A Novel Cutter Path Planning Approach to High Speed Machining

Zhiyang Yao

Chinese University of Hong Kong, zyyao@acae.cuhk.edu.hk

ABSTRACT

High speed machining is playing a more and more important role in modern machines shops, in particularly, in aerospace machining, mold and die machining. In high speed milling operations, a path that has the least number of sharp turns is preferred. Spiral curve is a special curve which can have no sharp turn, and the step-over between adjacent segments can also be controlled to a predefined value. However, traditional approaches to finding such a path either work for very simple convex regions, or, for irregular shapes, many overlap linkage paths have to be used. Thus, although spiral curves are promising, there is still not a good algorithm in generating a path which 1) covers the entire region, 2) has least number of turns, 3) uses a controllable step-over value. In this paper, we will introduce a novel approach which covers a 2D region in a very efficient manner. In particular, 1) it covers the 2D region using a set of modified spiral curves based on the geometry of the 2D region, 2) these curves are linked in a proper manner, 3) step-over values can be predefined and maintained.

Keywords: Path Planning; High Speed Machining; Spiral Curve.

1. INTRODUCTION

High Speed Machining (HSM is short) is the use of higher spindle speeds and feed rates to remove material faster as swell as maintain the quality of part finishing [1] . To achieve high speed during milling operations, the cutter should have a constant chip load and usually with very a small step-over at very high feed rates. Moreover, the path should avoid sharp turns. Otherwise, unexpected tool breakage that results from exceeding a tool's permissible loading conditions not only costs money, but also disrupts the machining process [2].

In this paper, we consider the problem of using a standard 2D shape called *mover* (in this paper, we will only focus on a cylindrical shape to simulate a flat-end milling cutter) to cover a larger 2D region called *pocket* that is bounded by a set of line or arc segments with a set of obstacles that cannot be intersected.

In the 2D path planning problem, usually a path consists of a set of line or arc segments, and along the path, there are three types of mover movements as shown in Fig. 1: (1) mover moves along the path to cover region that hasn't been fully covered (it is called effective move); (2) mover moves in the region that has been fully covered before (it is called repositioning move); (3) in milling operations, the cutter can be lift up (i.e., retracted) to a clearance height and then moved to another position using rapid movement and then lowered down to perform another cutting motion (it is called retraction). Both the repositioning move and the retraction are used to link disconnected effective moves together. For example, if there are two connected concave corners, by using traditional zigzag strategy, the mover can only cover one corner by a continuous path. In order to cover the other corner, it has to perform either a retraction or a repositioning move. However, in HSM, retraction moves are not preferred because the cutter has to be lift up and then moved down; therefore, retraction moves will not be considered in the path planning for HSM unless repositioning moves are not adequate, for example, if there are disconnected regions. Repositioning moves result in path segments that increase the total covering time. Therefore, in order to get an efficient path, such move should be eliminated or minimized.

Even for the effective moves, when the mover moves along this path, if most of the sweeping area has been covered before, this move is still not efficient. The ideal way of performing efficient coverage is to move the movers along a path to cover area that has not been touched before. In this case, the so called step-over is 0%. Obviously in milling operations, the step-over usually cannot be 0% because there are other factors like cutting force, surface finish etc. to be considered in order to determine a proper step-over. However, a constant step-over value is preferred especially in milling operations because it will result in constant cutting force. Thus, a path with constant step-over is preferred.

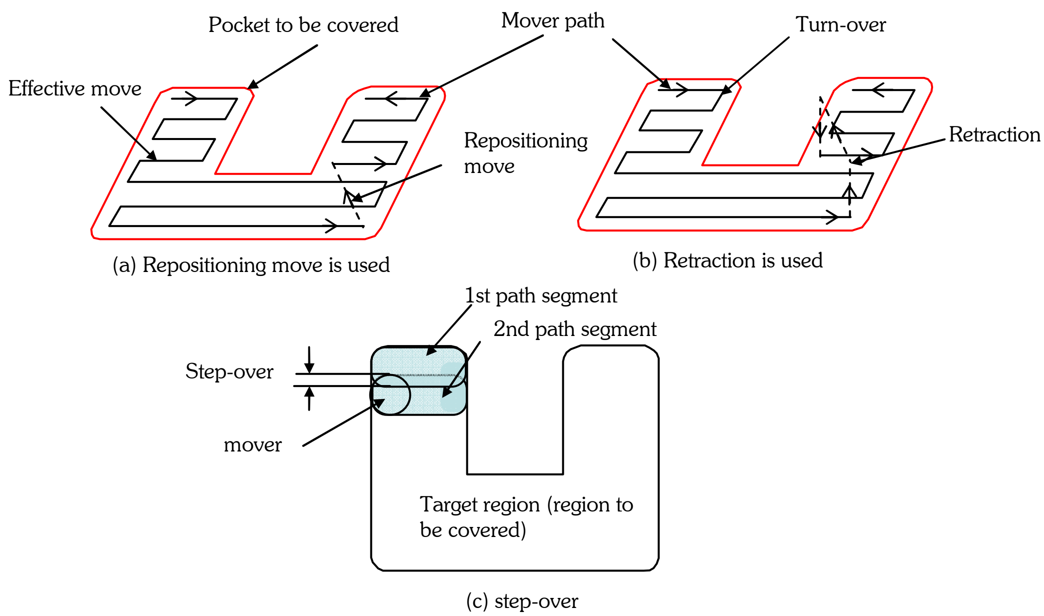


Fig. 1. Examples of mover path, turn-over, repositioning, and retraction and effective moves and step-over between two paths.

Moreover, as mentioned before, if the mover moves along a path with shape turns, for example, the so called turn-over in milling operations, the speed of the mover has to be slowed down and then the mover will take a turn and then resume its movement. This kind of shape turns has particular impact on HSM, because usually to make a turn takes much longer time than the normal movement, and the resulting sudden cutting load change may cause tool breakage. Therefore, a path with minimal number of sharp-turns is desirable.

Thus, the ideal path should be a path with minimal number of repositioning moves, minimal number of sharp turns, constant step-over value, and with most efficient effective moves. This type of path is desirable in milling operations, especially in high speed milling operations.

Traditional approaches in finding path for 2D regions are normally either using zigzag path, or using contour-parallel paths. These paths cannot meet the aforementioned requirements. Thus, we will consider another approach in finding such a path, i.e., using spiral curves.

Spiral curve is a special curve which can have no sharp turn, and the step-over between adjacent segments can also be controlled to a predefined value. However, traditional approaches in finding such a path either work for very simple convex regions, or, for irregular shapes, many overlap linkage paths have to be used. Thus, although spiral curves are promising, there is still not a good algorithm in generating a path which 1) covers the entire region, 2) has least number of turns, 3) uses a controllable step-over value. In this paper, we will introduce a novel approach which covers the entire 2D region in a very efficient manner. In particular, 1) it covers the 2D region using a set of modified spiral curves based on the geometry of the 2D region, 2) these curves are linked in a proper manner, 3) step-over values can be predefined and maintained. This proposed spiral path planning algorithm can be used in generating efficient milling cutter paths for HSM in which the cutter engagement value is controlled within a predefined value.

In this paper, Section 2 reviews the traditional approaches in generating a path, and concludes that spiral based paths are desirable in some applications. Following that, section 3 briefly introduces related background knowledge, i.e., spiral curves and Medial Axis transform, which will be used in Section 4 when introducing the novel algorithm to generating such a path. Section 5 presents the discussion and implementation of such an algorithm. We believe this algorithm can be used to generate better cutter paths than traditional cutter path planning algorithms in most of the cases.

2. RELATED WORK

In 2D path planning problems, there are roughly two main research directions [3]. The first one involves strategies based on predefined curves and the other direction involves emergent strategies based on grid traversing. Meanwhile, there are some special strategies particularly used to handle some specific requirements of HSM.

2.1 Strategies Based on Predefined Curves

Strategies based on predefined curves use a series of predefined curves to cover a region. These predefined curve segments are called base curves. Based on the differences of these base curves, there are several strategies:
The zigzag strategy uses a series of parallel open curve segments as the base curve to cover the pocket, and then these segments are linked together.

The contour parallel method uses a series of offset curves to the boundary to cover the pocket. Most of the published algorithms use Voronoi diagram to calculate the offset curves. The computation task is intense for a complex pocket.

Similar to the contour parallel method, another method uses a series of window shapes to cover the pockets. These window shapes are either rectangular or circular. The calculation is easy, however, for an irregular shape, it is hard to cover with the full windows.

The curvilinear tool-path method is developed in Boeing Company [4]. This method uses the solution of an elliptical partial differential equation boundary value problem defined on a pocket region, then morph a smooth low-curvature spiral path in a pocket interior to one that conforms to the pocket boundary. However, the path generated by this method works fine only for some simple convex pockets.

Space-filling curve is defined as a continuous mapping of a unit line segment onto the unit square. Several types of space-filling curves exist based on the types of initiators and generators used to create the space-filling curve with any resolution. Space-filling curve based path also does not need additional linkage cutter path segment, however, it will result in so many sharp corners that the efficiency is low and the surface finish is poor [5], and therefore, it is not commonly used.

Spiral curve based trajectory layout does not need additional linkage segments within one spiral. Therefore, in some cases it will result in efficient paths. Moreover, this method allows the mover to always cover some portion of un-covered region, which will lead to approximately constant step-over value. However, there is no good algorithm in generating such a path for 2D regions, and we are going to explore along this direction to find a better path using spiral curves.

2.2 Emergent Strategies Based on Grid Traversing

Emergent strategies based on grid traversing were developed in the computational geometry community. These strategies first represent pocket by grids, and then an optimal or near optimal tour that travels all grids with minimal cost (for example, the length of the tour) is found mostly by approximation algorithms. The rationale behind these strategies is as following.

Given a mover and a pocket, finding the path geometrically can be cast as a problem that finds a path or a tour whose Minkowski sum is the given pocket region. Therefore, if one can search all possible movements of the mover, there should be an optimal solution. However, if the mover can move along any direction as long as it does not touch the boundary, the search space is too large to be solved. In computational geometry community, several efforts have been made to find an optimal tour by restricting the mover movement directions. In this approach, the pocket is first decomposed into grids (the size of the grid is determined by the mover size). The mover is considered to have a unit square shape and can move only vertically or horizontally. After that, an optimal tour that covers the entire grid, subject to given objective functions, can be found.

The grid-traversal problem has been shown to be NP hard [6], therefore, several approaches have been proposed using approximation algorithms. For example, in [7], an approximation algorithm is given with a 3.75-approximation for minimum-turn axis-parallel tours for a unit square cutter that covers an integral orthogonal polygon (with holes).

In summary, in order to achieve the goal of finding a path with minimal number of turn-overs, minimal number of repositioning movements, and with constant step-over values, spiral curve based approach is preferred.

2.3 Special Strategies for HSM

Beside those aforementioned strategies, some special strategies are adapted in HSM to handle cases where special treatments are required. Following are some examples.

In HSM, cutter paths should be able to handle cases where internal sharp corners are needed. One possible way is to round the sharp motion out of the tool path. Usually, simple “looping” paths are used if there are sharper turns, for instance, linkage between two zigzag parallel cutter paths. Alternatively, a “golf club” step-over between passes can also be used in case the cutting speed is very high. However, this kind of movements is usually non-effective, in other words, they don’t really performing cutting [1].

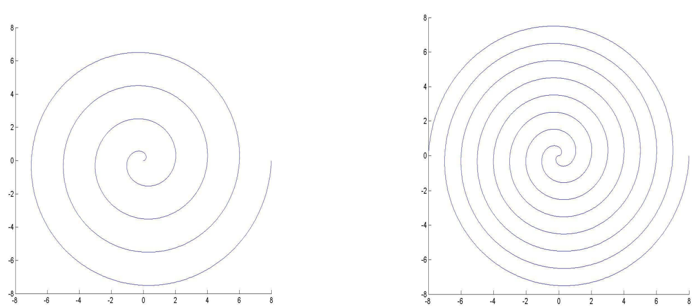
Trochoidal machining is one of new strategies in HSM [1]. A trochoid curve is formed by the locus of a point fixed to a curve, while the curve rolls on another curve without slipping. Trochoidal machining is well suited to HSM because the cutting tool always moves along a curve of constant radius. This allows a consistent feed rate to be maintained throughout the machining process. However, most part of the paths performs non-effective cutting.

One more new technique that particularly suited for HSM is the new G-code G6.2, which represents the NURBS spline. This command expands the choices from traditional linear and circular interpolation to interpolation along a spline represented by control points and knot points. By consolidating a complex, curving tool path into a single line of the program, this function saves on NC data, potentially resulting in more fluid high speed machining [1].

3. BACKGROUND INFORMATION

3.1 Spiral Curves

Spiral curves are curves that wind themselves round some certain points. While not being a circle, the radius will vary along the angle. There are different types of spiral curves, such as the Archimedean' spiral, Archimedes' spiral, atom-spiral, Atzema spiral, to name a few as shown in Fig. 2.



(a) An Archimedean spiral (b) An Archimedean spiral with two arms

Fig. 2. Archimedean spiral curves.

An Archimedean spiral is a curve which in polar coordinates (r, δ) can be described by the equation

$r = a + b\delta$

with real numbers a and b. Changing the parameter a will turn the spiral, while b controls the distance between the arms. Also, if δ takes negative as well as positive values, then the spiral curve will have two arms as shown in Fig. 2(b).

Because there is a linear relation between radius and the angle, the distance between the windings is constant. In this paper, we will use this property to find a path which can be used to cover a 2D region with minimal number of sharp turns and try to maintain the step-over value between two consequence paths (arms).

3.2 Medial Axis/Surface Transforms

The Medial Axis of a 2D region is defined as the locus of the centre of all the maximal inscribed circle of the object as shown in Fig. 3 [8-10]. Normally, physical objects are represented at a low level as a collection of boundary topological elements in most commercial computer aided design systems. Medial Axis captures the geometric proximity of the boundary elements in a simple form. It results in an abstract shaped representation known as skeleton. Medial Axis provides the unique description of shape which is of lower dimension than the original object. In this paper, we will only consider the cases that the Medial Axis consists of line, arc segments, or points. The term Medial Axis Transform is used to describe a method which reduces an object to its medial axis and associated radius function (distance from the medial axis to the closest point on the object boundary) and provide a unique representation of the object.

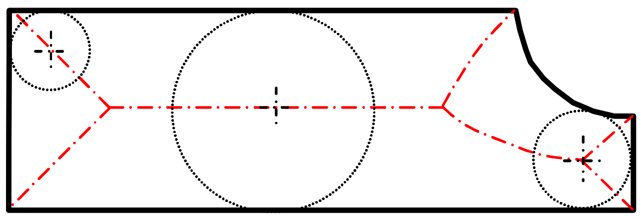


Fig. 3. Example of Medial Axis

In order to build a path based on spiral curves, if the medial axis of a 2D pocket can be found, then, we can try to use the medial axis as centres of segments of spiral curves. After that, we can try to link those spiral curve segments together to form a path which covers the retire pocket. This is the starting point of generating path for 2D regions using spiral curves.

4. ALGORITHM FOR GENERATING 2D PATHS

4.1 Problem Formulation

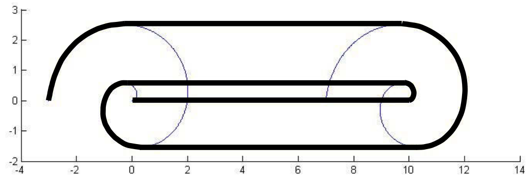
As defined in [3] and [11], given a pocket, the target region is the region that has to be covered. The obstruction region is the region that the mover cannot intersect with during moving. The region that can be covered by locating a mover C with radius r at a given point p is the circular region R whereas for every point q in R, the distance from p to q is less than or equal to r. A general definition of 2D region covering problem is to cover the target region without intersecting with the obstruction region. We assume the given mover is small enough to cover the entire target region. Without losing generality, we assume that the target region is a connected region. On the other hand, for each connected target region we can use the approach introduced in this paper to generate efficient paths.

The path generation problem is defined as: given a connected target region T and obstruction region O, and given a mover with radius r, find an efficient continuous path P such that: 1) for every point p in T, there is a location of the cutter on P to cover p; 2) for every point q on P, the covered region of locating the cutter at q is inside T.

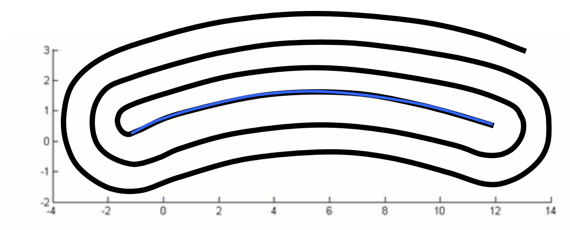
4.2 Definitions

A spiral curve centred at a given point ends at some angle θ is defined as a *spiral segment*. Traditionally, when a spiral curve is defined, it is always starting from a centre point, and by using the equation $r = a + b\theta$, a curve is un-winded as shown in Fig. 2.

Consequently, consider all points on a line or arc segment, if they are centres of spiral curves with the same a and b, and if we first try to create all spiral curves based on these points, and then find the bounding curves of these spiral curves given the equation $r = a + b\theta$, then, there will be a special spiral curve formed which is centred at the given line or arc segment, and also has the property that the distance between two neighbouring arms has the same value, which is controllable by b as shown in Fig. 4. Thus, we can use this property to create a general case of spiral curves. Just try to distinguish this spiral curve from traditional spiral curves which centred at a given point, this type of spiral curve is called *spiral curve centred at a curve segment*.



(a): Spiral curve centered at line segment



(b): Spiral curve centered at line segment

Fig. 4. Modified Archimedean Spiral Curve

Thus, if a curve segment is given, we can create a spiral curve centred at the given curve segment. In path planning problems, the spiral cannot un-wind infinitely. In other words, it has to be stopped at some point otherwise it will intersect with the obstacle or boundary of a given pocket. Thus, if a spiral curve is stopped at some point, the stopping θ is called the *bounding* θ, and the distance from the outermost point on the spiral curve to the centre is called the *bounding distance*. Usually, for a spiral curve centred at a point, the bounding distance is easy to find, i.e., find the nearest distance from the centre to the boundary of the pocket, which control the un-winding of the spiral curve. The same approach can be generated to find bounding θ of spiral curves centered at curve segments.

Given a spiral curve segment, either it is centered at a point or centered at a curve segment, if we know the bounding θ, we can estimate the area it covers. This area is called a *covered area*. Ideally, if the covered area of a spiral curve segment is equal to the target region of a given pocket, then the objective path is the same spiral curve segment. However, in most cases, one spiral curve segment can only cover part of the target region; we have to build several spiral curve segments in order to cover the entire target region. Because the path we expect has the least number of sharp turns, we want each spiral curve segment to be as large as possible. Thus, we use the greedy algorithm in finding each spiral curve segment. The idea is to find the spiral curve segment inside of the un-covered area whose covered area is as large as possible.

Thus, we can use different spiral curve segments to cover a region, with the condition that each spiral curve segment is bounded by some conditions, and the union of all covered areas of these spiral curve segments is the target region.

4.3 Algorithm

Input: A connected pocket P, a circular shape mover with diameter d, the value a, the overlapping rate, or the value controlled the distance between two neighbouring arms in one spiral curve segment b.
Output: A near optimal continuous path that can cover the whole region.

1. Use Medial Axis Transform to find the pocket's medial axis M and associated radius function R.
2. In the medial axis M, find the curve segment c on M by using which as the centre of a spiral curve segment, the covered area A is maximal. Usually this curve segment is a line or arc segment, but sometimes it is also a point.

3. Build spiral curve segment centred at *c*, and the bounding condition can be found by considering the boundary of the pocket and the obstacles.
4. Subtract the covered area *A* of spiral path segment centred at *c* from the initial pocket *P*, form a new pocket *P*', and go to step 1, until all area is covered.
5. Build a graph from all spiral curve segments, and find a path to link then together.
6. Link each spiral curve segment together, form the final path.

It is noticed that the algorithm is based on greedy strategy, and the optimal solution cannot be guaranteed. However, it can find a path which consists of different spiral curve segments, and can cover the entire region. Also, in some cases, with the growth of the covered area, the leftover area will be smaller and smaller, until only areas near the corners of the pocket boundary have not been covered. Therefore, if we still use spiral curve to cover those areas, it will result in many smaller segments. In this case, we can use the strategy of adding one or more contour parallel curves to cover the areas near the boundary of the pocket, for the rest region, we use the spiral curves to cover them, and then link the adjacent spiral curve and contour parallel curves by morphing. Thus, the resulted path will be a combination of several contour parallel paths, and several spiral curves segments.

5. IMPLEMENTATION AND EXAMPLES

We did a prototype implementation and conducted experiments using our algorithm. In this environment, after import a pocket, we can create the Medial Axis, and then generate spiral curves segment by segment.

As shown in Fig. 5, after the Medial Axis is generated, we can find the maximal covered area along the Medial Axis if a continuous spiral curve segment is created. Then, we create the spiral curve segment. After that, we subtract the covered area from the initial pocket, then for the rest area, we create Medial Axis and then find next spiral curve segment. However, in this case, there are two ways to generate the path. In the first case, we can still create spiral curves for the rest area. But as the left over area is smaller in size, the spiral curve segments created for the left over area is also in smaller pieces. After that, we can easily link those spiral curve segments together. However, we can also use another manner to cerate one or more contour parallel curves along the boundary of the initial pocket to cover the area along the boundary. After that, an interpolation is used to morph from the spiral curve to the contour parallel curve. In this case, the left over area is covered by the contour parallel paths instead of using smaller pieces of spiral curve. We can try different combinations of spiral curves and contour parallel curves until we can find the best combination. Thus, another optimization algorithm can be used here.

For most cases, the generated path performs superior to the existing zigzag or contour parallel path as shown in Fig.5 because there is no sharp turn and the step-over value between two neighbouring paths is a constant value. Therefore, the path generated by the new algorithm can guarantee that along most of the path segments, the step-over is constant, and the number of sharp turn-overs is limited.

6. CONCLUSION AND DISCUSSION

In this paper, a novel path planning algorithm for HSM is presented. The generated path is based on spiral curves, and has the following advantages:

1. For each spiral curve segment, there is no sharp turn-over, thus, a milling cutter can move rather smoothly.
2. The distance between adjacent path segments in a spiral curve is a constant value. This value can be used to control the step-over value between two neighbouring path. If this value is set to be the radius of the mover, then the mover covers the region with maximal efficiency.
3. Also, this constant step-over can be used to control the so called cutter engagement value in milling operations, which satisfies the requirement of constant cutting load, thus, in high speed machining, the cutting speed can be effectively controlled.
4. A greedy algorithm is used to generate a sequence of spiral curves, and then a graph search algorithm can be used to link these segments together to cover the entire region.

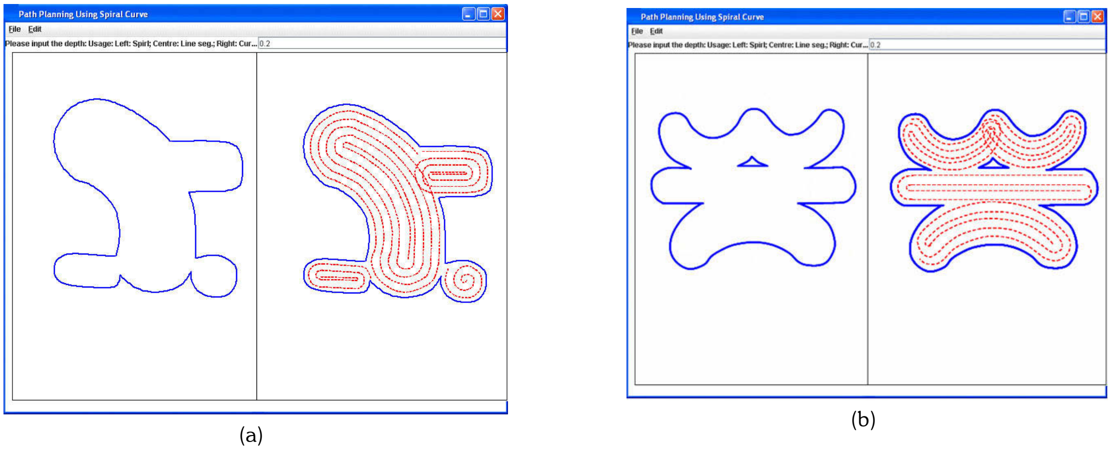


Fig. 5. Implementation Examples

There are several issues need to be tackled in the future:

1. For connected line or curve segment along the medial axis, the best way to generate a spiral curve should be further explored.
2. The currently used greedy algorithm in finding the spiral curve segments may not guarantee the optimal path, different approaches should be studied.
3. Currently used CNC code in interpolating a cutter path is either linear or circular. However, in order to interpolate a spiral curve, a better interpolation should base on spline curves, which will result in a much more efficient manner. Nowadays, several new CNC codes such as G6.2 , which represents a NURBS curve, is introduced by some companies, thus, we should later explore the possibilities of interpolating a modified spiral curve by NURBS curves [1].

7. REFERENCES

[1] Marinac, D., Tool path strategies for high speed machining, *Modern Machine Shop*, Vol. 72, no. 9, pp. 104-110. Feb. 2000.

[2] Gasparraj, E., Constant material removal: the key to hard milling, *Modern Machine Shop*, Vol. 77, no. 9, pp. 86-88. Feb. 2005.

[3] Yao, Z. and Gupta, S. K., Cutter path generation for 2.5D milling by combining multiple different cutter path patterns, *International Journal of Production Research*, Vol. 42, No. 11, 2004, pp 2141–2161.

[4] Bieterman, M. B. and Sandstrom, D. R., A curvilinear tool-path method for pocket machining, *Journal of Manufacturing Science and Engineering*, Vol. 125, Issue 4, 2003, pp 709-715.

[5] Sarma, R., An assessment of geometric methods in trajectory synthesis for shape creating operations, *SME Journal of Manufacturing Systems*, Vol.19, No.1, 2000, pp 59-72.

[6] Arkin, E. M., Fekete, S. P., and Mitchell, J. S. B., The lawnmower problem, In *Proc. 5th Canad. Conf. On Computational Geometry*, Waterloo, Ontario, 1993.

[7] Arkin, E. M., Bender, M. A., Demaine, E. D., Fekete, S. P., Mitchell, J. S. B., and Sethia, S., Optimal covering tours with turn costs. In *Proc. 12th ACM-SIAM Sympos, Discrete Algorithms (SODA'2001)*, January, 2001.

[8] Blum, H., A transformation for extracting new descriptors of shape, In *W. Wathen-Dunn, editor, Models for the Perception of Speech and Visual Form*, 1967, pp 362-380.

[9] Wolter, F. E., Cut locus and medial axis in global shape interrogation and representation, *MIT, Dec. 1993 (revised version), MIT Design Laboratory Memorandum 92-2 and MIT Sea Grant Report*, 1992.

[10] Hoffmann, C. M., How to construct the skeleton of CSG objects, In *A. Bowyer and J. Davenport, editors, Proceedings of the Fourth IMA Conference, The Mathematics of Surfaces*, University of Bath, UK, September 1990. Oxford University Press, New York, 1994.

[11] Yao, Z., Gupta, S. K., and Nau, D. S., Algorithms for selecting cutters in multi-part milling problems, *Computer-Aided Design*, Vol. 35, No. 9, 2003, pp 824-838.