Systematic Approach to Contour-Parallel Tool Path Generation of 2.5-D Pocket with Islands

Manseung Seo[1], Haeryung Kim[2] and Masahiko Onosato[3]

[1]Tongmyong University of Information Technology, sms@tit.ac.kr
[2]Tongmyong University of Information Technology, hrkim@tit.ac.kr
[3]Hokkaido University, onosato@ssi.ist.hokudai.ac.jp

ABSTRACT

In this study, the systematic integration of a complicated tool path generation process is performed. As a key concept, the Offset Loop Entity (OLE) is devised to be the unique object of every procedure. Upon the OLE approach, the tool path generation proceeds hierarchically on classified levels being engaged by the devised process flow. Throughout the entire path generation process, the procedure is conserved upon various pocketing conditions. The OLE is shifting from an offset curve segment, to an offset loop, to multiple offset loops, then to tool trajectory loops. The OLE enables the proposed approach to possess three distinctive features: simplicity, applicability, and robustness. The prototype system is implemented and evaluated with actual pocket machining. The results verify that the devised method is robust enough to achieve the optimal tool path under any pocket configuration with no burden treating specific occasions.

Keywords: Pocket, Offset Curve, Offset Loop, Uncut Regions, Tool Path.

1. INTRODUCTION

Conventionally in pocket machining, the contour-parallel tool path is used most widely for large-scale material removals. Many researches on contour-parallel cutting have been performed [1-4]. Various algorithms for generating tool paths have been developed [5-9].

The goals of previous researches may be grouped into three categories; 1) to find proper offset curves, 2) to prevent uncut regions, and 3) to link tool path without tool-retraction. For the first, algorithms based on pair-wise intersection and Voronoi diagrams have been developed in the past two decades. One of them was utilized into the commercial CAM system, in spite of shortcomings such as time-consuming and numerical instability [1-3,6]. For the second, a specific adjustment on successive offset distance through the Voronoi diagram approach or a local care on tool trajectories through the pair-wise intersection approach has been undertaken [3,5,6]. For the last, several methods have been developed but those are not flexible enough to handle every kind of pocket configuration [3,8,9].

Especially for the third, it is still neither simple to plan an optimal path satisfying the Guyder's guidelines [10] nor easy to find a robust algorithm generating the path at any pocketing condition such as nested offset loops. The limitations on existing approaches are found in the literature. Bridges connecting a pocket and islands are inserted to consider those as a boundary [3]. The particular sub-path is merged into the corresponding path to treat specific occasions such as nested offset loops [9].

In this study, we aim to achieve all of the goals in the three categories simultaneously. To attain our aim, problems finding proper offset curves, preventing uncut regions, and linking tool path without tool-retraction are resolved sequentially. The contour-parallel tool path generation is totally integrated based on the devised concepts, levels, and process flow. The offset curve generation algorithm is systematically developed in conjunction with the work of Seo et al. [11,12].

Focusing on tool path linking, the path determination algorithm is devised on the continuation from uncut free offset loop construction. The proposed tool path determination algorithm is so simple that the determination of the linking sequence is accomplished by two searches; 1) the breadth first search on the offset loop generation history graph and 2) the depth first search on the tool path tree. Moreover, the algorithm is robust enough to generate the tool-retraction free tool path at any pocketing condition even with nested island loops and clean-up curves to remove uncut regions. Through this study, optimal tool path generation for uncut free pocketing with islands becomes feasible with no burden treating specific occasions.

For the verification of the proposed approach, the prototype system is implemented and examined with actual pocket machining. The implemented system and machined parts ensure that the devised method is robust enough to achieve uncut free, tool-retraction free, and slotting minimized paths for pocketing with islands.

2. DEFINITIONS OF CONCEPTS IN POCKETING

Unfortunately, the unification of terminologies in pocketing has not been completely accomplished. Thus, we redefine some concepts in order not to be confused. Then to integrate the complicated tool path generation process, we introduce some new concepts. We define the boundary of pocket as Contour curve Entity (CE), and the sequential linkage of the CEs as Contour Loop Entity (CLE). Imagining that a circle with a radius that equals to the offset distance is rolling on the CE, we define the trajectory of the center of the circle as Offset curve Entity (OE). Then, we define the sequential linkage of OEs as Offset Loop Entity (OLE). To determine the tool path for pocketing, the element linking all validated OLEs is needed. We define the linking element as Path link Entity (PE). Then, we define the entire tool path composed of all validated OLEs and all PEs as Tool Path Entity (TPE). Figure 1 shows the concepts defined in this work; bold solid line, thin solid line, and thin dotted line represent CLE, OLE, and PE respectively, and consequently all thin lines belong to the TPE.

In general, the OLE is often formed into an open loop having intersections. Open or intersecting OLEs are formed mainly due to various shapes of the CLE, i.e., pocket. Those result in undesirable cuts. Therefore, open or intersecting OLEs must be reconstructed into closed and non-intersecting ones by removal or insertion of OEs. In order not to be confused with the OLE reconstruction stage, we adopt the classification of OLE by Seo et al. [11]. The inborn OLE is the OLE created directly from the CLE. The crude OLE is the OLE without adjacent (local) self-intersection. The simple OLE is the OLE without self-intersection (local and global). The valid OLE is the OLE confirmed as the offset curve.

Furthermore, the concepts related to uncut regions are defined. Uncuts appear mainly on two occasions. The first is due to the improper selection of tool diameter for pocket boundary geometry. There is no way to avoid this kind of uncut, unless we change the tool. The second is due to the complexity of pocket geometry under the offset distance properly fixed for tool diameter and cutting time. It is found to be avoidable [6], and is still worthwhile to develop a better way of obviation. Uncuts should appear, when offset distance (d) is larger than tool radius (r) and smaller than tool diameter ($2r$). Those may be grouped into corner uncuts, neck uncuts, and center uncuts, as mentioned by Park et al. [6]. Usually, corner uncuts and neck uncuts exist in two successive offsetting, i.e., current and previous offsetting. These uncuts are shown as blacked out regions in Fig.2. By taking a glance at Fig.2, we easily notice that the uncut region exists if there is an intersection between curves created by a previous inner tool path and by a current outer tool path. Thus, we additionally define the trajectory made by tool radius as the Tool envelope Loop Entity (TLE). Then, we name the inward trajectory made by the previous tool path $[(n-1)^{th}]$ as previous TLE and the outward trajectory made by the current tool path $[(n)^{th}]$ as current TLE. By virtue of the devised OLE/TLE concept, it is possible to treat uncut region detection and clean-up curve generation problems as usual offsetting problems.

For the integration of entire tool path generation, defined concepts are classified hierarchically into tool path generation levels. The overview of the concepts and those relationships are shown in Fig. 3. At the local level, the OEs for all CEs are generated individually but in sequence. At the global level, valid OLE is generated by four procedures, *i.e.,* connection of OEs into a linked loop defined as OLE, detection of an intersection or interference between OLEs, decomposition or composition of OLEs at the intersection, and confirmation of OLE validity. At the planning level, the TPE is generated with OLEs linked by PEs, in consideration of technological constraints such as tool retraction or cutting strategy as pointed out by Guyder [10]. Therefore, OE, OLE, and TPE may be the pivotal concept of local, global, and planning levels respectively.

The systematic tool path generation scheme may be established by devising successive procedures in accordance with the defined concepts over hierarchically structured levels upon precise analysis considering constraints and requirements. Fig.4 shows the flow of the tool path generation process adopting the Structural Analysis and Design Technique (SADT). In comparison of Fig.4 with Fig.3, the tool path generation process is condensed into three vital procedures; offset curve generation procedure at local level, offset loop construction procedure at global level, and path determination procedure at planning level. Therefore, the relationship between levels and procedures is intuitively built. Eventually the definitions, the generation activities, and even the successiveness of every generation activity are clarified.

3. OFFSET LOOP GENERATION AT LOCAL AND GLOBAL LEVELS

In this study, we focused our attention more on tool path determination. Thus, the OE and OLE generation procedures for uncut free pocketing without/with islands are briefly discussed through illustrated examples, based on the defined concepts, in accordance with the devised tool path generation level and flow, and adopting the Offset-loop Dissection

Method (ODM) by Seo et al. [11,12]. The ODM and extended ODM were proposed based on the OLE concept that enables the algorithm to be implemented easily into the system at any condition, regardless of the number of offsets, the number of intersections, the number of islands, and even the number of uncut regions.

3.1 Generation of OE

To avoid complication, let us assume that all CEs are constructed with lines and circular arcs, even though some consider CE as a free-formed curve. Then conventional expression for OE becomes $C_i(t) + d \cdot N_i(t)$, where, $C_i(t)$ is the CE, d is the offset distance, and $N_i(t)$ is the unit normal vector with offsetting direction. Fig. 5(a) shows the pocket CLE consists of CEs and Fig. 5(b) shows the grouped OEs obtained from the CEs.

3.2 Construction of OLE

By connecting all OEs in the sequence of creation, the inborn OLE is constructed. In pocket machining, there is the strong possibility that the inborn OLE is formed into an open loop having local and global self-intersections as shown in Fig.5 (b). Since the open OLE results in undesirable cuts, it should be reconstructed into a closed one. As mentioned by Seo et al. [11], the self-intersection of OLE occurs at two grades; local self-intersection and global self-intersection. Therefore, the care on the self-intersection is distinguished into the local OLE reconstruction and global OLE reconstruction.

The local OLE reconstruction is performed sequentially through the OLE creation from OEs. It is done by inserting

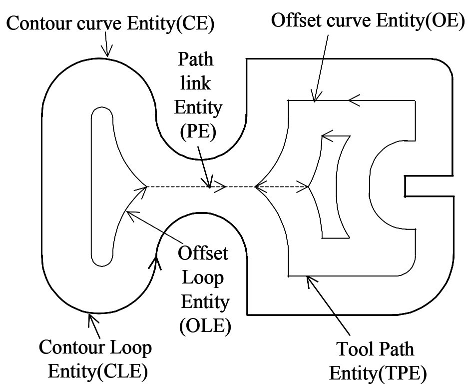

Fig. 1. Concepts for tool path generation

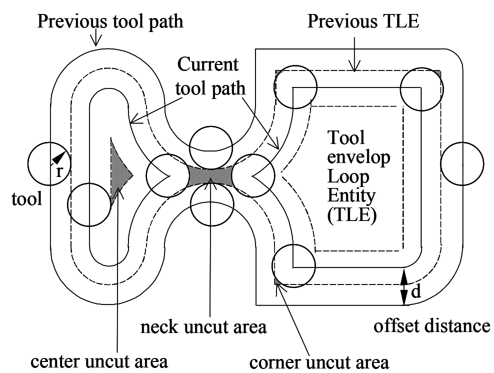

Fig. 2. Types and concepts related to uncut regions

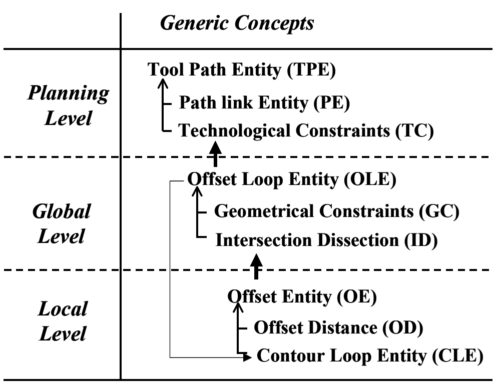

Fig. 3. Tool path generation process level

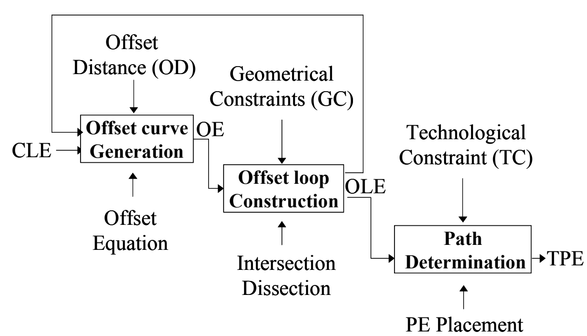

Fig. 4. Tool path generation process flow

additive OEs or removing local self-intersections between two adjacent OEs. The minor OLEs created by two end portions of OE are not closed. Those are discarded to remove local self-intersections. Through the local OLE reconstruction process, the inborn OLE is directly transformed into the crude OLE that is closed and has no intersection between adjacent OEs. Figure 5(c) shows the dissection of local self-intersection which creates one crude OLE (L_1) and four open OLEs (L_2, L_3, L_4, L_5) being discarded.

The global self-intersections are shown in Fig.5 (d). The crude OLE (L_1) is intersected by itself at three points. Whether it is local or global, the care in ODM on intersected OLEs is the only one, i.e., the dissection. Thus, the dissection procedure used in the local OLE reconstruction is applicable to the global OLE reconstruction, with no modification. However, in global OLE reconstruction, the detection of the intersection is needed before the dissection, and the reconnection of dissected OLEs is needed after. The global OLE reconstruction is completed by two consequent procedures; 1) detection of OLE intersection, 2) dissection of OLE at the intersection and then reconnection of dissected OLEs. The detailed description of global OLE reconstruction procedure can be found at Seo et al. [11].

Figure 5(e) shows the global OLE reconstruction. The crude OLE is transformed into the simple OLE without self-intersections. Detecting the intersection (q_a) and applying the dissection on crude OLE (L_1), the OLE (L_1) is decomposed into one simple OLE (L_1) and one crude OLE (L_6). By the second detection of intersection (q_b) and dissection of crude OLE (L_6), the OLE (L_6) is decomposed into one simple OLE (L_6) and one crude OLE (L_7). By the third detection of intersection (q_c) and dissection of crude OLE (L_7), the OLE (L_7) is decomposed into two simple OLEs (L_7 and L_8). Finally, no self-intersection exists, and all OLEs in Fig.5(e) are simple OLEs.

In ODM, the dissection performed right after the detection often causes unintentional self-acting dissections of OLEs. Therefore, the ODM might be even better.

3.3 Validation of OLE

The simple OLE obtained by the global OLE reconstruction may still not be an appropriate OLE as an offset curve for pocket machining. The characteristics of OLE, i.e., closeness and orientation, need to be examined to confirm the validity of OLE. In pocket machining, the closeness of OLE reflects the possibility of a continuous tool path, thus a valid OLE must be completely closed. The orientation of OLE determines cutting directions, inwards or outwards of the pocket. Fixing the orientation of CLE to be Counterclockwise (CCW), if the orientation of OLE is CCW then cutting is done on the inner side of the pocket, and if it is Clockwise (CW) then cutting is done on the outer side of the pocket. Thus the orientation of valid OLE must be CCW.

In Fig.5(e), two OLEs (L_6, L_8) are discarded since those are CW. Only two OLEs (L_1, L_7) are selected as valid OLEs, since those are completely closed and CCW. Then, the valid OLEs in Fig.5(f) are kept to play the role of an offset curve for pocketing and the role of CLEs in next offsetting turn.

3.4 Generation of offset curves for a pocket with islands by ODM

One of the salient features of the ODM is the applicability. The offset curve generation method for one OLE works as

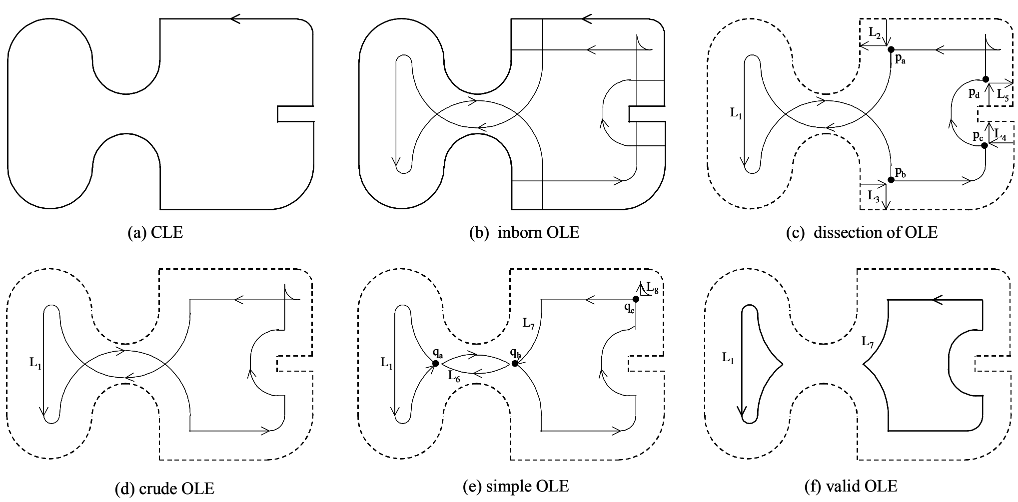

Fig. 5. Offset curve generation procedure for pocket without island

the method for multiple OLEs. In other words, the method for a pocket without islands works also for a pocket with islands, although the latter is complicated and even troublesome. This is anticipated since the ODM method is based on the OLE concept. The OLE concept enables the method to flexibly deal with the intersections at any situation and to integrate the complicated offset curve generation process. To ensure the merits, the ODM is applied to the generation of an offset curve for a pocket with islands, by shifting the object of intersection detection, dissection, and validation, from one OLE to multiple OLEs.

Using an illustrated example of the offset curve generation process for a pocket with an island, the ODM is evaluated. Figure 6(a) shows the CLEs of one pocket and one island in dotted line, and two simple pocket OLE (L_1, L_7) and one simple island OLE (L_9) in solid lines. As shown in Fig.6(b), one pocket OLE(L_7) and one island OLE (L_9) are dissected at the intersection (q_3). Dissected OLEs are reconnected into one combined OLE (L_7') conserving orientations but vice versa as shown in Fig.6(c). Applying dissection one more time at the intersection (q_4 in Fig.6(d)) and reconnecting again, one combined OLE (L_7') is decomposed into two combined OLEs (L_7", L_{10}) as shown in Fig.6(e). Performing OLE validation with the rule that the characteristic of the pocket OLE is transferred to the composed OLE when a pocket OLE and an island OLE are composed into an OLE, one CW oriented simple OLE (L_{10}) is discarded and two valid OLEs (L_1, L_7") are kept to play the role of offset curves for pocketing and the role of CLEs in next offsetting turn as shown in Fig.6(f). So, the ODM works for a pocket with islands.

In ODM, the complicated offset curve generation process is simplified into three consequent procedures regardless of islands; 1) Detection of OLE relationship, 2) Dissection of OLE, and 3) Validation of OLE. The ODM algorithm for generating an offset curve for a pocket with islands can be found at Seo et al. [11].

3.5 Detection of uncut regions by extended ODM

In this subsection, uncut region detection and clean-up curve generation procedures are briefly discussed based on the defined concepts, in accordance with the extended ODM by Seo et al. [12]. Consideration on uncuts due to the improper selection of tool diameter for the pocket boundary is excluded in this study. Emphasizing the fact that the uncut regions exist if there is an intersection between curves created by a previous inner tool path and by a current outer tool path, the ODM algorithm was extended to detect uncut regions and to generate clean-up curves. This was possible since the OLE/TLE concepts enable the ODM algorithm to be easily applied to uncut region detection by shifting the object of ODM from OLEs to TLEs.

To verify the extended ODM, the entire process of uncut region detection and clean-up curve generation is evaluated through an illustrated example shown in Fig.7. Figure 7(a) shows the previous [(n-1)th] tool path, the current [(n)th] tool path, the inward trajectory made by the previous tool path (previous TLE, L_0), and the outward trajectory made by the current tool path (current TLE, L_1). By taking a glance at Fig.7(a), we easily notice that the uncut region exists if there is an intersection between previous TLE(L_o) and current TLE(L_1). Moreover, by imaging that the previous tool path to

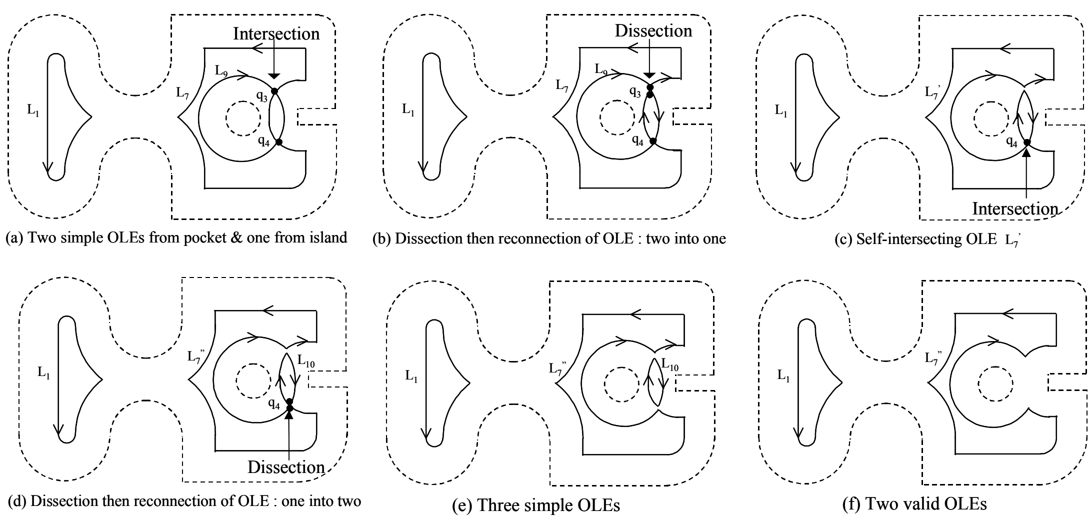

Fig. 6. Offset curve generation procedure for pocket with an island

be like a pocket CLE and the current tool path to be like an island CLE, the previous TLE(L_o) may be considered as a pocket OLE and current TLE(L_1) may be considered as an island OLE, and then, we could see that those exactly match as shown in Fig.7(b). Therefore, we just need to carry out the ODM procedure to detect the uncut regions upon OLE/TLE concepts. After the previous/current TLEs construction, the TLE reconstruction is processed as we did in the offset curve generation of the pocket with one island in Fig.6. Then, non-intersecting simple TLEs are obtained as shown in Fig.7(c). Performing TLE validation with the rule that the characteristic of the previous TLE is transferred to the composed TLE when a previous TLE and a current TLE are composed into a TLE, four CW oriented simple TLEs are discarded. Finally, four valid TLEs (L_c, L_d, L_e, L_f) with CCW orientation are kept to play the role of the clean-up curve. The clean-up curves are then appended to current valid OLEs taking the shortest line segment for the construction of uncut free tool path, as shown in Fig. 7(d).
Here, we may conclude that ODM scheme is flexible and robust enough to generate offset curve for pocket machining with island, even for uncut-free pocket machining with islands.

4. TOOL PATH DETERMINATION AT PLANNING LEVEL

In this section, the tool path is determined at the planning level, on the continuation from the offset loop generation process at the global level discussed in the previous section. The tool path for efficient pocket machining is known as the tool path with minimal slotting, no over-milling, and no tool retraction [8-10]. In order to find the most efficient tool path, the path determination algorithm is devised based on the OLE/PE concepts, pocket geometry, and accumulated data structure throughout offsetting. Our method is not only simple as being consisted of five steps but also excellent in minimizing slotting/over-milling and preventing tool-retraction under any kind of pocket geometry and configuration. Figure 8 shows an illustrated example of the tool path determination procedure.
As the first step, the OLE Generation History (OGH) graph is constructed by connecting valid OLEs during the OLE generation procedures mentioned in the previous section. By means of the OGH graph, the OLE is stored in a linked list data structure such that the lineage of OLE is expressed as tree data structure. The information about the relationships among OLEs through the successive offsetting is contained in the OGH graph. Figure 8(a) shows the OLE map generated by ODM algorithm for a pocket with two islands, and Fig. 8(b) shows the OGH graph corresponding to the OLE map. In the OGH graph, the OLE itself is represented as a node and the birth of OLE is expressed using branches like a family tree. For this specific example, 13 valid OLEs are connected with 15 branches in the sequence of generation throughout offsetting. However, anyone looking at the graph could find that the minimal number of branches needed to connect 13 OLEs is 12. Over-milling is unavoidable if all 15 branches in Fig. 8(b) are selected as the linking tool path segments for 13 OLEs. Thus, the modification of the OGH graph is inevitable to discard the redundant branches responsible for over-milling.
As the second step, the OLE where the cutting starts is selected by considering technological constraints, such as tool retraction and/or cutting strategy. From the selected starting OLE, the Breadth First Search (BFS) on the undirected OGH graph is performed [13]. Then, the Tool Path (TP) tree is constructed upon the BFS result by linking all OLEs with the minimal number of PEs instead of excessive branches. The relational information about the OLEs/PEs with linking sequence and the geometric information on the OLEs are contained in the TP tree. Figure 8(c) shows the TP tree for this specific example. The OLE_{13} is selected for outward cutting to minimize slotting. The result shows that the reduction on the number of over-milling is accomplished by the BFS search. By linking all OLEs with the minimal

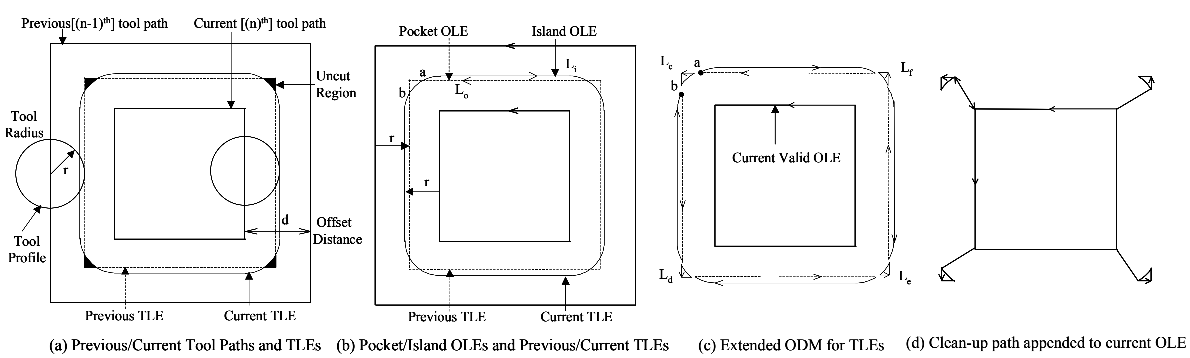

Fig. 7. Uncut region detection procedure and appended clean-up path

number of PEs in Fig. 8(c), three redundant branches (b_{10}, b_{11}, b_{13}) from Fig. 8(b) are discarded. Although the TP tree contains the additional information on PEs, it is still not possible to start an actual cutting relying only upon the information contained in the TP tree. Therefore, the geometric information on the PEs such as its position and entry/exit point must be determined.

As the third step, an ending OLE is selected by considering technological constraints and by accounting the number of PEs to the starting OLE. The one-way path from the starting OLE to the ending OLE is set. The PEs in the one-way path are placed onto the OLE map with the location of an entry/exit point on the OLE. The OLEs/PEs in the one-way path are marked on the TP tree. Forming the shortest line segment upon upper PE position, the rest of PEs in breadth of the marked path are placed onto the OLE map in the breadth order with the location of an entry/exit point on the OLE. As the result of the PE placement on the OLE map, the parental OLE having two and more successive OLEs in TP tree appears with multiple entry/exit points on it. The consideration on the visiting sequence of multiple entry/exit points existing on a single OLE is left for the next step. In this specific example, OLE_1 is selected as an ending OLE. The PEs (PE_{15}, PE_8, PE_4, PE_1) in the one-way path are placed onto the OLE map as shown in Fig. 8(d). The OLEs (OLE_{13}, OLE_{10}, OLE_6, OLE_4, OLE_1) and the PEs in the one-way path are marked on the TP tree to be searched last. Then, the rest of PEs in breadth of the marked path are placed onto the OLE map with the location of an entry/exit point as shown in Fig. 8(d). The possibility of over-milling is not negligible, since the parental OLEs having two and more successive OLEs showed up with multiple entry/exit points on them shown as OLE_{10}, OLE_6, OLE_5 in Fig. 8(c) and (d). Therefore, the visiting sequence of multiple entry/exit points should be ordered properly.

As the forth step, the OLEs/PEs in the marked path are placed on the rightmost of the TP tree to be searched last. The OLEs/PEs in breadth of the marked path are reordered comparing the travel distance upon parental OLE orientation for visiting sequence. In Fig.8(e), the bold lined marked path $OLE_{13} \rightarrow OLE_{10} \rightarrow OLE_6 \rightarrow OLE_4 \rightarrow OLE_1$ is shown on the rightmost side being planned to be searched last. The breadth order $OLE_4 \rightarrow OLE_8 \rightarrow OLE_9 \rightarrow OLE_{11}$ is changed to $OLE_{11} \rightarrow OLE_8 \rightarrow OLE_9 \rightarrow OLE_4$, to prevent over milling considering the CCW orientation of parental OLE (OLE_6) as shown in Fig. 8(c), (d) and (e).

As the last step, the Depth First Search (DFS) on the arranged tool path tree is performed [13]. To avoid tool-retractions, the marked one-way path is searched last. The shortest path upon DFS result is written down and saved as TPE with minimal slotting, no over-milling, and no tool-retraction. For this specific example, the relational information on the OLEs/PEs in the shortest tool path obtained by DFS is shown in Fig. 8(f), along with their geometric information engraved on the OLE map shown in Fig. 8(e). The result verifies the simplicity and excellence of the devised method in treating nested offset loops. The proposed tool path determination algorithm may be summarized as:

- Step 1: Generation of OGH graph from OLE map.
  - 1-1. Connect valid OLEs in the sequence of generation throughout offsetting.
- Step 2: Construction of TP tree.
  - 2-1. Perform BFS on undirected OGH graph starting from a selected OLE.
  - 2-2. Link OLEs by inserting PEs upon BFS result.
- Step 3: Placement of PEs onto OLE map.
  - 3-1. Select an ending OLE considering the number of needed PEs and the constraints.
  - 3-2. Locate PEs in one-way path from the starting OLE to ending OLE and mark the path.
  - 3-3. Locate PEs in breadth accordant with upper PE position forming the shortest line segment.
- Step 4: Arrangement onto TP tree.
  - 4-1. Allocate the marked path on the rightmost to be searched last.
  - 4-2. Reorder OLEs in breadth of the marked path upon the parental OLE orientation.
- Step 5: Determination of the shortest path.
  - 5-1. Perform DFS through every OLE down to the ending OLE on arranged TP tree.
  - 5-2. Confirm the shortest path upon DFS result and save it as TPE.

The proposed tool Path Determination Method (PDM) generates a tool path free from over-milling and tool-retraction, even with minimal slotting. Moreover, the PDM algorithm is robust enough to generate a tool path under any kind of offset loop configuration including a nested one for uncut free pocketing with islands.

5. IMPLEMENTATION

To verify the proposed tool path generation approach, the prototype system is implemented upon the ODM algorithm [11], the extended ODM algorithm [12], and the PDM algorithm, using C language and Open GL graphic library. Then,

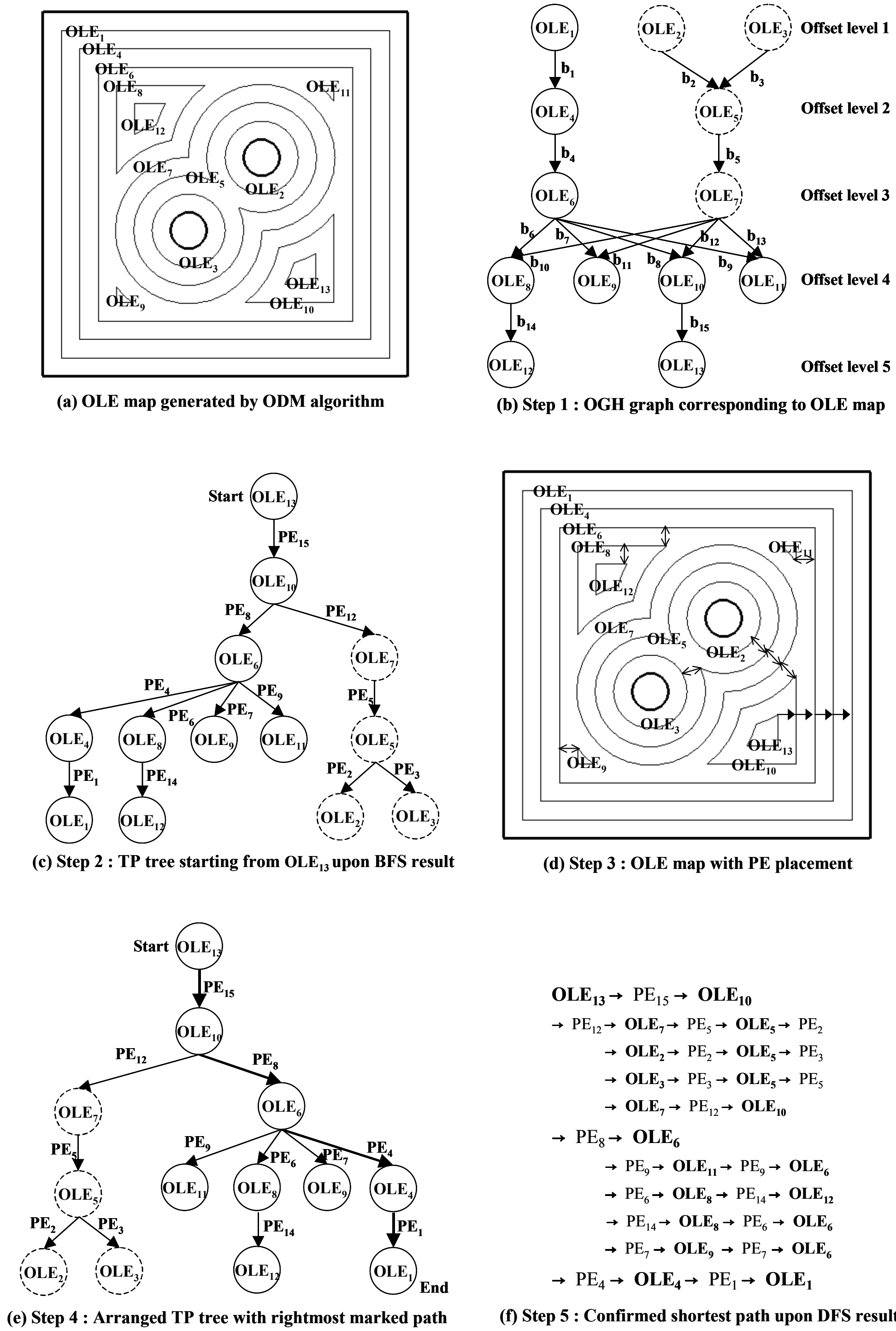

Fig. 8. Tool path with minimized slotting/over-milling and without tool-retractions

the implemented pocketing system with multiple islands and even for uncut free pockets is evaluated on IBM RS6000 140/43P, and examined with actual pocket machining.
The screen images of the tool path obtained from the implemented system are shown in Fig.9. The photograph of actually machined wax parts with the tool path created by the PDM algorithm on the mini-milling machine [Modella 50] is shown in Fig.10. The results ensure that the proposed approach is robust enough to generate an uncut free, over-milling free and tool-retraction free tool path for any kind of pocket.

6. CONCLUSION

In this study, we proposed the systematic tool path generation approach. The integration of the tool path generation process for uncut free and tool-retraction free pocketing is carried based on the devised concepts, levels, and process flow. Throughout the entire tool path generation process, the object of every procedure is devised to be the unique one,

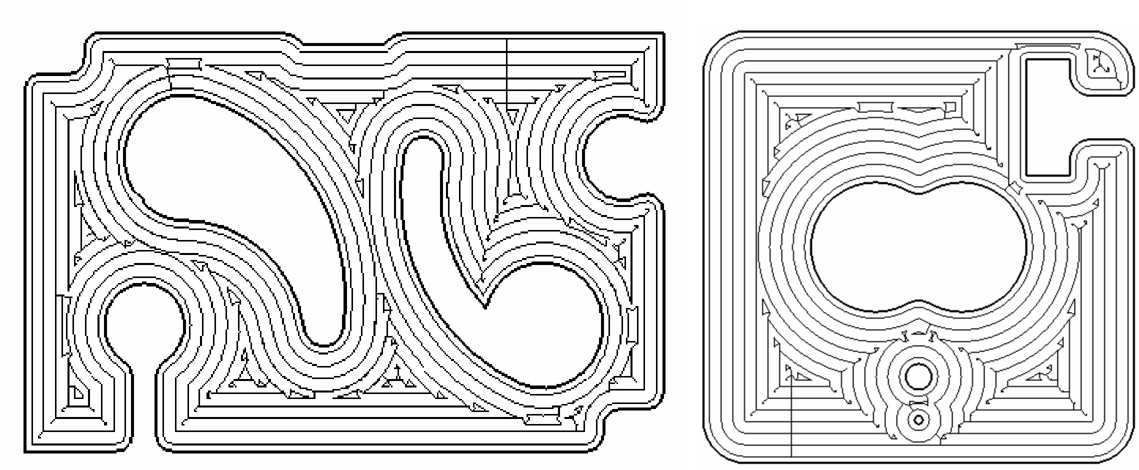

Fig. 9. Uncut free tool path for pocket with islands

Fig. 10. Photograph of machined parts corresponding to Fig.9.

i.e., the OLE. Thus, the systematic integration of the tool path generation becomes feasible and the algorithm is easily implemented into the system under any pocketing condition. The prominent features of the proposed approach are:
(1) Entire offsetting procedure is systematically integrated using the OLE. Thus, only three consequent procedures are needed for offset curve generation, namely, intersection detection, OLE dissection, and OLE validation.
(2) Every offsetting procedure deals only with the OLE, which enables the algorithm to be easily implemented into the system at any condition regardless of the numbers of offsets, the numbers of intersections, the numbers of islands, and even the numbers of uncut regions.
(3) Each offsetting procedure is designed based on the OLE. Regardless of the treatment level, the procedure itself is conserved upon various pocketing conditions; the object of the procedure is shifting from the OE, to the OLE, to the OLEs, then to the TLEs.
(4) Entire tool path determination procedure is systematically integrated on the continuation from the offsetting procedure.
(5) Every tool path determination step deals only with the OLE, which enables the algorithm to be easily implemented into the system under any condition regardless of nested loops and clean-up curves.
(6) Each tool path determination step is devised based on the geometry and the data structure. Therefore, the algorithm is so simple being accomplished in two searches, but robust enough to generate the slotting free, over-milling free, and tool-retraction free tool path under any pocketing condition.
The implemented system and actually machined parts verify that the proposed tool path generation approach based on the OLE concept is the proper and reliable. Furthermore, the PDM algorithm is robust to fulfill the optimal tool path generation, under any pocket configuration, upon any technological requirement, and with no burden treating specific occasions.

7. ACKNOWLEDGEMENTS

This work was supported by the Tongmyong University of Information Technology under the Research Abroad Grant.

8. REFERENCES

[1] Suh, Y.S. and K. Lee, K., NC Milling tool path generation for arbitrary pockets defined by sculptured surfaces, *Computer Aided Design*, Vol.26, No.5, 1990, pp.273-284.

[2] Hansen, A. and Arrab, F., An algorithm for generating NC tool paths for arbitrarily shaped pockets with islands, *ACM Transactions on Graphics*, Vol.11, No.2, 1992, pp.152-182.

[3] Held, M., Lukacs, G. and Andor, L., Pocket machining base on contour-parallel tool paths generation by means of proximity maps, Computer Aided Design, Vol.26, No.3, 1994, pp.189-203.

[4] Hatna, A., Grieve, R. J. and Broomhead, P., Automatic CNC milling of pockets: geometric and technological issues, *Computer Integrated Manufacturing Systems*, Vol.11, No.4, 1998, pp.309-330.

[5] Shan, Y., Wang, S. and Tong, S., Uneven offset of NC tool path generation for free-form pocket machining, *Computers in Industry*, Vol.43, 2000, pp.97-103.

[6] Park, S. and Choi, B., Uncut free pocketing tool-paths generation using pair-wise offset algorithm, *Computer Aided Design*, Vol. 33, No.10, 2001, pp.739-746.

[7] Castellino, K., D'Souza, R. and Wright, P. K., Toolpath optimization for minimizing airtime during machining, *Journal of Manufacturing Systems*, Vol.22, No.3, 2002, pp.173-180.

[8] Park, S., and Chung, Y., Offset tool-path linking for pocket machining, *Computer Aided Design*, Vol.34, No.4, 2002, pp.299-308.

[9] Park, S., Chung, Y. and Choi, B., Contour-parallel offset machining without tool-retractions, *Computer Aided Design*, Vol.35, No.1, 2003, pp.35-45.

[10] Guyder, M., Automating the optimization of 2.5 axis milling, *Computer Industry*, Vol.15, 1990, pp.163-168.

[11] Seo, M., Kim, H. and Song, J., Generation of offset curve for pocket with islands based on offsetloop dissection method, *The proceedings of the 30th Annual Conference of the IEEE Industrial Electronics Society,* 2004, 0-7803-8730-9.

[12] Seo, M., Kim, H. and Onosato, M., Detection of uncut regions in pocket machining using loop dissection, *The proceedings of the 5th International Conference on Machine Automation,* 2004, pp.131-136.

[13] Horowitz, E., Sahni, S. and Anderson-Freed, S., *Fundamentals of data structures in C*, Computer Science PRESS, W. H. Freeman and Company, 1993.