of Manufacturing Systems, 17*(5), 371–388.

Cormen, T., Leiserson, C., & Rivest, R. (1997). *Introduction to Algorithms*. New York: McGraw Hill.

D'Souza, R., Wright, P.K., & Séquin, C.H. (2001). Automated microplanning for 2.5D pocket machining. *Journal of Manufacturing Systems, 20*(4), 288–296.

Echave, J., & Shah, J.J. (1999). Automatic setup and fixture planning for 3-Axis milling. In *ASME Design Automation Conference on Las Vegas, NV*.

Gao, S., & Shah, J.J. (1998). Automatic recognition of interacting machining features based on minimal condition subgraph. *Computer Aided Design, 30*(9), 695–705.

Joo, J., & Cho, H. (1999). Efficient sculptured pocket machining using feature extraction and conversion. *Journal of Manufacturing Systems, 18*(2), 100–112.

Kannan, B., & Wright, P.K. (2001). Efficient algorithms for automated process planning for 2.5D machined parts considering fixturing constraints. *International Journal of Computer Integrated Manufacturing*, *17*(1), 16–28, 2004.

Kunwoo, L., Kim, T.J., & Hong, S.E. (1994). Generation of toolpath with selection og proper tools for rough cutting. *Computer Aided Design, 26*(11), 63–180.

Lim, T., Corney, J., Ritchie, J.M., & Clark, D.E.R. (2000). Optimizing automatic tool selection for 2.5D and 3D NC surface machining. *Computers in Industry, 26*(1), 41–59.

Lee, Y.S., & Chang, T.C. (1995). Application of computational geometry in optimization of 2.5D and 3D NC surface machining. *Computers in Industry, 26*(1), 41–59.

MASTERCAM (2002). http://www.mastercam.com.

Regli, W. (1995). *Geomteric Algorithms for Recognition of Features from Solid Models.* PhD Thesis, University of Maryland at College Park.

SURFCAM (2002). http://www.surfcam.com.

Veeramani, D., & Gau, Y.S. (1997). Selection of an optimal set of cutting tool sizes for 2.5D pocket machining. *Computer Aided Design, 29*(12), 869–877.

Yao, Z., Gupta, S.K., & Nau, D.S. (2001). A geometric algorithm for finding the largest milling cutter. *SME Journal of Manufacturing Processes, 3*(1), 1–16.