MACHINING OF ARCHIMEDEAN SPIRAL BY PARAMETRIC PROGRAMMING

Vratraj Joshi[1], Keyur Desai[2], Harit Raval[3]

[1] Department of Mechanical engineering, Ichchhanath Circle, Athwa, Surat, Gujarat 395007, India
[2] Department of Mechanical engineering, Ichchhanath Circle, Athwa, Surat, Gujarat 395007, India
[3] Department of Mechanical engineering, Ichchhanath Circle, Athwa, Surat, Gujarat 395007, India

Corresponding author: Vratraj Joshi, vkj.fetr@gmil.com

Abstract: Considering quality and design requirement, machining of various curve shapes is a necessity in today's manufacturing world. Straight line segments are used to connect consecutive cutter location points of the tool path which results in non-linearity error for every interpolation. CAD software is necessary to machine curve shape. Computer numerical control (CNC) interpolators for curvilinear tool path can't achieve desired position accuracy as massive data are communicated between CAD software and CAM systems. Length and memory size of the part program are increased as each linear segment is adding single block. Parametric programming is attempted to machine curved profile (Archimedean spiral). The purpose of the present work is to machine Archimedean spiral expressed by mathematical equation using parametric programming. Surfaces quality for the machined tool path is measured and reported. The aim is to propose a tool path to achieve desired shape accuracy by adapting the geometry of the curve.

Key words: Archimedean spiral, parametric programming, Shape error, Surface roughness

1. INTRODUCTION

Various curve shape components are used in industries. Machining of these curved shape on CNC machine causes difficulty as appropriate preparatory functions are not available. Understanding of commercial Computer Aided Design (CAD) software is required to plot these curves in routine practise. The software generates preparatory codes automatically from X and Y co ordinates of the cutter location (CL) points on that curve. Cutter interpolates tool path linearly between two successive CL points of the segments. It results into deviation of the tool path from the actual curved profile to be machined. This deviation of the tool is directly proportionate with a curvature of the geometry. To reduce the deviation, numbers of CL points are to be increased by decreasing length of each linear segments of the tool path. Each linear segment to machine the curve is adding single block in the part program. This will increase length of the part program and add to program memory size. Program memory size will be increased at the cost of the surface quality. To achieve better shape accuracy, parametric interpolation (MACRO programming) is required. Number of CL points can be enhanced without affecting program length in case of parametric programming. It has been widely used due to its simplicity and can avoid requirement of the CAD software. Any curve which can be expressed by different mathematical expression (parametric expression) can be machined by MACRO programming.

Like any other computer programming language, programming with MACRO can be prepared while working with CNC Fanuc controller. It can be executed by preparatory codes of CNC controller. It is a flexible programming that can utilize conditional and logical loops, arithmetic variables and operators. A subprogram is required for parts with similar machining operations. But, for parts with the similar design, a single MACRO part program is to be developed that can considerably reduce part programming time. It can be further beneficiary in product development. It is to be prepared frequently for changing workpiece variable such as dimensions, holes, threads, slots etc. A single MACRO program can machine similar parts and most suitable for a same part family because tool position coordinates are not needed to be changed every time as and when the part is changed. It helps in reducing duplication of the similar program. Some of the commands used in MACRO program are GO TO, IF [condition] THEN [do something condition], EQ [equals], NE [not equals], GT [greater then], LT [Less then], +, -, *, /, SIN, COS, TAN, ATAN, SQRT, ABS etc. Good amount of work has been carried by various authors to develop algorithms for command generation to reduce program length as well increase shape flexibility.

Literature related to machining of curved tool path, spline interpolation and algorithm development to minimize position error is studied. In present work, MACRO and Conventional programming techniques for machining curved shape with continuously changing radius (Archimedean spiral) are studied and compared. Explicit function is used while programming with MACRO interpolator.

M. Boujelbene et al. (2004) suggested that second order continuous tool path comprising arc and straight line segments can be compared with first order continuous tool path. Yuan-Jye Tseng and Yii-Der Chen (2000) observed that the final surface produced may lack smoothness due to interpolation of line segments. The number of tool contact points and the number of segments are reduced compared to the linear approximation methods. Daniel C.H. Yang and Tom Kong (1994) reported that parametric interpolator is favorable in the machining of free-form geometry. The linear interpolator has the advantages of simplicity. Feed is varied according to numbers of cutter location points to achieve predetermined deviation of the tool path with the actual profile. Qing Zhen Bi et al. (2012) carried out analytical computation to determine control points for the curvature continuous tool path. Q. Liu et al. (2010) discussed the relative deviation of the computed feed rate. Difference between the desired feed rate and the actual feed rate is calculated. Interpolation algorithm for explicit parametric curve is proposed. The algorithm is dominant factor which affect the machining accuracy of CNC machine tools. The chord error can be decreased if a smaller feed rate is chosen at the segment of the curve with large curvatures. The algorithm is general and applicable to any smooth curve that can be formulated by parametric equations without specific computation.

In the present work, numbers of cutter location points are determined according to curvature of the path. C.H. Chu et al. (2008) proposed an error estimation method to construct the envelope surface of the real tool motion and compared it with the design surface to interpolate tool positions for better surface quality. The improvement is observed for the large curvature surface region. In present work, Archimedean spiral tool paths are machined considering 2μm, 3μm, 4μm and 5μm tolerance from the actual profile. Numbers of control points are calculated accordingly.

2. MACHINNING OF ARCHIMEDEAN SPIRAL

From the literature, it is noted that geometrical data of the cutter path is stored in a CAD model and transformed to CAM to machine the curve. Commands for CNC machining can be developed that convert motion trajectory from the desired path. Parametric (i.e. MACRO) programming is attempted and compared with the Conventional approach.

Constant separation distance between two consecutive turnings is the salient property of Archimedean spiral curve. Equation (1) is equation for the radius of the spiral curve. It shows that radius ‘r’ varies with variation in angle ‘θ’ (Lawrence, Dennis J., 2014).

$$r = a + b\theta \tag{1}$$

$$X = r\cos\theta \quad Y = r\sin\theta \tag{2}$$

Equation (1) provides radius ‘r’ of the spiral curve for the value of ‘θ’ that varies at regular interval. X and Y co ordinates are calculated at various points on curved geometry using Archimedian spiral function. Value of X and Y co ordinates of the cutter location points are required to machine the curve. From equation (2), it is cleared that the values of X and Y co ordinates are dependent on value of ‘r’ and as per equation (1).

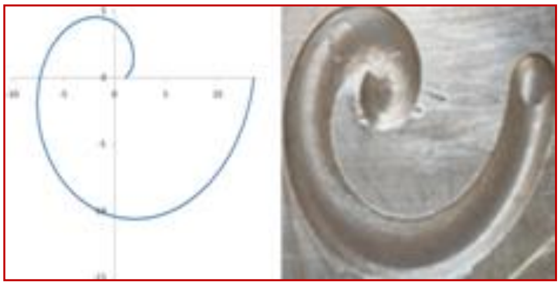

Fig. 1. Simulation and actual machining of Archemedian Spiral

Figure 1 shows interpolation and actual machining of Archimedean spiral. It is noted that value of ‘r’ is dependent on ‘a’ and ‘b’. So, by varying the value of ‘a’ and ‘b’, positions of X and Y coordinates can be changed. Equation (2) is equation to obtain value of X and Y co ordinates from the polar value of the radius ‘r’.

Table 1. Assigning MACRO address

MACRO Address	Variable	Description	Initial Value
#101	a	Determine inner radius	1
#102	b	Determine distance between successive path	2
#103	θ	Initial Angle	0
#104	r	Polar co ordinate	1
#105	X	X co ordinate of CL point	1
#106	Y	Y co ordinate of CL point	0

Table 1 shows variables stored at MACRO addresses. Total 100 cutter location points are determined to machine Archimedian spiral by MACRO programming method.

3. RESULTS AND DISCUSSIONS

Archimedean spiral is interpolated with MACRO programming. Angular increment starts from initial value ‘0^0’ and ends at ‘360^0’ with the increment of 3.6^0. It means total calculated cutter location points are determined by dividing the subtraction of maximum value and minimum value of angle with the step size [i.e. (360-0) / 3.6 = 100]. Drawing Archimedian spiral in AutoCAD with measuring cutter location points take more time. Table 2 shows a comparison between Conventional and MACRO programming for machining of the curved geometry. In the present case, time for drawing Archimedian spiral with 100 cutter location points for each quadrant in AutoCAD is 2758 seconds.

Table 2. Comparision of Conventional and MACRO programming

			Conventional programming	MACRO programming
		Length (Words)	540	57
		Size (KB)	14	0.38
Time (Seconds)	Program writing	Drawing	2758	0
		Feeding	1165	241
	Program Execution		35	73
		Total	3948	314

Time for feeding these co ordinates is 1165 seconds. In case of MACRO programming, cutter tool path is equation driven. No such drawing is needed. Hence, program feeding time (i.e.241 seconds) is very less. It is notable that machine feed is 100 mm/min and program execution time to cut Archimedian spiral by Conventional programming is 35 seconds. In case of MACRO programming, the processor has to calculated X and Y co ordinate for every cutter location points. So, program execution time is 73 seconds, more than double of Conventional programming. Considering the total time, program writing time (including drawing and feeding) with program execution time for the Conventional method to cut the Archimedean spiral is quite high (3948 seconds) compared to MACRO programming method (314 seconds). The additional advantage of MACRO programming is that program size is 14 KB and length 540 words for Conventional programming. Which is more than MACRO programming program size is 0.38 KB and length is 57 words.

Table 3. Mesurement of surface roughness

Sr	Surface Roughness (µ) Conventional programming	MACRO programming	Difference
1	3.951	2.894	1.057
2	3.892	2.977	0.915
3	3.967	2.841	1.126
4	3.882	3.184	0.698
5	3.853	3.014	0.839
6	3.858	3.146	0.712
7	3.831	3.259	0.572
8	3.785	2.873	0.912
9	3.783	2.881	0.902
10	3.842	3.112	0.730
11	3.549	2.951	0.598
12	3.814	2.971	0.843
Avg	3.834	3.009	
Std dev	0.106	0.136	

The Archimedian spiral is cut by Conventional and MACRO programming method. The surface quality is examined, for both the methods with the help of SURFTEST (SJ-210 Mitutoyo). The work piece is marked clockwise at 30^0 for the measurement of surface quality at 12 positions ($360^0/12^0$) of the test piece. Table 3 shows the surface roughness of Archimedean spiral (i.e. variable curvature) machined by Conventional and MACRO programming. The average value of surface roughness is quite less in case of MACRO programming method (3.009µ) compared to Conventional programming method (3.884 µm). The length of program is based on cutter location points co ordinates in case of Conventional prograaming. While, length of MACRO programming less as it is equation driven. Some time is required to execute the program by processor to calculate coordinates of cutter location points according to equation, the machine gets sufficient time to change the direction according to curvature region of the tool path, which causes the less fluctuation of cutting force. This minimizes the tool overdrive and hence improves the surface quality. Figure 2 indicates a comparison of the average surface roughness for both programming methods.

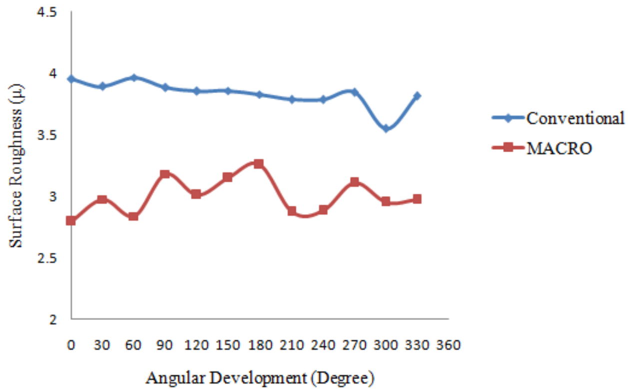

Fig. 2. Simulation and actual machining of Archemedian Spiral

As shown in Figure 2, surface roughness is less in case of the Archimedean spiral cut with MACRO compared to Conventional programming. The tool stops for a while processor calculates co ordinates of cutter location point. It improves surface quality while machining Archimedian spiral using MACRO programming method.

3.1 Measurement of shape accuracy

Cutting tool experiences tangential force near curved section and tends to move in tangential direction. As tool has to move on the profile according to processor's instruction written in part program, dimensions of machined tool path width are found little more than dimension of tool diameter. The term shape error is introduced in the present analysis. Shape error is the difference of radial distance of arcs and chord. The origin is aligned with the centre of the curve to calculate the shape error. Shape error can be decreased by reducing the difference between arc and chord length between successive cutter location points. Arvind Narayan, and Mohammad Farooque Khan approaches are studied to determine arc length.

Arvind Narayan approach

Arvind Narayan (2012) reported a relationship between the arc and the chord length of tool path as given below.

$$Arc\_length = \frac{L}{2\sin\frac{(\theta_f - \theta_i)}{2}}(\theta_f - \theta_i) \tag{3}$$

θ_i is subtended angle at Point P (X_1, Y_1) on the tool path with the horizontal. Similarly θ_f is subtended angle at Point Q (X_2, Y_2) with horizontal. Equation (3) shows the relation between the arc and chord length assuming constant radius of curvature between two points on the tool path (i.e. Point P and Point Q). The accuracy of this method is based on the selection of a numbers of control points. Here, L is chord length which is obtained from equation (4).

$$L = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \tag{4}$$

The tool path arc is assumed to equal length of circular arc. Chord length is calculated according to the transformed circular arc. The assumption of constant radius of curvature between two points is the limitation of this approach.

Mohammad Farooque Khan approach

Mohammad Farooque Khan (2013) developed a model to find the relation between the arc and chord. Point P (X_1, Y_1) and point Q (X_2, Y_2) are points of a curved profile. The relation between the arc length (L) and a chord length is as given in the equation (5) and (6). Computation of Arc length introduced by this approach is based on geometrical theorems. As per this method, arc length is simply 1.11 times more than chord length.

$$Arc\_Length = \frac{\pi}{2\sqrt{2}}\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \tag{5}$$

$$Arc\_Length = \frac{\pi}{2\sqrt{2}} \times Chord\_Length \tag{6}$$

Hence, the method is fast and simplest compared to Arvind Narayan approach. Method suggested by Arvind Narayan (2012) has more variables and involves more steps to compute. The salient feature of this method is that it does not require differentiation. Start and end points co ordinates are sufficient input to determine the arc length of the curve. The main drawback of the method is the ratio of the arc length to chord length. It is observed that the ratio is not constant in case of Archimedean spiral. Due to this, the method proposed by the Mohammad Farooq Khan is less adaptable for Archimedean spiral.

The other approach to reduce shape error is to reduce Chord height. Daniel C.H. Yang approach is studied to calculate chord height for curved profile.

Daniel C.H. Yang approach

Daniel C.H. Yang (1994) proposed the model based on chord height. Chord length is denoted by ‘a’. Figure 3 shows that vertical distance from control point to centre and chord height are mentioned by ‘r’ and ‘h’ respectively. Radius of arc is shown as “R”. The accuracy of Daniel C.H. Yang approach depends on magnitude of curvature between two control points and the length between two control points.

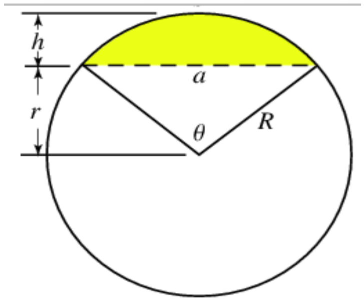

Fig. 3. Shape error for tool path, (Daniel, C. H., Yang, 1994)

As per Daniel C.H. Yang model, Chord length ‘a’ can be calculated as per equation (7).

$$a = \sqrt{8Rh - 4h^2} \tag{7}$$

Considering all above approaches, one method is proposed in the present work to improve shape accuracy.

Proposed method (Joshi approach for measurement of shape error)

In proposed method, centre of curvature is transformed to origin. Points on both sides of arbitrary point at definite angular distance are selected. These points are joined by curve as well as chord. Arbitrary point is joined to origin with line. Segment of this line between curve and chord is defined as shape error. Shape error is varied with angular movement of tool between these two points.

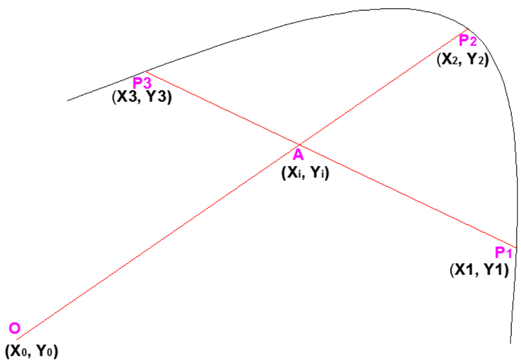

Fig. 4. Shape error in tool path

As shown in Figure 4, line OP_2 is passing from origin. Line OP_2 is intersected by chord P_1P_3 at point A. Line segment AP_2 can be represented as shape error. It is noted that shape error is not perpendicular to chord. It varies as per this approach. $P_1(X_1, Y_1)$ and $P_3(X_3, Y_3)$ are points on chord as. O (X_0, Y_0) is a centre of the curve aligned with an origin. Line passing from origin comprises end point P_2 (X_2, Y_2) on the arc of Archimedean spiral. From the equation (8) and (9), value of variables 'P' and 'Q' can be obtained as:

$$P = X_2Y_1 - X_1Y_2 \tag{8}$$

$$Q = (X_2 - X_1)Y_2 - (Y_2 - Y_1)X_2 \tag{9}$$

The point of intersection can be obtained by substituting equation (8) and (9) in equation (10).

$$(Xi, Yi) = \frac{PX_2}{Q}, \frac{PY_2}{Q} \tag{10}$$

The point of intersection (X_i, Y_i) is to be put in equation (11) to find shape error. It is noted that the approach utilizes X and Y co ordinates of points P_1 and P_3 of the curve.

$$L = \sqrt{(X_i - X_2)^2 + (Y_i - Y_2)^2} \tag{11}$$

Hence, this method facilitates position of point P_2 arbitrarily on the curve. It means that when position of point P_2 is changed on tool path, co ordinates (X_i, Y_i) of intersection point A will be changed from equation. Less computational time is required to calculate shape error in present approach. The shape error calculated with Joshi approach is compared with the CAD software. Joshi approach is adopted to decide location of the cutter location points to achieve the predetermine value of tool path deviation. It is observed that the deviation is more as the curvature is more and it reduces when the curvature is less. It is required to have cutter location points having different chord length to minimize the deviation or to have relatively equal deviation throughout the length of an arc/curve.

Though, deviation is achieved with the predetermined/acceptable value. Numbers of cutter location points effect on the length of the chord, ultimately total cutter path length. With this logic, it is also planned to apply present approach to study on the variable chord length per cutter location points. The present method approach is attempted to determine cutter deviation from the curvature tool path. The numbers of control points depends on the magnitude of predetermined deviations viz. 2µm, 3µm, 4µm and 5µm. Control points on tool path are placed at angular interval to get the predetermined value of tolerance. Hence, definite interval between control points is also to be changed to get the desired value of shape error. Number of program set is decided to position cutter location point at fixed interval to confine shape error within predetermined value. Tool path for the particular interval is to be divided into number of segments considering curvature of the tool path with the purpose to obtain uniform cutter deviation.

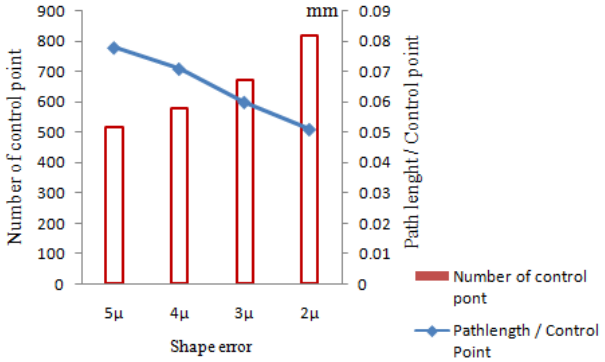

Fig. 5. Effect of number of control point on shape error

Corresponding program is run for every segment. The first program set is a program to machine near major axis. This portion consists more curvature. Therefore, numbers of control points are more. Hence, path length per control point is less. It is observed from Figure 5 that numbers of control points are increased as value of shape error is decreased (i.e. from 5µm to 2µm). Path length per control point is decrased as the value of shape error is decreased (i.e. from 5µm to 2µm). Table 4 shows that path length per control point is increased as the tool moves on curved profile for the same value of shape error. It is justified as curved portion becomes relatively less and control points can be positioned relatively at far distance from each other to achieve predetermed shape error. It means for the 2µm predetermined of the shaper error, value of shape error is dependent on the angular distance between control points. Hence, control points should be placed nearer to each other and numbers of control points required are more. Numbers of control points are increased near the region of curvature. Hence, cutter is less deviated from the actual profile. Numbers of program set, path length and control points to machine entire tool path are mentioned in Table 4. Considering curvature of the tool path, to get a uniform cutter deviation of 5µm, tool path for the particular angular interval is to be divided into number of segments. Corresponding program is to be run for every segment. The first program set is having more curvature; hence each control point is required to be nearer. So, path length per control point is less (i.e. 0.054 mm). The last program set is a program set to machine path length with less curvature. So, path length per control point is more (i.e. 0.112 mm). Same calculations are carried for the 4µm, 3µm and 2µm deviation, Control points for 5µm shape error are less in number (i.e. 519), while for 2µm shape error, control points are more (i.e 818). This is because; angular distance between control points will be less to achieve less shape error. For that, control points should be placed nearer to each other.

Table 4. Comparision of conventional with MACRO programming

5µm					4µm				3µm				2µm			
Program set	Included Angle	Path Length	Control points	Path Length / Control points	Included Angle	Path Length	Control points	Path Length / Control points	Included Angle	Path Length	Control points	Path Length / Control points	Path Length	Included Angle	Control points	Path Length / Control points
1	1.67	0.271	5	0.054	1.48	0.300	6	0.050	1.25	0.464	10	0.046	20.496	0.98	21	0.039
2	1.48	0.384	6	0.064	1.30	0.514	9	0.057	1.06	0.781	16	0.049	33.840	0.75	45	0.040
3	1.34	0.502	8	0.063	1.14	0.730	13	0.056	0.89	1.332	27	0.049	55.776	0.58	96	0.044
4	1.21	0.692	11	0.063	1.01	1.087	19	0.057	0.75	2.308	44	0.052	93.150	0.45	207	0.053
5	1.09	0.967	15	0.064	0.88	1.683	28	0.060	0.64	4.249	73	0.058	154.386	0.35	443	0.067
6	0.98	1.344	20	0.067	0.78	2.584	40	0.065	0.54	8.043	120	0.067	1.620	0.27	6	0.064
7	0.89	1.921	27	0.071	0.69	4.179	59	0.071	0.46	15.356	199	0.077				
8	0.80	2.826	37	0.076	0.60	6.677	85	0.079	0.38	15.488	185	0.084				
9	0.73	4.049	49	0.083	0.53	10.985	125	0.088								
10	0.66	6.033	67	0.090	0.47	17.890	181	0.099								
11	0.59	8.872	90	0.099	0.41	1.379	14	0.099								
12	0.54	13.086	121	0.108												
13	0.49	7.047	63	0.112												
			519	0.078			579	0.071			674	0.060			818	0.051

4. CONCLUSION

MACRO programming of curved profile eliminates need of CAD software subjected to expression of curve in terms of mathematical expression. Machine processor gets sufficient time to calculate cutter location points co ordinates. Hence, tool stops for a moment at the preceding point while processor calculates the next point. It gives enough time to track the direction as per curvature of the tool path. This leads to less cutting force fluctuation and favours to achieve good surface finish. Program is prepared for curved shape motion of the tool. Though, cutting force directs tool in tangential direction. Due to this contrary, cutting force fluctuation is occurred. Value of surface roughness is more for Conventional programming compared to MACRO programming methods. Arvind Narayan and Mohammad Farooque Khan approaches are observed to be less accurate. Required shape accuracy can be achieved by determining the numbers of control points with the present method (Joshi approach).

5. REFERENCES

1. Arvind, Narayan, (2012). *Perimeter of the Elliptical Arc a Geometric Method*, IOSR Journal of Mathematics, 3(2), 8-13.

2. Mohammad, Farooque Khan, (2013). *Arc length of an elliptical curve*, International Journal of Scientific and Research Publications, 3(8), 1-5.

3. Chu, C. H., Huang, W. N., Hsu, Y. Y. (2008). *Machining accuracy improvement in five-axis flank milling of ruled surfaces*, International Journal of Machine Tools & Manufacture, 48, 914–921.

4. Daniel, C.H. Yang, Tom, Kong, (1994). *Parametric interpolator versus linear interpolator for precision CNC machining*, Computer Aided Design, 26, 225-234.

5. Lawrence, Dennis J., (2014). *A Catalog of special plane curves*, Dover Publication, pp. 174.

6. M., Boujelbene, M., Moisan, A., Tounsi, N., Brenier, B., (2004). *Productivity enhancement in dies and molds manufacturing by the use of C^1 continuous tool path*, International Journal of Machine Tools & Manufacture, 44, 101–107.

7. Qing, Zhen Bi, Yong, Qiao Jin, Yu., Han Wang, Li, Min Zhu, Han, Ding, (2012). *An analytical curvature-continuous B´ezier transition algorithm for high-speed machining of a linear tool path*, International Journal of Machine Tools & Manufacture, 57, 55-65.

8.Q., Liu, X., J. Jin, Y., H. Long, (2010). *A real-time high-precision interpolation algorithm for general-typed parametric curves in CNC machine tools*, International Journal of Computer Integrated Manufacturing, 23(2), 168–176.

9.Yuan-Jye, Tseng, Yii-Der, Chen, (2000). *Three dimensional biarc approximation of freeform surfaces for machining tool path generation*, International Journal of Production Research, 38, 4739-763.

Received: June 5, 2016 / Accepted: December 10, 2016 / Paper available online: December 20, 2016 © International Journal of Modern Manufacturing Technologies.