# Automatic Tool Path Generation for Parametric Surfaces in Terms of Bezier Patches

**Dr.Tahseen Fadhil Abbas***

Received on :4/5/2008
Accepted on 9/10/2008

**Abstract:**

The objective aim of this paper is to develop an algorithm that generate NC tool path for parametric surfaces based on the accuracy of a desired surface. A designed surface is represented by sufficient control points , using these control points , the surface has been represented depending on Bezier technique to generate reliable and near-optimal tool path as well as cutter location data file for post-processing; the proposed algorithm include two functions:

- Forward step function that determine the maximum distance between cutter contact points (CC) with the given surface tolerance.
- Side step function which determine the maximum distance between two adjacent tool paths with a given scallop height.

The output of these two functions are engaged with the generated interior data of the desired surface and feed automatically to a vertical CNC machine (Bridge Port) through the serial port (RS232), then one tool paths has been implemented for manufacturing one parametric surface, with tolerance of 0.5mm, using vertical CNC milling machines (Bridge Port) with ball nose end mill cutter ø 14mm.

The proposed algorithm has been implemented for tool path generation of several parametric surfaces to illustrate its flexibility,

**Keywords**: Parametric surfaces, Bezier patches, Tool path generation, Forward step, Side step, Scallop height.

## توليد مسار العدة للسطوح البارامترية بستخدام مفهوم بيزر

**الخلاصة**

يهدف البحث الى بناء وتطوير خوارزمية لاشتقاق وتوليد مسار العدة لبرامج مكائن التفريـــز ذات التحكم العددي لتشغيل السطوح البارامترية معتمدين بذلك على الدقة المطلوبة لتشغيل السطح المعين. تم تمثيل السطوح باعتماد مضلع السيطرة وهو عبارة عن مجموعة من نقاط السيطرة التي تتحكم بالـــسطح البارامتري , حيث تم استخدام مفهوم "بيزر" للمنحنيات والسطوح لاشتقاق وتوليد البيانات الكاملة للسطح المعين ومن ثم استثمار هذه البيانات لتوليد مسار العدة بشكل مؤتمت.

تتضمن الخوارزمية المقترحة دالتين اساسيتين في توليد مسار العدة وهما :-

- دالة الانتقال الطولي باتجاه احد متجهي السطح .
- دالة الانتقال العرضي بالاتجاه العمودي على المتجه السابق.

حيث تم تعشيق مخرجات هاتين الدالتين وبيانات السطح المشتقة باستخدام مفهوم "بيزر" لتوليـــد مـــسار العدة لعدد من السطوح البارامترية المختلفة وقد اثبتت الخوارزمية المقترحة ملائمتها لمختلف السطوح , وللتأكد واختبار الخوارزمية تم ربط المخرجات بـــ ماكنة التفريـــز المبـــرمج ( Bridge Port ) مـــن خلال واجهة الربط ( RS232 ) وتشغيل احد السطوح البارامترية باعتمـــاد دقـــة [ 0.5 mm ] وباستخدام عدة تفريز كروية ذات قطر [ø14 mm] وكانت النتائج جيدة لجميع مراحـــل الخوارزميـــة المقترحة.

*Production Engineering and Metallurgy Department, University of Technology/ Baghdad

## Introduction

A CAD/CAM system within manufacturing facility generates a NC program containing a series of coded instruction directly from the CAD model. Also the NC program directly affects the accuracy and the cost of the part and makes specific trajectories on the part being processed called tool path. In milling operation, a tool moves along the tool path at cutter contact CC linearly, that is a curved surface is approximated by a series of straight line segments as shown in Fig.(1).

The accuracy of this linear approximation controlled by deviation is called tolerance. There is also scallop between adjacent tool paths. After machining a grinding operation needed to smooth the machined surface. Therefore appropriate tool path is very important to reduce the amount of secondary processes such as grinding. It is also important to generate tool path with less cutter location points with given tolerance and scallop height [1].

This paper focuses on developing an algorithm that generate tool path for parametric surfaces based on the accuracy of a desired surface. A designed surface is represented by mathematical Bezier curves, using the mathematical representation of the desired surface , the near optimal tool path will be generated as well as cutter contact points(CC) and cutter location points (CL) files.

CLdata provides readable data on cutter locations and machine tool operating commands. The machine tool commands can be converted to specific instructions during post-processing, the final output is a part program in a word address format that can be post-processed for (Bridge Port) CNC machine tool on which the job will be accomplished. The output of the post-processing is a part program consisting of G-Code [x, y, and z coordinates] and other functions in word address format.

## Literature Review

The aim of tool path generation is to approximate the part being processed with a number of curves that can be approximated by line segments.

Parametric surface representation is most widely used to model engineering surfaces in CAD/CAM systems, their capabilities to model complicated surfaces are better than other types of representation such as explicit and implicit representation.[2 ]. There are a number of different parametric forms such as Hermit, Bezier and B-splin, other more advanced forms are the rational forms such as: Rational Bezier and NURBS [3].Parametric surfaces, are often machined using a flat end mill for roughing and a ball end mill for finishing [4].Once a surface model is developed a tool path generation mechanism is invoked to generate the cutter contact file [CC file]. The size of the file depends mainly on the required accuracy [5].

Chen [6], presents two different approaches to generate the tool path: surface normal and offset surface, approximation are always used to describe the offset surface.

The goal of tool path generation algorithm is to develop an accurate and efficient tool path to machine object surfaces, Dragmatz [7], classified tool path generation into isoparametric and non-isoparametric paths , he deduced that the accuracy of the surface depends on the desired tolerance.

## Parametric Curves and Surfaces

A three dimensional curved line can be represented in an analytical form with the pair of functions:

$$y = f(x) \qquad \text{…..……. (1a)}$$

$$z = g(x) \qquad \text{…… (1b)}$$

With coordinate (x) selected as the independent variable, values for dependent variables (y) and (z) are then determined from equation …, the step through values for (x) from one line end point to the other end point. This representation has some disadvantages, when the aim is to plot the curve smoothly, then the independent variables whenever the first derivatives (slop) of either *f(x)* or *g(x)* become greater than one, and this demanded a continually checking the derivatives values that may become infinite at some points.

A more convenient representation of curves for graphics applications is in term of parametric equations. By introducing a fourth parameter (*u*), into the coordinate

description of a curve, each of the three Cartesian coordinates can be expressed in parametric form [8]; any point on the curve can then be represented by the vector function:

$$P(u) = (x(u), y(u), z(u)) \qquad \ldots. \ (2)$$

Usually the parametric equations are set up so that the parameter (u) is defined in the range from [0 to 1] .

Parametric equations for surfaces are formulated with two parameters [ $u$ and $v$ ], a coordinate position on a surface are then represented by the parametric vector function:

$$P(u,v) = (x(u,v), y(u,v), z(u,v))$$
$$(u,v) \in [0,1] \qquad \ldots\ldots.. \ (3)$$

In design applications, a curve or surface is often defined by interactively specifying a set of control points which indicate the shape of the curve, these control points are used to set up polynomial parametric equations to display the defined curve. When the displayed curve passes through the control points, it is said to interpolate the control points. On other hand, the control points are said to be approximated if the displayed curve passes near them. Many techniques exist for setting up polynomial parametric equations for curves and surfaces, basic methods for displaying curves specified with control points are Hermit, Bezier, and B-spline curves [9].

**The Proposed Algorithm**

In this section we will explain the proposed algorithm that used to generate tool paths, the algorithm steps are summarized in the flowchart shown in Fig.(2).

**Define Desired Surface**

Designed surfaces are presented using Bezier curves and surfaces with two parameters ( $u$ and $v$ ). In parametric surface, holding one parameter constant defines an iso-parametric curve. If one of the parameters reaches zero or one, the curve becomes exactly one of the boundary curves surrounding the surface. If both parameters are held constant, a point is specified on the surface patch

**Bezier Curve**

The degree of polynomial defining the curve segment is one less than the number of defining polynomial points. The curve generally follows the shape of the defining polygon. The first and last points on the curve are coincident with the first and last points defining polygon points. The tangent vectors at the ends of the curves have the same direction as the first and last polygon spans respectively. The curve is contained within the convex hull of the defining polygon. Four control points and resulting cubic Bezier curve are illustrated in Fig. (3).

Mathematically a parametric Bezier curve is defined by [10]:

$$P(u) = \sum_{i=0}^{n} B_{n,i}(u) Pi \qquad \ldots\ldots\ldots \ (4)$$
$$for \ldots\ldots n = 3$$

$$P(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

**Bezier Patches**

A Bezier patch is a special type of surface patch, defined by given a doubly indexed set of control points ( $Pij$ ), forming a control net to define the individual curves. Suppose the double indexing as an integer points in a rectangular grid. Each point in the grid is associated to a control point, and each connecting line is used to shape the surface. The surface can be viewed as a mapping of the rectangular grid into space [11].The definition of a Bezier patch is as a tensor product. The doubly indexed set of control points are viewed as an [( $m+1$ ) × ( $n+1$ ) × 3] matrix, that is a three dimensional matrix or a tensor. To generate a surface from the control net, each column or row of the control net can generate a Bezier curve. Fig. (4) illustrate the control grid and Bezier patch view as a mapping grid to space.

The size of the rectangular grid determines the type of Bezier patch. Given the grid is

$(m+1) \times (n+1)$ the surface is a polynomial function of degree $(m \times n)$, meaning the surface is sum of *mth* degree polynomial in the variable $(u)$ time's nth degree polynomial in the variable $(v)$ [12].The patch is thus functionally described by:

$$X(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n} P_{i,j} B_i^m(u) B_j^n(v) \quad .. (5)$$

A cubic Bezier patch can be written in a matrix form:

$$P(u,v) = \sum_{j=0}^{3}\sum_{i=0}^{3} P_{i,j} B_i^3(u) B_j^3(v)$$

$$P(u,v) = \sum_{j=0}^{3}\left[\sum_{i=0}^{3} P_{i,j} B_i^3(u)\right] B_j^3(v) \quad .....(6)$$

$$= \sum_{j=0}^{3}[1 \quad u \quad u^2 \quad u^3]\begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}\begin{bmatrix} P_{0,j} \\ P_{1,j} \\ P_{2,j} \\ P_{3,j} \end{bmatrix} B_j^3(v)$$

$$= \sum_{j=0}^{3}[1 \quad u \quad u^2 \quad u^3]\begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}\begin{bmatrix} P_{0,0} & P_{0,1} & P_{0,2} & P_{0,3} \\ P_{1,0} & P_{1,1} & P_{1,2} & P_{1,3} \\ P_{2,0} & P_{2,1} & P_{2,2} & P_{2,3} \\ P_{3,0} & P_{3,1} & P_{3,2} & P_{3,3} \end{bmatrix}\begin{bmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ v \\ v^2 \\ v^3 \end{bmatrix}$$

.

And so the cubic Bezier patch is frequently written:

$$P(t) = [1 \quad u \quad u^2 \quad u^3] M \begin{bmatrix} P_{0,0} & P_{0,1} & P_{0,2} & P_{0,3} \\ P_{1,0} & P_{1,1} & P_{1,2} & P_{1,3} \\ P_{2,0} & P_{2,1} & P_{2,2} & P_{2,3} \\ P_{3,0} & P_{3,1} & P_{3,2} & P_{3,3} \end{bmatrix} M^T \begin{bmatrix} 1 \\ v \\ v^2 \\ v^3 \end{bmatrix}$$

*Where*

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

The previous mathematical formulation of Bezier techniques have been implemented to built an algorithm to generate the interior data of a desired surface depending on initial control points (4×4).

**Tool Path Generation**

Three axis machining of surfaces usually starts with rough cutting using flat end mill cutter then the finish cutting are performed using ball end mill cutter. Isoparametric tool path have been used in this approach, surface points are calculated as a function of ( $u$ , $v$ ) parameter space, the tool path indexed along the surface by incrementing $(u)$ and $(v)$. Tool path planning is accomplished by holding the $(v)$ parameter constant and indexing the $(u)$ parameter, which is *forward step*. Forward step increment $(s)$ in $(u)$ direction must be carefully chosen since tool movements are linearly interpolated and the chordal deviation between the straight lines and the actual surface must be less than the desired tolerance $(\delta)$. Step-over increment in $(v)$ direction (*side step g*) must be small enough to keep the scallop height between spherically shaped cutter paths to less than the desired tolerance. Fig. (5) illustrates the tool path of a desired surface.

**Calculating of Forward Step Increment:**

The choice of forward step length depends on the calculated tolerance (δ) between the true curve and the chord between successive points on the curve. The maximum deviation occurs at $[r(u+1/2\Delta u)]$ for a sub-segment from $(u)$ to $(u+\Delta u)$ Fig.(6). This leads to a straightforward algorithm of the maximum for general parametric curves. [13]

The maximum normal distance (δ) for the chord joining the points at $(u=0)$ and $(u=1)$ is evaluated, because parameter transformation can be used to transform any segment ( $u_i \le u \le u_{i+1}$) into ($0 \le u' \le 1$).

$$u = (1-u)'u_i + u'u_{i+1} \quad ..... (8)$$

It can be seen that the parameter (u'=0,1) correspond to ( $u = u_i, u_{i+1}$) as required. So standard results to sub-segments ($u_i \le u \le u_{i+1}$), according to the geometric relationship as shown in Fig.(7)can be applied:

$$r(1/2) = r(0) + \lambda c + p \quad ..... (9)$$

In which the chord vector joining the points $r(0)$ and $r(1)$; then:

$$p = r(1/2) - r(0) - \lambda c$$

Since

$c.p=0$

It follows that

$$c.[r(\tfrac{1}{2})-r(0)=l\,|c|^2$$

So that

$$l=\frac{c.[r(\tfrac{1}{2})-r(0)}{(c)^2}$$

Then

$$p=r(\tfrac{1}{2})-r(0)-\frac{c.[r(\tfrac{1}{2})-r(0)}{(c)^2}.c \quad \text{..... (10)}$$

In general ( $l \approx \tfrac{1}{2}$ ), [13], then the formula (10) Can be simplified as:

$$p \approx r\tfrac{1}{2}-\tfrac{1}{2}[r(1)+r(0)] \ldots \text{ (11)}$$

This equation has been used in this work to compute a sequence of segments of maximum length which are within the tolerance specified, that is:

$$|p| \le d$$

**Convert Cutter Contact Points to Cutter Location Points**

The result of calculating the forward step is a CC point that can be any point on the tool, to reduce machining errors, it has to be converted to a CL point by which the tool moves along the surface [14].

$$CL = CC + rn \quad \text{.... (12)}$$

Where ( $r$ ) is the radius of the ball nose end mill cutter and ( $n$ ) is the surface normal at CC point on the surface as shown in Fig.(8).

$$n=(S_u \times S_v)/|S_u \times S_v| \quad \text{... (13)}$$

Where ***Su*** and ***Sv*** are the derivatives along the ***u*** and ***v*** directions on a surface ***S*** at point ***P***.

**Calculate the Side Step (*g*) with a given Scallop Height (*h*)**

A manufactured part can be approximated by a series of tool paths. When the parameter value of an iso-parametric curve reach one at the end of current curve, the side step (*g*), should be calculated for the next tool path. The side step size is the maximum distance between two adjacent tool paths which can keep the given scallop height (*h*)[1], as shown in Fig.(9).The side step (g) is a function of:

- The scallop height (*h*)
- Tool nose radius ( $r$ )

$$h = r-\sqrt{r^2-(g/2)^2}$$
$$g^2 = 4r^2-4(r-h)^2$$
$$g = 2\sqrt{r^2-(r-h)^2} \quad \text{.....(14)}$$

**Machining Code Generation**

After selecting the correct machining direction and machining side of surface, in order to generate the machining codes the tool moves along a direction, each successive tool path is divided into a number of straight segments. So machining of each segment could be represented by (G01) code. By calculating the coordinates of start and end points of each line segments, it becomes possible to generate the required part program.

**Implementation**

The proposed algorithm was developed and implemented for several surfaces for which the CC and CL points were generated and one of these surfaces machined using a vertical CNC milling machine with a desired tolerance and scallop height. The hardware and software used to implement proposed algorithm are:

**Hardware**: CNC Milling machine [Bridge Port].

**Software**: **MATLAB 6.5**

The proposed algorithm was coded in MATLAB Ver. (6.5) on a personal computer Pentium IV, 2.0GHz CPU, 1.0 GB of physical memory, operating under Microsoft XP Professional.

A parametric shaped part has been machined using a block of aluminium alloy, also a software are build to control data transfer from PC to BridgePort CNC machine through the serial port RS232. Fig. (10) Show the PC linked with the machine, the part during machining, and the finished part.

Several bi-cubic Bezier patches were designed and tested to generate the tool

path, as shown in the following examples, then one of these patches has been machined to illustrate the flexibility of the proposal algorithm in practical environment.

**Example One**: A bi-cubic Bezier Patch with the following control points

**px**

[0,0,0,0:50,50,50,50:100,100,100,100:150, 150,150,150]

**py**

[0,50,100,150:0,50,100,150:0,50,100,150: 0,50,100,150]

**pz**

[0,18,24,28;18,24,6,30;30,45,30,32;30,36, 30,50]

Fig. (11) illustrate the control net, mesh surface, tool path, and machined surface with a maximum allowable tolerance and scallop height of (0.5 mm), cutter diameter of ø14 mm. While the table (1) shows the required cutter location points to machine the desired surface using the proposed algorithm for different tolerances and different cutter diameters.

**Example Two**: A bi-cubic Bezier Patch with the following control points

**px**

[0,0,0,0:50,50,50,50:100,100,100,100:150, 150,150,150]

**py**

[0,50,100,150:0,50,100,150:0,50,100,150: 0,50,100,150]

**pz**

[10,15,65,70:10,15,65,70:10,15,65,70: 10,15,65,70]

Fig. (12) illustrate the control net, mesh surface, tool path, and solid model with a maximum allowable tolerance and scallop height of (0.3 mm), cutter diameter of ø12 mm, while the table (2) shows the required cutter location points to machine the desired surface using the proposed algorithm for different tolerances and different cutter diameters.

**Example Three**: A bi-cubic Bezier Patch with the following control points

**px**

[0,0,0,0:50,50,50,50:100,100,100,100:150, 150,150,150]

**py**

[0,50,100,150:0,50,100,150:0,50,100,150: 0,50,100,150]

**pz**

[10,10,10,10;10,60,60,10;10,60,60,10;10,1 0,10,10]

Fig. (13) Illustrate the control net, mesh surface, tool path, and solid model with a maximum allowable tolerance and scallop height of (0.4 mm), cutter diameter of ø16 mm. While the table (3) shows the required cutter location points to machine the desired surface using the proposed algorithm for different tolerances and different cutter diameters.

**Conclusions**

The proposed algorithm for tool path generation was developed and implemented successfully through the integration of mathematical modelling used for calculating forward step and side step size into the core of the proposed algorithm, additional contribution is related to mathematical representation of manufactured surface through the use of parametric curve and surface depending on Bezier form.

As it can be seen from tables 1, 2 and 3 the number of cutter location points [tool path length] is independent of surface type, but its greatly effected by allowable tolerance and diameter of end mill; as the allowable tolerance increase, using larger cutter diameter, the required number of cutter location points decrease that means the total length of the tool path will decrease and this will reduce cost of manipulation as well as storage. On other hand the cutter diameter should be selected carefully according to the curvatures of the curved surface to prevent under cut problems.

**References**

[1]-Young Keun choi, A. Banerjee," Tool path generation and tolerance analysis for free-form surfaces", International Journal of Machine Tools & Manufacturing, 2006.

[2]-J.D.Foley and A.Van Dam," Computer Graphics: Principales and Practice, New York, Addison-Wesley 2nd edition, 1997.

[3]-Kimura, F., Yamaguch, Y., Koboyashi, K, Nakajima, H.," Free-

form surfaces modelling for complicated product description ", Annual of the CIRP Vol.39/1/1990.
[4]-Oliver J.H., Wysocki, D.A., Goodman, E.D.,"Gouge free detection algorithms for sculptured surface NC generation ", Journal of Engineering for Industry, Vol.115, 1993.
[5]-Loney G.C., Ozsoy, T.M.,"NC machining of free-form surfaces", Computer Aided Design, Vol.19, 1987.
[6]-Chen, Y.J., Ravani, B.,"Offset surface generation and contouring in computer aided design", ASME, Journal of Mechanism, Transmission and Automation in Design, Vol.109, 1987.
[7]-Dragomatz, D., Mann, S.,"A classified bibliography of literature on NC milling path generation ", Computer Aided Design, Vol.29, No.3, 1996.
[8]-Farin, G.," Curves and Surfaces for Computer Aided Geometric Design ", Academic Press, INC., 2002.
[9]-M.E. Mortenson, " Geometric Modelling", New York, John Wiley & Sons, 1985.
[10]-M.E. Mortenson, "Mathematics for Geometric Modelling Applications", Industrial Press Inc. 1999.
[11]-Fredrik Anderson, "Bezier and B-spline Technology ", 2003. http://www.cs.umu.se/education/Reppporter/461.pdf.
[12]-Colm Mulcahy, "The basic curves and surfaces of computer aided geometric design ", Computer Aided Geometric Design, 1994.
[13]-Patrikalakis N.M., "Computational Geometry", Massachusetts Institute of Technology, USA, 2003
[14]-Young Keun Choi, "Tool path generation and 3Dtolerance analysis for free-form surfaces", PhD Thesis, Industrial Engineering, Texas A&M University, 2004.

**Table (1) the required cutter location points**

| Tolerance(mm) | Cutter Diameter ( mm ) | | | |
|---|---|---|---|---|
| | Ø10 | Ø12 | **Ø14** | Ø16 |
| 0.1 | 5854 | 4832 | 4162 | 3622 |
| 0.2 | 2972 | 2452 | 2160 | 1894 |
| 0.3 | 1982 | 1724 | 1408 | 1262 |
| 0.4 | 1562 | 1262 | 1124 | 994 |
| **0.5** | 1262 | 1058 | **872** | 758 |

**Table (2) the required cutter location points**

| Tolerance(mm) | Cutter Diameter ( mm ) | | | |
|---|---|---|---|---|
| | Ø10 | **Ø12** | Ø14 | Ø16 |
| 0.1 | 5854 | 4832 | 4162 | 3622 |
| 0.2 | 2972 | 2452 | 2160 | 1894 |
| **0.3** | 1982 | **1724** | 1408 | 1262 |
| 0.4 | 1562 | 1262 | 1124 | 994 |
| 0.5 | 1262 | 1058 | 872 | 758 |

**Table (3) the required cutter location points**

| Tolerance(mm) | Cutter Diameter ( mm ) | | | |
|---|---|---|---|---|
| | Ø10 | Ø12 | Ø14 | **Ø16** |
| 0.1 | 5854 | 4832 | 4162 | 3622 |
| 0.2 | 2972 | 2452 | 2160 | 1894 |
| 0.3 | 1982 | 1724 | 1408 | 1262 |
| **0.4** | 1562 | 1262 | 1124 | **994** |
| 0.5 | 1262 | 1058 | 872 | 758 |

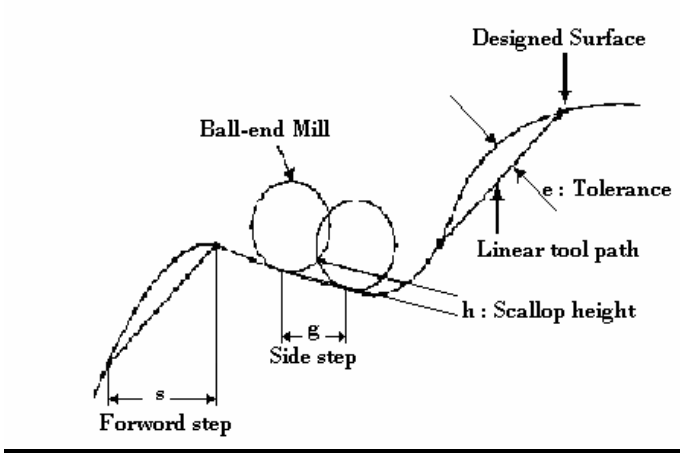


**Figure (1).Linear approximation of a curved surface**

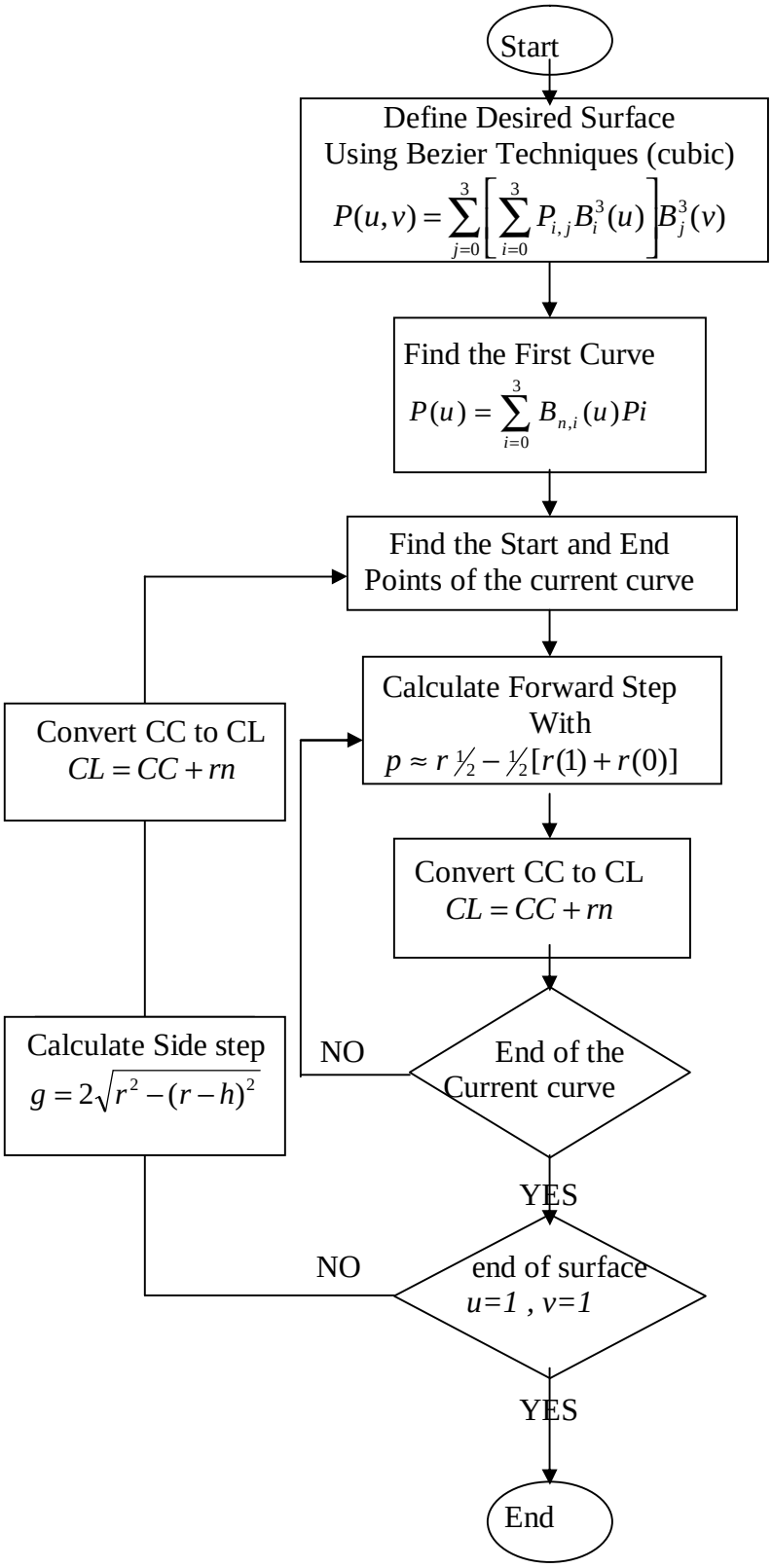


**Figure (2).Flowchart of the proposed algorithm**

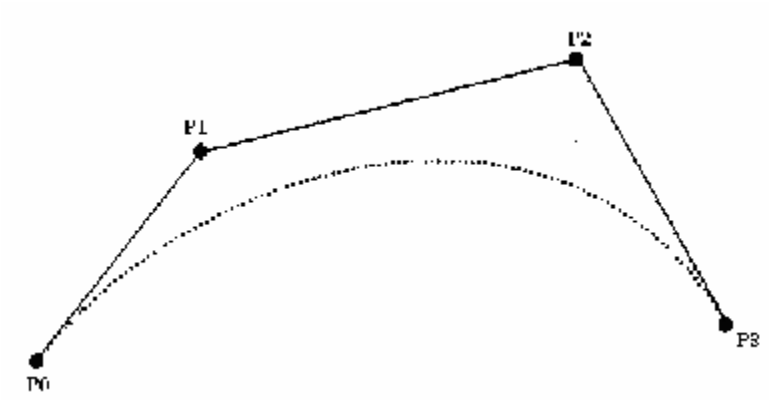


**Figure (3) Cubic Bezier curve with control points**

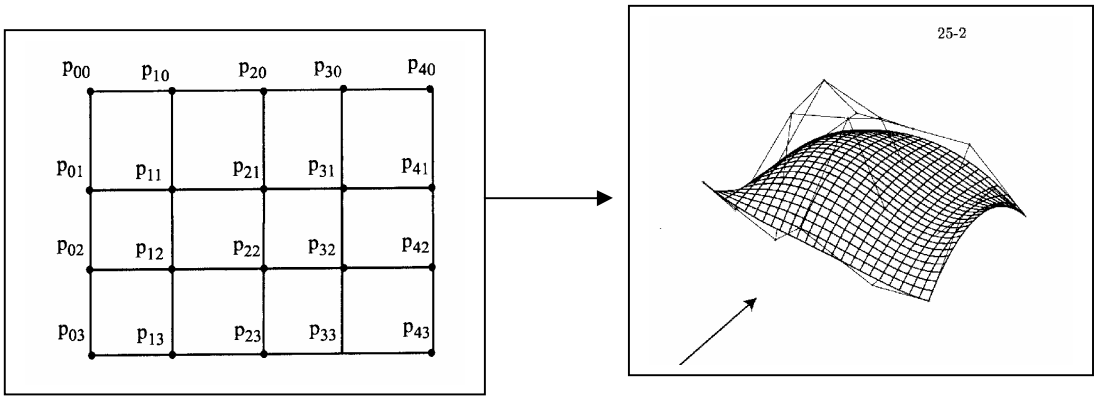


**Figure (4) mapping of control net to a Bezier patch.**

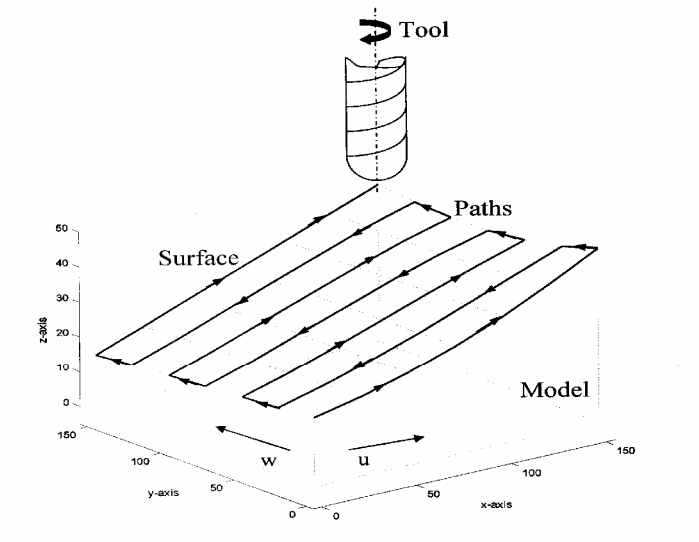


**Figure (5) Tool path**

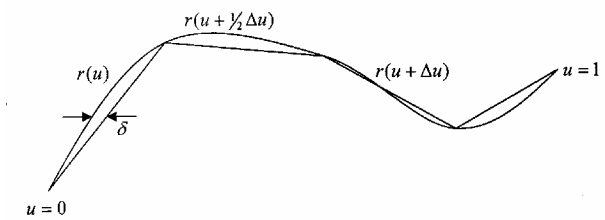


**Figure (6) Forward step**

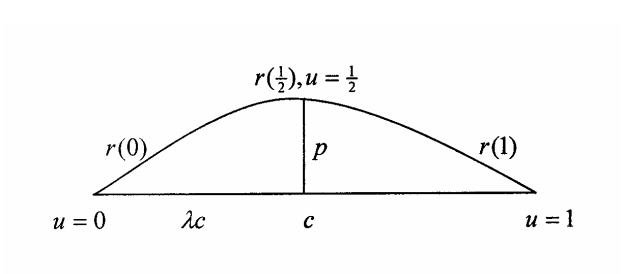


**Figure (7) Chord vector**

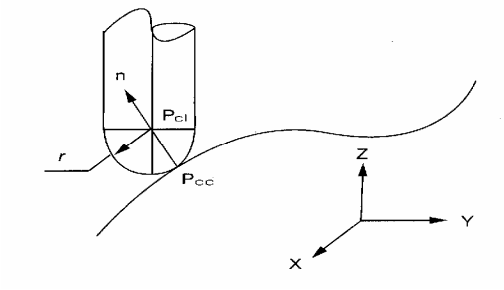


**Figure (8) Cutter contact and cutter location points**

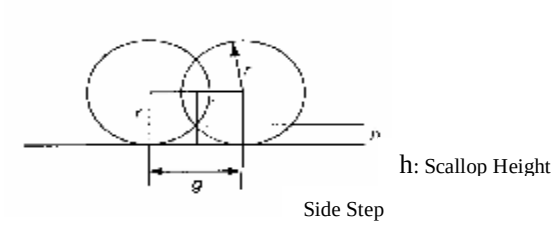


**Figure (9) Side step**

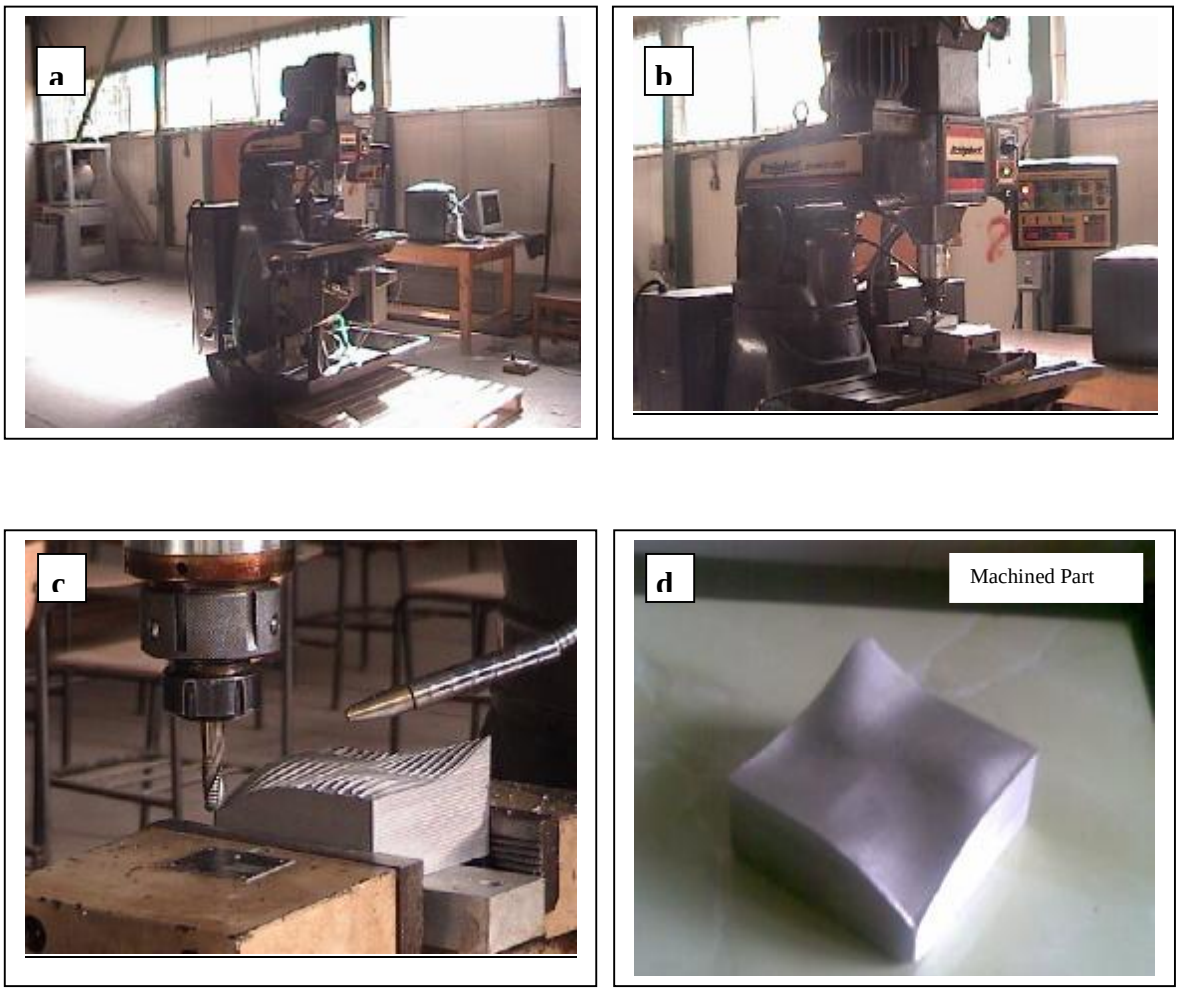


**Figure (10) Bridge Port CNC milling machine and the machined part**

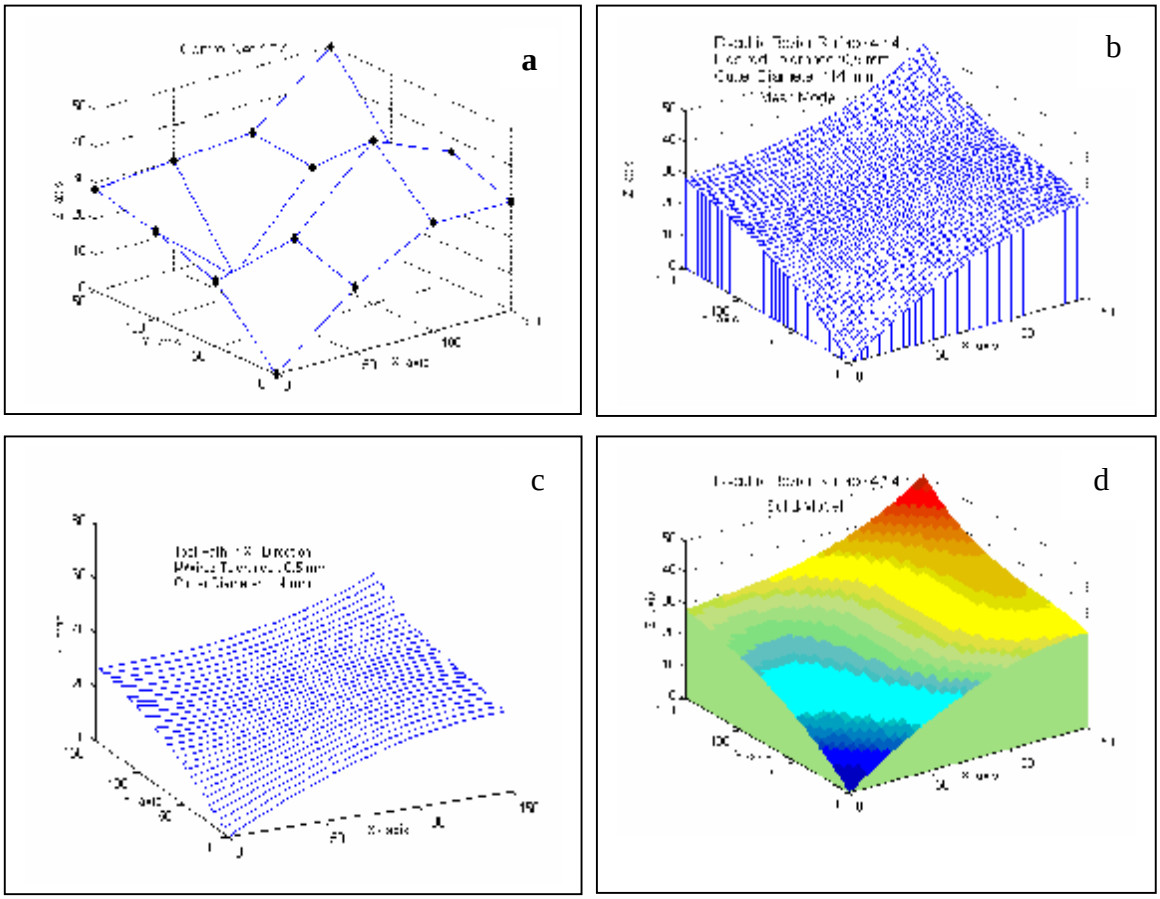


**Figure (11) a-control net. b-mesh model. c- generated tool path. d-solid model**