MATHEMATICAL BASIS FOR THREE DIMENSIONAL CIRCULAR INTERPOLATION ON CNC MACHINES

A J LUBBE

Department of Industrial and Systems Engineering

University of Pretoria

ABSTRACT

The control units of numerically controlled manufacturing machines allow the programmer only a limited number of mathematical functions with which programmes can be written. Despite these limitations it is now possible to write programmes with which three dimensional (3D) circular interpolation can be performed directly on the machines. The necessary mathematical techniques to perform 3D circular interpolation directly on the machines are deduced, although somewhat roundabout to overcome the programming limitations. The main programming limitations as well as cutter speed limitations are indicated.

OPSOMMING

Die beheereenhede van numeriesbeheerde vervaardigingsmasjiene beskik oor 'n beperkte aantal wiskundige funksies wat tot die beskikking van die programmeerder gestel word om programme mee te skryf. Selfs met hierdie beperkings is dit nou moontlik om programme te skryf waarmee driedimensionele (3D) sirkelinterpolasie direk op die masjiene uitgevoer kan word. Die nodige wiskundige tegnieke waarmee 3D sirkelinterpolasie gedoen kan word, al is dit 'n effens omslagtige manier om die beperkings te oorkom, word afgelei. Die belangrikste programmeringsbeperkings asook snyspoedbeperkings word aangetoon.

1. INTRODUCTION

No Computer Numerically Controlled (CNC) machine has the capability, as a function, to perform three dimensional (3D) circular interpolation. With the help of dedicated software such as CAD or CAM, a large number of points can be calculated on the circumference of a required 3D circle. These points can be joined together by short straight lines to create a 3D circular movement on a CNC machine. To create a curved 3D surface such as the surface of a bent pipe, many thousands or even millions of points could be necessary. These points can be drip fed into the memory of the CNC machine. CAD-CAM software is expensive and there are many applications where 3D surfaces can be programmed manually if 3D circular interpolation models were available.The drip feed option is only available on relatively modern machines.

Well-developed circular interpolation capabilities are available on CAD/CAM systems but the same mathematical techniques cannot be applied with manual programming because the controls of the CNC machines have only a very limited number of mathematical functions available.

Cutter radius correction on CNC machines can only be applied two dimensionally. For 3D surfaces on a milling machine the programmer must calculate the equidistant path of the centre of the ballnose cutter or, on some machines, specify the normal vector to the surface at each 3D point.

Three dimensional circular interpolation should not be confused with helical interpolation. Most modern CNC machines can perform helical interpolation but the movement of the cutter in this case is spiral and not circular.

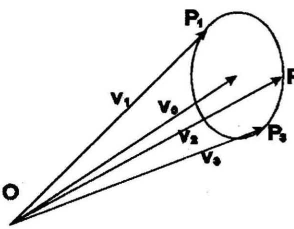
The aim of this paper is to provide the programmer with mathematical tools with which 3D circular interpolation models can be developed for direct execution on CNC machines. Depending on the type of 3D circle definition that is used, some 3D interpolation models are much slower to execute than others, resulting in ceilings on the maximum attainable cutting speed. Maximum cutting speeds were measured for each developed model on a MAHO 432 milling machine with a 386 processor. These speeds provide guidelines as to whether the specific model can be used in practical applications or not.

2. 3D CIRCLE DEFINITION: THREE POINTS ON THE CIRCUMFERENCE OF A CIRCLE

2.1 CLASSICAL METHOD

The mathematical equation of a 3D circle can be obtained from three points on the circumference of the 3D circle by solving three equations with three unknown quantities.

FIGURE 1: A 3D circle in space with known coordinates of three points (P_1,P_2,P_3).



Let V_1, V_2 and V_3 be the three vectors from the origin to P_1 ,P_2, P_3 and let V_o be the vector from the origin to the centre of the circle. The vectors V_4 and V_5 can be defined as

$$V_4 = V_2 - V_1$$
$$V_5 = V_3 - V_1$$

and the radius vector V_o can be defined as

$$V_o = aV_2 + bV_2 \text{ where}$$

a and b are constants, indicating a linear relationship.

If the radius of the 3D circle is R, then the following three equations can be defined.

$$| V_1 - V_o |^2 = R^2 \quad \ldots\ldots\ldots\ldots \quad (1)$$

$$| V_2 - V_o |^2 = R^2 \quad \ldots\ldots\ldots\ldots \quad (2)$$

$$| V_3 - V_o |^2 = R^2 \quad \ldots\ldots\ldots\ldots \quad (3)$$

Equation (1) can be rewritten as follows:

$$V_1 - V_o = V_1 - a(V_2 - V_1) - b(V_3 - V_1)$$

$$= V_1(1 + a + b) - aV_2 - bV_3 \quad \ldots\ldots\ldots \quad (4)$$

Substituting the coordinates (x_1, y_1, z_1) of P_1 into equation (4), the following set of equations is obtained.

$$V_1 - V_2 = \begin{vmatrix} x_1 - a(x_2 - x_1) - b(x_3 - x_1) \\ y_1 - a(y_2 - y_1) - b(y_3 - y_1) \\ z_1 - a(z_2 - z_1) - b(z_3 - z_1) \end{vmatrix} \quad \ldots\ldots \quad (5)$$

In terms of the radius R, the first equation of the equation set (5) can be expressed

as:

$$\{x_1 - a(x_2 - x_1) - b(x_3 - x_1)\}^2 + \{y_1 - a(y_2 - y_1) - b(y_3 - y_1)\}^2 + \{z_1 - a(z_2 - z_1) - b(z_3 - z_1)\}^2 = R^2 \quad \ldots\ldots \quad (6)$$

Similarly, two more equations can be obtained by making use of equations (2) and (3). It is possible to solve the three equations simultaneously to obtain the three unknown entities (a, b and R).

The three equations of which equation (6) is the first one are nonlinear and a search routine (such as Newton-Raphson [6]) can be applied to obtain the unknown values.

Once the three unknown values are solved the centre of the circle is known from V_o.

A vector V from the centre of the circle to a point on the circumference can be expressed as:

$$V = V_o + \alpha V_1 + \beta V_2 \quad \ldots\ldots \quad (7)$$

where α and β are two variables.

The relationship between α and β is the following

$$| \alpha V_1 + \beta V_2 | = R \quad \ldots\ldots \quad (8)$$

From equation (8) it is possible to express β as a function of α and R, and thus equation (7) can be expressed as a function of only one variable namely α.

It should be noted that α is highly non-linear and will result in points on the circumference of the 3D circle which are not evenly spaced for a constant increase in α. Equation (7) can be used to calculate coordinates of points very fast on the circumference of 3D circles but any attempt to use it as an interpolation equation will result in flat areas on the circle circumference.

The mathematical tools to solve equations like (6) are not available on the controls of CNC machines and therefore a different approach is needed. Because of these mathematical limitations, the speed and efficiency of solving spatial curve equations, as with CAD/CAM-models, cannot be compared with hand programming techniques.

2.2 ROTATION MATRIX APPROACH

In the case where three points on a 2D circle are defined, it is possible to calculate any number of points on the circumference of the circle with which circular interpolation can be performed for the 2D circle. It is always possible to create any 3D circle from this 2D circle by two rotations and one translation, Lubbe [7].

Given three points on the circumference of a 3D circle, it is possible to perform the reverse process as described above, to create the interpolation points on a 2D circle. Thus, if the two angles can be calculated through which the 3D circle was initially rotated from one of the main planes of the Cartesian coordinate system, then the three points can be rotated in a reverse order to lie on the main plain where 2D circular interpolation can be performed. Once the coordinates of enough points on the 2D circle are known to perform 2D circular interpolation, all these points can be rotated and translated back to the 3D positions.
The following expression, Klafter [1], can be used for a combination of two successive rotations.

$$R(\theta_z,\theta_y) = [Rot(y,\theta_y)]\cdot[Rot(z,\theta_z)] \quad \ldots\ldots\ldots \quad (9)$$

where $R(\theta_z,\theta_y)$ is a combination rotation matrix with which a point can be rotated firstly through θ_z degrees about the z-axis and then through θ_y degrees about the y-axis.

Taking the dot product in equation (9) results in

$$R(\theta_z,\theta_y) = \begin{bmatrix} c\theta_y c\theta_z & -c\theta_y s\theta_z & s\theta_y & 0 \\ s\theta_z & c\theta_z & 0 & 0 \\ -s\theta_y c\theta_z & s\theta_y s\theta_z & c\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots \quad (10)$$

where $c\theta_y = \cos(\theta_y)$ and
$s\theta_z = \sin(\theta_z)$ etc.

$$A = R(\theta_z,\theta_y),\{Trans(a,b,c)\} = [Trans(a,b,c)]\cdot[R(\theta_z,\theta_y)] \quad \ldots\ldots\ldots\ldots \quad (11)$$

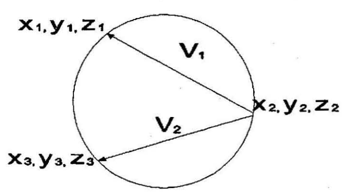
where A is a combination matrix for performing firstly the double rotation $R(\theta_z,\theta_y)$ and then the translation, Trans(a,b,c).

Taking the dot product in equation (11) results in

$$A = \begin{bmatrix} c\theta_y c\theta_z & -c\theta_y s\theta_z & s\theta_y & a \\ s\theta_z & c\theta_z & 0 & b \\ -s\theta_y c\theta_z & s\theta_y s\theta_z & c\theta_y & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots\ldots\ldots \quad (12)$$

Sometimes one is forced to use absolute as well as incremental programming in a single CNC program. For this purpose it should be pointed out that if matrix A is premultiplied by another matrix B, it has the effect of performing B on the original axes system whereas postmultiplying A by B, results in performing B on the new axes system. Premultiplication thus implies absolute programming whereas postmultiplication implies incremental programming.

FIGURE 2: Three points on a 3-D circle.



Referring to Figure 2

$$V_1 = \begin{bmatrix} x_1-x_2 \\ y_1-y_2 \\ z_1-z_2 \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots \quad (13)$$

$$V^2 = \begin{bmatrix} x_3-x_2 \\ y_3-y_2 \\ z_3-z_2 \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (14)$$

where V_1 and V_2 are the vectors as shown in Figure 2 and x_1, x_2 and x_3 are the x-coordinates of the three points on the circle.

$$V_n \quad = \quad \begin{bmatrix} x_n \\ y_n \\ z_n \\ 1 \end{bmatrix} = \begin{bmatrix} (y_1-y_2)(z_3-z_2)-(z_1-z_2)(y_3-y_2) \\ (z_1-z_2)(x_3-x_2)-(x_1-x_2)(z_3-z_2) \\ (x_1-x_2)(y_3-y_2)-(y_1-y_2)(x_3-x_2) \\ 1 \end{bmatrix} \quad \ldots . (15)$$

where V_n is the normal vector on the circle obtained by calculating the cross product of V_1 and V_2 and x_n, y_n, and z_n are the components of the normal vector in the x, y and z directions.

If it is assumed that the 2D circle is initially in the y-z plane and then first rotated θ_z about the z-axis and then θ_y about the y-axis the two angles can be calculated from

$$\theta_z \quad = \quad \text{atan} \quad \left[y_n / \sqrt{(x_n^2 + z_n^2)} \right] \quad \ldots\ldots\ldots\ldots\ldots\ldots (16)$$

and $$\theta_y \quad = \quad \text{atan} \quad \left[-z_n / x_n \right] \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (17)$$

The three points (given in Figure 2) can now be rotated to the y-z plane by multiplying each point with the inverse of rotation matrix (10) .

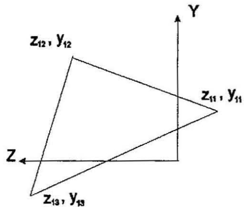
Vector V_n is a vector from point 2 and which is perpendicular to the 3D circle.

Points 1 and 3 can now be rotated back (ie $-\theta_y$ and $-\theta_z$) about the y- and z-axis with origin at point 2.

$$\text{Rot}(\theta_y,\theta_z) \quad = \quad \begin{bmatrix} c\theta_z c\theta_y & -s\theta_z & c\theta_z s\theta_y & 0 \\ c\theta_y s\theta_z & c\theta_z & s\theta_z s\theta_y & 0 \\ -s\theta_y & 0 & c\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots\ldots\ldots (18)$$

Applying equation (18) to points 1 and 3 results in three points on the circumference of a circle that is parallel to the y-z-plane with point 2 at coordinates (0, 0, 0).

FIGURE 3: Three points on a plane parallel to the y-z plane and obtained by back rotating from the 3D circle.



Referring to Figure 3, which depicts three points on the circumference of a 2D circle, the centre (y_c, z_c) can now be calculated.

In this situation, the perpendicular bisector of each chord of the circle passes through the centre. Since the circle is to pass through each of the three points, the lines joining them are chords. Two of the three perpendicular bisectors will suffice to locate the centre.

The following two equations describe the two chords (from second to first and second to third points) each of which passes through the centre (y_c, z_c).

$$(y_{11}-y_{12})\,y_c+(z_{11}-z_{12})\,z_c+\frac{y_{12}^2-y_{11}^2+z_{12}^2-z_{11}^2}{2}=0 \qquad \ldots\ldots\ldots (19)$$

and

$$(y_{13}-y_{12})\,y_c+(z_{13}-z_{12})\,z_c+\frac{y_{12}^2-y_{13}^2+z_{12}^2-z_{13}^2}{2}=0 \qquad \ldots\ldots\ldots (20)$$

$$K_1 = (y_{11}-y_{12})$$

$$K_2 = (z_{11}-z_{12})$$

$$K_3 = \frac{y_{12}^2-y_{11}^2+z_{12}^2-z_{11}^2}{2}$$

$$K_4 = (y_{13}-y_{12})$$

$$K_5 = (z_{13}-z_{12})$$

$$K_6 = \frac{y_{12}^2-y_{13}^2+z_{12}^2-z_{13}^2}{2}$$

where y_{11} is the y-coordinate of point 1 and y_{12} is the y-coordinate of point 2, etc.

Thus, equation (19) and (20) may be rewritten as

$$K_1 y_c + k_2 z_c + k_3 = 0$$

$$K_4 y_c + k_5 z_c + k_6 = 0$$

Solving for z and y

$$z_c = (k_1 k_6 - k_4 k_3) / (k_4 K_2 - K_1 k_5) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (21)$$

and

$$y_c = -(k_3 + k_2 z_c) / k_1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (22)$$

The radius (R) of the circle, which is measured from (y_c,z_c) to any of the three points in Figure 2, can be calculated from

$$R \quad = \quad \sqrt{[(y_{11} - y_c)^2 + (z_{11} - z_c)^2]} \quad \ldots\ldots\ldots\ldots\ldots (23)$$

It is important to note that y_c and z_c are incremental values measured from the three points on the y-z plane.

The centre point (a,b,c) of the 3D circle can now be calculated as

$$V_s = \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = \left[Trans(x_2, y_2, z_2) \right] \cdot \left[R(\theta_z, \theta_y) \right] \cdot \begin{bmatrix} 0 \\ y_c \\ z_c \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots (24)$$

The values a,b and c in vector V_s are the values needed in matrix A of equation (12).

Points on the circumference of the 2D circle can now be calculated in relation to its centre point, rotated around the centre point and then translated to the 3D circle position with Trans (a, b, c) as incorporated in matrix A.

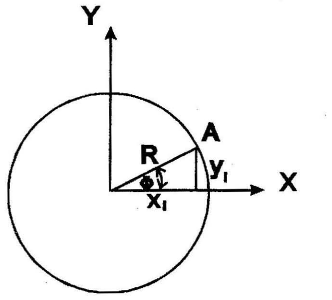


FIGURE 4: Calculation of points on a 2D circle.

X and y coordinates of points such as point A in Figure 4 can be calculated by increasing ϕ in very small steps

$$x_i = R\cos\phi \qquad (25)$$

and

$$y_i = R\sin\phi \qquad (26)$$

where R is the radius obtained from equation (23) and x_i and y_i are the x and y coordinates of points such as A on the circumference of the circle in Figure 4.

All these x and y values can now be transformed with the matrix in equation (12) to obtain points on the desired 3D circle. 3D circular interpolation is performed when all these points are joined together with short straight lines.

3. CNC PROGRAMMING CONSTRAINTS

Before attempting to write a CNC programme there are several constraints that the programmer must consider.

3.1 Only basic mathematical and trigonometric functions are available on CNC controls and no matrix algebra is possible. A very low level language is used for CNC programming, MAHO [3].

3.2 With CNC control units it is not possible to do double precision arithmetic so that large inaccuracies can arise when rounding-off errors accumulate with incremental programming.

3.3 The double-argument function ATAN2 that is usually available in most mathematical packages and which considers quadrant information defined by the signs of its arguments, is not available on CNC controls. Equations such as (16) and (17) do not give an indication regarding the quadrant in which a tilted circle lies.

3.4 Angles of 90° and 270° in equation (17) result in the divider becoming zero which is arithmetically illegal.

3.5 In CNC programmes a maximum of 37 characters are allowed in any equation and longer equations must be subdivided, MAHO [3].

3.6 For normal 2D circles there is a convention regarding clockwise and counterclockwise direction of machining but for 3D circles these directions have no meaning, MAHO [3].

3.7 For 3D circles there is no reference frame from which angles can be measured and thus there is no fixed position that suggests the position of zero degrees.

3.8 When programming for CNC milling machines the cutters can be mounted perpendicular to either the XY-plane, the XZ-plane or the YZ-plane, Horath [2], and the programme must make provision for all these cases.

3.9 One very seldom comes across single 3D circles in practical machining because these circles are invariably part of 3D surfaces and for machining such surfaces, provision must be made to cut the circles in a forward and backward direction. To accomplish this necessitates the keeping of tallies of odd and even repetitions in order to fit in the correct number of cuts onto a given surface.

3.10 Optimal calculation time is normally achieved by removing all constant calculations from repetitive cycles. In programmes for laser machines the pile up of calculations outside a cycle, but between circles, can cause the laser to burn a hole while the beam dwells for the time the control takes to finish calculations. In such cases the calculations should be distributed evenly on the inside and the outside of cycles.

3.11 Cutter radius compensation must be considered by the programmer and this alone could be a major task considering that normal vectors to a surface are needed for proper placement of the cutter relative to the surface.

4. COMMENTS ON THE CNC PROGRAMME

A CNC programme was developed for the MAHO-432 control, MAHO [3], by making use of the mathematical equations developed in section 1.2. Various verification techniques were followed to ensure correct results. The following observations were made.

4.1 With the normal 2D circular interpolation function available on the control a cutting speed of up to 5 m/min is possible. A maximum of about 400 mm/min cutting speed can be obtained for the same circle when circular interpolation is performed making use of SINE and COSINE functions. The main reason for this low execution time is that the control of the CNC machine must wait for the cutter to arrive at the endpoint of an interpolating line before the next calculation can be performed. An extremely fast search procedure is used for normal 2D interpolation but the mathematical tools to programme such a procedure are not at the disposal of the CNC programmer.

4.2 To compare execution times of various 3D interpolating techniques a radius of 50 mm and interpolating angles of 0.5 of a degree were used as standard. The maximum cutting speed with this interpolation technique was about 200 mm/min. This cutting speed might seem very low in contrast with possible cutting speeds of up to 5 m/min with these machines, but such high speeds can only be reached with straight line cutting and very rigid tools. For 3D sculpturing and with a ball nose cutter (typically 15 mm diameter) cutting speeds of 250 mm/min in tool steel is very realistic.

4.3 Accuracy is limited to about 1/100 mm while CNC machines can normally cut up to an accuracy of 10μ. This accuracy can be increased if the interpolation angle is made smaller than 0.5 degrees, but then the maximum cutting speed is reduced.

5. OTHER 3D CIRCLE DEFINITIONS

With the same mathematical tools as described above, two more interpolation models were developed. The first additional one was for circular interpolation where the coordinates of two points on the circumference of the 3D circle as well as the radius are known. For these cases there are always two circles possible and the programmer must decide on a rule as to ensure that the correct circle is selected. A maximum cutting speed of about 185 mm/min was obtained with this particular model.

The second additional model required the coordinates of two points on the circumference of the 3D circle as well as the coordinates of the centre point. With this model it is not necessary to transform the circle in an inverse direction back to the YZ main plane, because the centre point of the 3D circle is already at a known point. Secondly, transformation of the centre point from the main plane to the 3D location is not necessary. A maximum cutting speed of about 230 mm/min was obtained by using this model to interpolate the circle.

6. CONCLUSIONS

With fast 386-processors on CNC machines now available it is possible to do practical 3D circular interpolation directly on the machines. It is also expected that manufacturers of CNC machines will in future add more mathematical functions and make more memory available to the programmer and thus provide the programmer with more computing power with which higher cutting speeds would be attained.

The technique of 3D circular interpolation can be applied widely in the field of CNC manufacturing and can fill an important gap in the programming of 3D surfaces for the plastic injection and foundry industries. This technique will eliminate the purchase of very expensive CAD/CAM software for applications like milling the inside surface of a bent pipe cut open along the centre line.

7. REFERENCES

1 Klafter R D, Chmielewski T A and Negin M, "Robotic Engineering", Prentice-Hall International, 1989.

2 Horath Larry, "Computer Numerical Control: Programming of Machines", Macmillan Publishing Company, 1993.

3 MAHO Aktiengesellschaft, "Schulungsunterlagen", MAHO AG, Pfronten, 1994.

4 Thomas G B, "Calculus and Analytic Geometry", Addison-Wesley Publishing Co. Inc., 1962.

5 Foley J D, "Fundamentals of interactive Computer Graphics", Addison Wesley Publishing Co. Inc., 1984.

6 Thomopoulus N T, "Applied Forecasting Methods", Prentice-Hall International, 1980.

7 Lubbe A J, "Praktiese Driedimensionele sirkelinterpolasie vir Numeriesbeheerde Masjiene", S A Journal of Industrial Engineering, Vol 5, No 2, December 1992, pp. 66-80.