Path Generation for High Speed Machining Using Spiral Curves

Zhiyang Yao[1] and Ajay Joneja[2]

[1]Chinese University of Hong Kong, zyyao@mae.cuhk.edu.hk
[2]Hong Kong University of Science and Technology, joneja@ust.hk

ABSTRACT

As high speed machining is becoming more and more important in modern machines shops, generating efficient cutter path for high speed machining is highly desirable. For this type of path, it should have the least number of sharp turns as well as the second derivatives along the path is controllable is preferred. Archimedean spirals have no sharp turn, and the step-over between adjacent segments can also be controlled to a predefined value. A clothoid spiral has the property that the second derivative varies linearly with the length of the curve. Therefore, these spiral curves can be used to generate paths with constant cutter engagement values as well as maintain smooth movements while performing cornering and linking movements. In this paper, we will introduce a detailed approach which covers a 2D region in a very efficient manner by using the combination of Archimedean and clothoid spirals.

Keywords: Path Planning; High Speed Machining; Archimedean Spiral; Clothoid Spiral.

1. INTRODUCTION

High speed machining is becoming more and more important in modern machines shops, especially in aerospace machining, mold and die machining. High Speed Machining (HSM is short) is the use of higher spindle speeds and feed rates to remove material faster as well as maintain the quality of part finishing [1] .Ideally, along a cutter path in HSM, 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 cutter (in this paper, we will only focus on flat-end milling cutter) to cover a larger 2D region called *pocket* that is bounded by a set of line or arc segments. Although if the scallop height is considered, this approach can also be extended to other types of cutters, in this paper we only consider flat-end mill for the purpose of computational simplicity.

In the 2D path planning problem, along the path, there are usually three types of movements: (1) cutter moves along the path to cover region that hasn't been fully covered (it is called an effective move); (2) cutter moves in the region that has been fully covered before (it is called a 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 a retraction). Both the repositioning move and the retraction are used to link disconnected effective moves together. These moves result in path segments that increase the total covering time. Therefore, in order to have an efficient path, such moves should be eliminated or minimized. Even for the effective moves, when the cutter 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 cutter 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.

Moreover, as mentioned before, if the cutter moves along a path with shape turns, for example, the so called turn-over in milling operations, the speed of the cutter has to be slowed down and then the cutter 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. For some cases when an extra path segment has to be used to link disconnected path segments together, the movement of cutter has to be controlled such that the movement should be smooth enough to prevent a sudden cutting load on the cutter as well as the machine. Therefore, a path that curvature is controllable, for example, within an upper bound, is also preferred.

Thus, the ideal path should be a path with minimal number of repositioning move, minimal number of sharp turn, constant step-over value, with most efficient effective moves and the curvature or the 2nd derivative is controllable. 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.

An Archimedean spiral 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. A clothoid spiral has the property that the second derivative varies linearly with the length of the curve. Therefore, it can be used to replace sharp turns and also be used in linking several disconnected segments together.

In our previous work, preliminary findings of applying modified Archimedean spirals in path planning are presented [3]. However, there are several un-solved problems in [3], for example, finding the maximal area a cutter can cut along a spiral path, which will be studied in detailed in this paper. Moreover, the problem of linking disconnected segments is not tackled in [3], which will also be covered. Compared to the algorithm proposed in [3], in this paper, the newly proposed path planning algorithm uses a combination of Archimedean and clothoid spiral, which can be used in generating efficient milling cutter paths for HSM.

In this paper, Section 2 reviews some background information of spiral curves and path planning using modified Archimedean spiral curves. Section 3 briefly introduces the algorithm in finding coverable area for a given spiral, which is critical in finding the centers of Archimedean spirals. Section 4 introduces the application of clothoid spiral in path planning. Then Section 5 presents the algorithm of generating spiral paths for 2D pockets, and section 6 provides 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. BACKGROUND AND RELATED WORK

For detailed information about path planning in HSM, please refer to [3] and [4]. Especially in [4], research work and industrial applications in HSM path planning are reviewed in detail. In this section, we will only focus on using spirals in path planning for HSM. Although some CAM providers also include spiral movements in their path planning packages like Catia, Cimatron, MasterCAM, etc., most of these paths center at given points. Another commonly used path in HSM is the so called trochoidal paths, which can be found both in CAM providers like MasterCAM and PowerMill, as well as research works [5, 6]. However, trochoidal movements require a large amount of empty movements, which lowers the cutting efficiency.

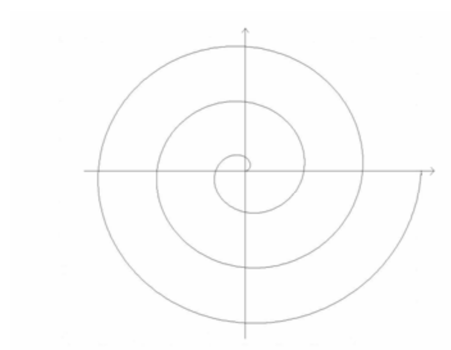


Fig. 1: Archimedean spiral with three rounds.

2.1 Archimedean Spiral

Archimedean spiral can be made by a point moving in a circle with constant speed. It starts in the origin and makes a curve with un-winded arms as shown in Fig. 1. The distances of intersection points along a line through the origin are same.

An Archimedean spiral is usually represented by $r = a + b\theta$, for a simple version, we take $r = b\theta$, thus we have its parametric form

$$x(\theta) = b\theta\cos(\theta)$$
$$y(\theta) = b\theta\sin(\theta)$$

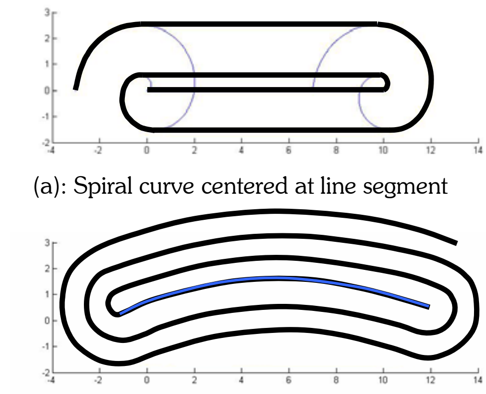
where b is the constant which defines the distance between two adjacent arms. The curvature of an Archimedean spiral is given by

$$k = \frac{2+\theta^2}{b\left(1+\theta^2\right)^{3/2}}$$

Complex geometries are in general made up of sharp corners. A path that has the least number of sharp turns is preferred in high speed machining. Archimedean spiral is a special spiral curve which can have no sharp turn, and the step-over between adjacent segments can also be controlled to a predefined value [3], i.e., the value b in $r = a + b\theta$ defines the step-over value between two adjacent path. Thus the spiral tool paths are more suitable for high speed machining than in traditional machining.

2.2 Extended Archimedean Spiral

Consider all points on a continuous curve segment (for example a linear or circular one as shown in Fig. 2), 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 neighboring arms has the same value, which is controllable by b. 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, these spiral curves are centred at curve segments, which are called *extended Archimedean spirals* (EASs). We now have a new definition of Archimedean spiral, which can be simply represented by the following: $A(c, b)$, where c is either a point or a continuous curve and b is the value controls the distance between two arms along the spiral. In this definition, is c is a point, then a traditional spiral is defined, if c is a curve, then an extended spiral is obtained. Notice that in practice, we prefer c to be continuous and not self-intersecting. Also, c should at least have C1 continuity, otherwise, the extended Archimedean spiral will have sharp turns itself, which is not preferred in HSM. Obviously, C1 continuity is a minimal requirement for center curves, because when growing the extended Archimedean spiral curve, it may overlap, which can be checked, and the growing process will be stopped.



(a): Spiral curve centered at line segment

(b): Spiral curve centered at curve segment

Fig. 2: Extended Archimedean Spiral.

Thus, if a curve segment is given, we can create an extended Archimedean spiral curve centred at this 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 extended Archimedean spiral curves.

2.3 Path Planning Using Archimedean Spirals

In our previous work, we proposed a preliminary algorithm in using extended Archimedean spirals in path planning[3]. The basic idea is to generate the medial axis for a given pocket, then for each segment along the medial axis we find the one which can covert the largest area, then we used that segment as a center to generate an extended spiral, and then for the left-over area, we iterate the same procedure until this pocket is fully covered by Archimedean spirals.

However, there are still several unsolved issues with that approach. Here are three major unsolved problems.

1. In [3], a center is found within the curve segments along the medial axis which can cover the largest area. However, in practice, sometime an extended Archimedean spiral which centered at part of a curve segment may cover the largest area because when the Archimedean is un-winded, it may intersect with the boundaries of the pocket. Therefore, we need a systematic way of finding the area that an extended Archimedean spiral can cover before we can find the proper centers.
2. Each time after an Archimedean spiral is used, the area it can cover is extracted from the initial pocket, and then a new medial axis is created. However, this may results in too much computation work, and also, the left over area is irregular after some portions are extracted, which will result in a medial axis with smaller segments. However, if we allow some overlapping between two adjacent cutter paths, we do not need to re-generated the medial axis again and again, we can always follow the initial medial axis without re-considering the segments along the medial axis which has been covered before.
3. In [3], several disconnected Archimedean spirals has to be connected using straight line, which may result in un-necessary sharp turns. However, we can apply the clothoid spiral in solving this problem.

2.4 Clothoid Spiral

The clothoid spiral is also called Euler's spiral, or spiral of Cornu (Fig. 3). The characteristic property of clothoid is that its curvature is a linear function of the arc length, or in other words the curvature is proportional to the length of the curve measured from the origin of the spiral.

A general definition of clothoid spiral is given by $k(s) = \sigma s + k(0)$. The parametric form of a clothoid is given by the Fresnel integrals

$$x(u) = \int_{u=0}^{t} \cos\frac{u^2}{2}du$$

$$y(u) = \int_{u=0}^{t} \sin\frac{u^2}{2}du$$

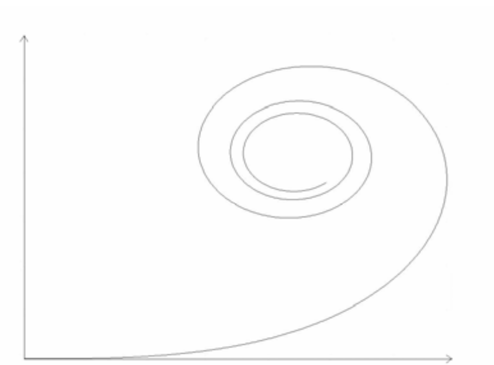


Fig. 3: A Clothoid Spiral.

where u is a non-negative parameter. As u approaches ∞, the curve spirals inwards towards its centre $\pm\left(\sqrt{\pi/2}, \sqrt{\pi/2}\right)$. The Fresnel integrals can only be solved by numerical methods. However, according to the definition, the curvature of a clothoid changes linearly with the arc length.

In the traditional machining cutting tools undergo sudden changes in directions as they approach the corners, and thus the acceleration of the tool decreases instantaneously. There is also a sudden increase in resultant forces acting on the cutting tool. The usage of clothoid spiral can smoothen the sharp corners of the traditional tool paths and reduce the magnitude of the sudden direction changes that the cutting tool has to undergo [7].

3. COVERABLE AREA OF ARCHIMEDEAN SPIRAL

Given an extended Archimedean spiral $A(c, b)$, either c is a point or a curve segment, if we know the bounding θ, we can estimate the area it covers. This area is called a *coverable area*. Ideally, if the coverable 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 coverable area is as large as possible.

When using the algorithm in [3], one of the important elements is to find the maximal area that a cutter can cut along an extended Archimedean spiral. There are some research works on finding the coverable area of a cutter in cutter selection problems [8-14]. These approaches usually assume that the cutter can cover a circular region as long as we can locate the cutter inside of the pocket. In other words, the cutter can move along any path in order to reach that center located. However, our problem is different in the manner that the cutter path is along extended Archimedean spiral, no any arbitrary movement. Therefore, this problem can be defined as the following: given a 2D pocket P bounded by a set of linear or arc segments and a cutter C, given a set of possible centers of Archimedean spirals (these centers can be points, line or arc segments or a smooth continuous non-self-intersecting curve), given a constant step-over value (in other terms, given the distance between two arms along the Archimedean spiral b), find the maximal area the cutter can cover without intersecting the boundaries.

In the approach we used in generating spiral cutter paths, we start with the medial axis of a given pocket, and then along the medial axis (which is represented by $\{c_i\}$, where each c_i is either a linear segment or a parabola), we try to find the center segment of an Archimedean spiral which can cover the maximal area. We will use the slicing method in finding the coverable area. The approach is as following. For any given point along the medial axis, we know the maximal enclosed circle, thus, we can build a 3D graph using the medial axis. In this graph, the xy plane contains the pocket along with the medial axis; the z axis indicates the diameter of the maximal enclosed circle for each point along the medial axis as shown in Fig. 4. After this graph is built, we can use the following algorithm in finding the maximal area a cutter can cover along an extended Archimedean spiral.

FindingMaximalCoverableArea (P, C, $\{c_i\}$, b)

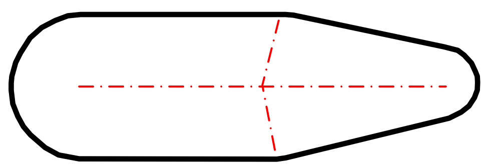
1. Build a 3D graph G based on the aforementioned method.
2. Start with $z=b$, slice the graph using a plane parallel to the xy plane, and then increase z by b
3. For each z

3.1 Find the intersection curve $\{c_i'\}$ with G, where the projection of each c_i' on xy plane $p(c_i')$, should have the following property: $p(c_i') \subseteq c_i$ (along the medial axis, each c_i is either a line or parabola segment)

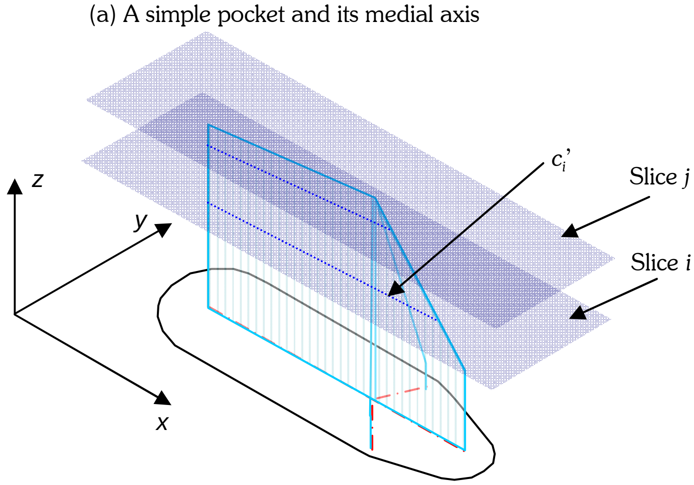
3.2 For each c_i', the coverable area is obtained by $a_i = z \times$ length of (c_i').

3.3 Store the maximal value a_i^* for all a_i, and corresponding $p(c_i')^*$ along this slice

4. Find the maximal value of a^* among all slices, and record the segment along the medial axis $p(c_i)^*$ as well as corresponding z value.



(a) A simple pocket and its medial axis



(b) Build a 3D graph G using medial axis and the diameter of maximal enclosed circle at each point

Fig. 4: Find the coverable area of an extended Archimedean spiral.

The above algorithm finds the maximal area the cutter can cover. It is clear that this is an approximation, because it assumes that the coverable area of an Archimedean centered at a point is equal to the circular area centered at the point. However, this approximation is acceptable because in path planning, we do not require the Archimedean spiral to cover exact the area it can maximally covered. Meanwhile, when an extended Archimedean spiral $A(c, b)$ is built by using the slicing method, for any point p along the center curve c, the maximal distance between this point to the boundary of pocket P is less than the maximal enclosed circle's diameter at that point, which proof that $A(c,b)$ will not intersect with the boundary. After the graph G is built, the running time of this algorithm is linear to the number of slices, i.e., if the diameter of the maximal enclosed circle of the pocket is d, then the running time is $O(\lfloor d/b \rfloor)$.

Now, we have a concrete algorithm in

finding the maximal area a cutter can cover as well as the center of the Archimedean spiral. We can use the results in the algorithm.

4. LINKING ALGORITHM USING CLOTHOID SPIRAL

What we can obtain so far is a set of Archimedean spirals. The next problem we are facing is to link these segments together in a proper manner. As discussed before, we would like the final path be smooth, in other terms, we want the 2^{nd} derivative along the path can be controllable. Therefore, we will apply clothoid spiral in linking those segments together. There are several different approaches in applying clothoid spiral in linking two different curvilinear segments together, for example [7]. We will adopt these approaches in this research. For example, the approach used in [7] proofed that a so called bi-clothoid can be used to link two straights lines together, and this bi-clothoid exists in between the arc segments linking these two lines and the intersection point of these two rays as shown in Fig. 5. Also, it is clear that the curvature changes smoothly when using clothoid spiral compared to when using arc segments.

For each pair of ending segments along two Archimedean spirals, we create a ray along each segment. In case these two rays intersecting inside of the pocket and they are not intersecting with the boundary, we create a clothoid near the intersecting point. In case these two rays intersects at a point outside of the boundary, or they intersects with the boundary, or they have no intersection, we create extra linear segments inside of the pocket and also intersect with these two rays, then, at each intersection point along these extra linking segments, we create a clothoid spiral. With that, we will guarantee that the linking is inside of the pocket, and also, the linking segment is smooth in term of the 2^{nd} derivative.

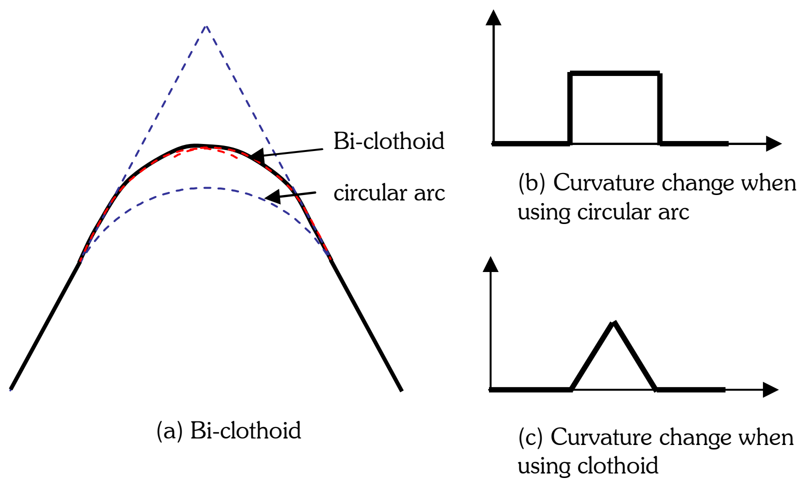


Fig. 5: Use clothoid spirals in linking two segments together.

5. ALGORITHM FOR GENERATING 2D PATHS

With all the newly developed techniques, we have a modified algorithm in generating spiral curves in HSM. 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.

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. Apply algorithm FindingMaximalCoverableArea (P, C, $\{c_i\}$, b), return center c
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. find the segments on medial axis that is covered by the above spiral curve, subtract them from $\{c_i\}$, and do iteration from step 2 (as shown in Fig. 6) until the maximal coverable area for the leftover area is less than a predefined value,

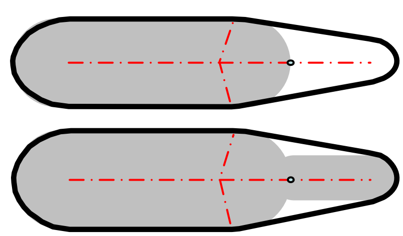


Fig. 6: After one archimedean spiral is built, use the rest segments along medial axis to find the next spiral.

which terminates the process when the Archimedean spiral is too small

5. Build a graph from all spiral curve segments, and each connected pair, use a clothodial spiral to link them together

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.

6. IMPLEMENTATION AND DISCUSSION

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. 7, 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 medial axis segments that are covered, then for the rest segments on medial axis, we create the next spiral curve segment. If the left over area is smaller in size, then we stop the process, for the rest area, we use offset curves to cover them. After that, we can easily link those spiral curve segments together. For most cases, the generated path performs superior to the existing zigzag or contour parallel path as shown in Fig.7 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.

The work presented in this paper is different from the one in [3] in the following manners:

1. We studied the problem of coverable area of extended Archimedean spiral in this paper.
2. The centers are curves instead of only linear and circular segments.
3. The clothoid is applied in linking disconnected segments together.
4. The new algorithm in generating the spiral path only requires one set of medial axis.

This new algorithm can generate paths that:

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

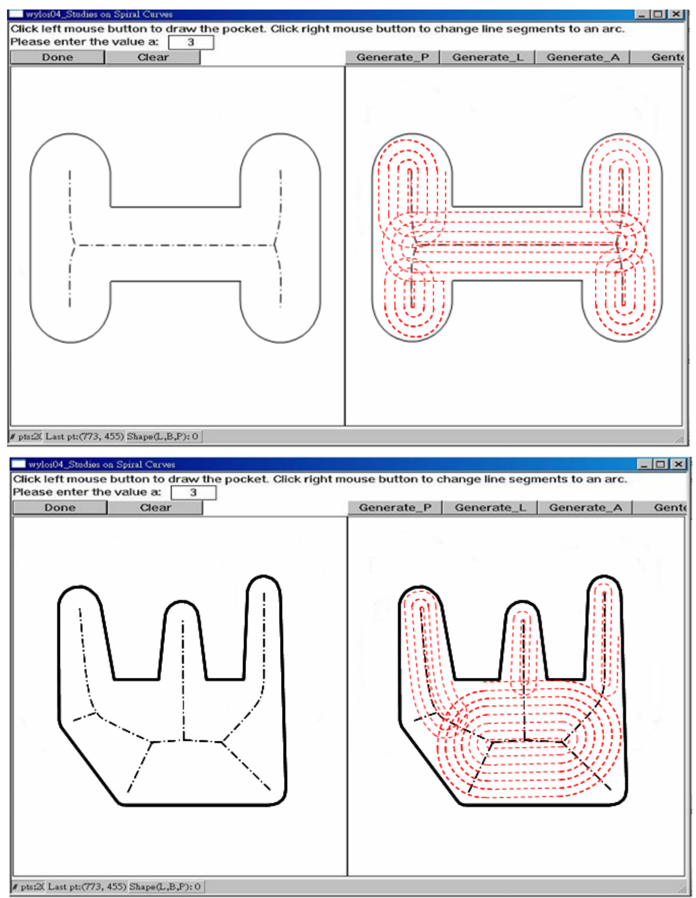


Fig. 7: Two Examples.

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. The linkage segment is smooth in terms of 2^{nd} derivative is linear with the curve length.
5. 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.

In the current algorithm, we consider the center curve of a spiral to be machined by a cutter that is fully engaged with the material. Obviously, this is not preferred in HSM. However, this problem can be easily solved by applying trochoidal movements when creating centers. In other words, instead of creating the center using one single path, we use trochoidal path along the center line, such that the final path will be acceptable for HSM. In the future, we would like to explore and implement more linking algorithms. For example, besides clothoid spiral, there are other types of curves like Reeds and Shepp curves we can use [15]. Meanwhile, we will study the problem of covering the left-over portions near the boundary of the pocket, and merging different types of path segments together.

7. REFERENCES

[1] Marinac, D.: Tool path strategies for high speed machining, Modern Machine Shop, 72(9), 2000, 104-110
[2] Gasparraj, E.: Constant material removal: the key to hard milling, Modern Machine Shop, 77(9), 2005, 86-88.
[3] Yao, Z.: A novel cutter path planning approach to high speed machining, Computer-Aided Design and Applications, 3(1-4), 2006, 241-248.
[4] Ramana, K. V.; Yao, Z.: Tool path planning in high speed machining, Submitted to the International Journal of Process Planning, 2006.
[5] Elber, G.; Cohen, E.; Drake, S.: MATHSM: medial axis transform toward high speed machining of pockets, Computer-Aided Design, 37, 2004, 241-250.
[6] Elber, G.; Cohen, E.; Drake, S.: C1 continuous toolpath generation toward 5-axis high speed machining, Computer-Aided Design and Applications, 3(6), 2006, 803-810.
[7] Pamali, A. P.: Using clothoidal spirals to generate smooth tool paths for high speed machining, MS thesis, North Carolina State University, 2004.
[8] Bala, M; Chang, T. C: Automatic cutter selection and optimal cutter-path generation for prismatic parts, International Journal of Production Research, 29(11), 2163-2176, 1991.
[9] Lee, Y. S.; Chang, T. C.: Application of computational geometry in optimization 2.5D and 3D NC surface machining. Computers in Industry, 26(1), 1995, 41-59.
[10] Lee, K; Kim, T. J.; Hong, S. E.: Generation of toolpath with selection of proper tools for rough cutting process, Computer-Aided Design, 26(11), 1994, 822-831.
[11] Veeramani, D.; Gau, Y. S.: Selection of an optimal set of cutting-tool sizes for 2.5D pocket machining, Computer-Aided Design, 29(12), 1997, 869-877.
[12] Lim, T.; Corney, J.; Clark, D. E. R.: Exact tool sizing for feature accessibility, International Journal of Advanced Manufacturing Technology,16, 2000, 791-802.
[13] D'Souza, R.; Wright, P.; Séquin, C.: Automated microplanning for 2.5-D pocket machining, Journal of Manufacturing Systems, 20(4), 2001, 288-296.
[14] Yao, Z.; Gupta, S. K.; Nau, D. S.: Algorithms for selecting cutters in multi-part milling problems, Computer-Aided Design, 35(9), 2003, 824-838.
[15] Fraichard, T.; Scheuer, A.: From Reeds and Shepp's to continuous-curvature paths, IEEE Transactions on Robotics, Dec., 20(6), 2004, 1025-1035.