

Y. Koren
O. Masory

Faculty of Mechanical Engineering,
Technion—Israel Institute of Technology,
Haifa, Israel

Reference-Pulse Circular Interpolators for CNC Systems

Interpolation techniques for CNC manufacturing systems are either of the Reference-Pulse or Reference-Word type. Reference-Pulse interpolations emit a sequence of pulses as reference signals to the control loops, whereas Reference-Word interpolators provide binary words as references to Sampled-Data control loops. In the present paper, a number of Reference-Pulse interpolation methods are described, all of which function in an on-line iterative mode controlled by an external interrupt source. These methods are compared in terms of accuracy, maximum radius, uniformity of feedrate along the circular path, and maximum feedrate attainable. Selection of the appropriate interpolation method is found to depend on the specific application. Reference-Word interpolators will be covered in a subsequent paper.

Introduction

Computerized numerical control (CNC) is attracting increasing attention for manufacturing. With CNC, a minicomputer is provided as part of the controller to perform the basic numerical control functions. The low prices and impressive capabilities of the current minicomputers and microprocessors are naturally contributing to the increasing use of CNC over a broad spectrum of manufacturing systems, including virtually all types of machine tools, laser-beam cutters, welders, EDM and ECM machines.

From a control point of view, the significant common requirement of all CNC systems is to generate coordinated movement of the separately driven axes-of-motion to achieve a desired path of the "tool" relative to the workpiece. This involves both the generation of reference signals prescribing the shape of the produced part and the control of the machine to follow these reference signals. Generation of the reference signals is accomplished by software within the computer, and the control loops are constructed mainly by hardware.

Reference signals from the computer can be transmitted either as a sequence of reference pulses (Reference-Pulse System), or as a binary word in a Sampled-Data System [1]. In the Reference-Pulse System, the computer produces a sequence of reference pulses for each axis-of-motion, each pulse generating a motion of one BLU.¹ The accumulated number of pulses represent position, and the pulse frequency is proportional to the axis velocity. These pulses can either actuate a stepping motor in an open-loop system (see Figure 1), or be fed as a reference to a closed-loop system [3, 4, 5] (see Figure 2). With the Sampled-Data technique, the control loop of each axis is closed through the computer itself, which generates reference binary words.

¹ Dimensions in NC systems are expressed by integers in units corresponding to the position resolution of each axis-of-motion [2]. This unit is referred to henceforth as the "basic length unit" (BLU), which might be typically 10 μm in a machine tool system.

Contributed by the Production Engineering Division and presented at the Winter Annual Meeting, Chicago, Ill., November 16-21, 1980 of the American Society of Mechanical Engineers. Manuscript received at ASME headquarters July 1980. Paper No. 80-WA/PROD-7.

Both of these CNC types require distinct interpolation routines in the control program to generate their corresponding reference signals: pulses or binary words.

The overall design of a CNC system requires firstly the selection of the appropriate control technique (Reference-Pulse or Sampled-Data) and the optimal setting of the control loop parameters. Subsequently, the appropriate interpolator routines must be written. The Reference-Pulse interpolators are simpler to program, but there is a restriction on the maximum axis velocity imposed by the interpolation execution time. Therefore, this method is unsuitable for manufacturing systems requiring high axial velocities. By contrast, the maximum velocity in Sampled-Data systems is not limited by the computer, but the interpolator routine is more complex and the control-loop gain is lower, which consequently causes bigger axial position errors [4].

The machining path is usually built up from a combination of linear and circular segments, and accordingly, both a linear and a circular interpolation routine are contained in the control programs. These interpolator routines must be written in assembly language in order to conserve memory volume and improve computing speed, since the maximum attainable feedrate is proportional to the computing speed of the interpolator. Although linear interpolators can be easily written by assembly language, the difficulties arise in assembly programming of the circular interpolators which necessitates on-line solution of second-order equations.

The present paper deals with circular interpolators for the Reference-Pulse technique; interpolators providing reference binary words for Sampled-Data systems will be considered in another paper to follow.

The Concept of Reference-Pulse Interpolators

Reference-Pulse Interpolation generates a sequence of output pulses representing position and velocity. Each axis-of-motion is controlled by two pulsed lines, one for CW (clockwise) and the other for CCW (counter-clockwise) motions. Since each pulse is equivalent to one basic length-unit (BLU), the axis velocity is proportional to the pulse frequency and its position to the number of transmitted pulses. For generating a circular arc, the pulse frequency is propor-

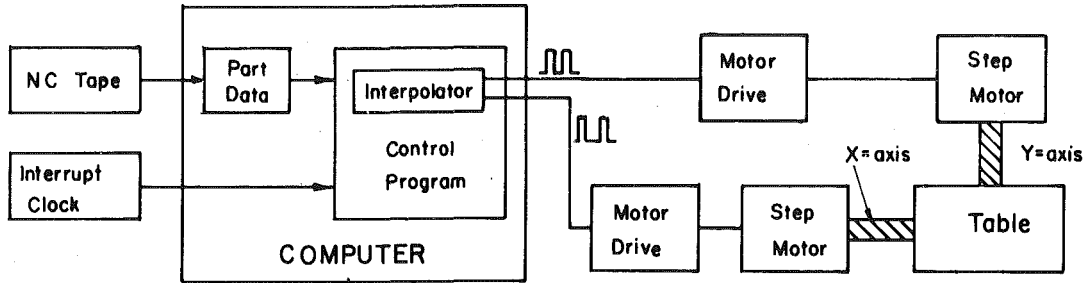


Fig. 1 Open-loop CNC system

tional to $\sin(\omega t)$ and $\cos(\omega t)$ for the X-axis and Y-axis, respectively.

All Reference-Pulse interpolations are based upon an iterative technique controlled by an adjustable interrupt clock. At each interrupt, a single iteration of the routine is executed, which in turn can provide an output pulse. The relationship between the required feedrate V (the velocity of the X-Y table) and the interrupt frequency F in circular motions is given by:

$$F = \frac{N}{L} V \quad (1)$$

where L is the arc length in BLU's, N is the number of iterations (or interrupt pulses) to complete the arc, and V is given in BLU's/sec. The maximum feed-rate is proportional to the maximum attainable interrupt rate, which in turn depends on the execution time of the algorithm.

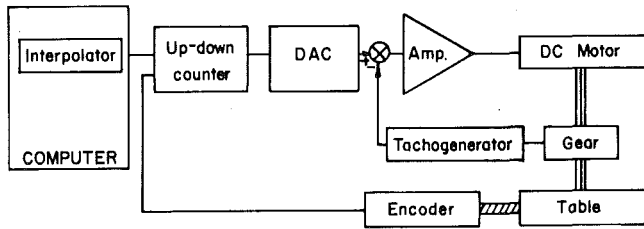


Fig. 2 Single-axis closed loop system

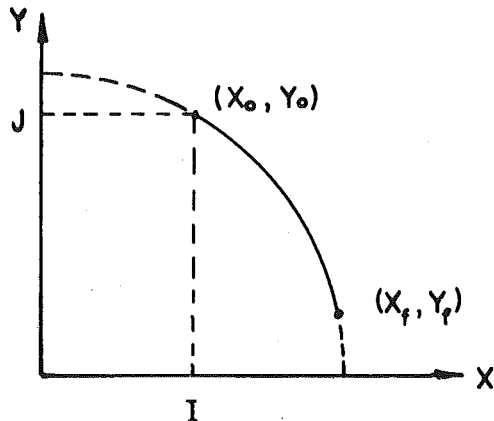


Fig. 3 Initial conditions of a circle

Three Reference Pulse Methods will be described and compared:

- (1) Software DDA method;
- (2) Stairs Approximation method;
- (3) Direct Search method

It will be shown that each of these methods has a typical N/L value and therefore requires a different setting of F .

Software DDA Method

The digital differential analyzer (DDA) method is an interpolation procedure which was developed for use in hardware interpolators (3, 6, 7]. The method requires successive additions in order to create new interpolated points, and is therefore ideally suited to assembly language simulation. The rate of additions in the hardware interpolator is controlled by a high frequency clock. In order to adapt the clock frequency to a permitted interrupt rate, the software interpolator requires some modification, essentially in the definition of register length [8]. The register length is adapted to each segment of the workpiece, which for a circular arc is equal to the radius R .

The flow-chart of the algorithm for generating the circle in Fig. 3 is given in Fig. 4. It can be seen that the DDA algorithm requires only

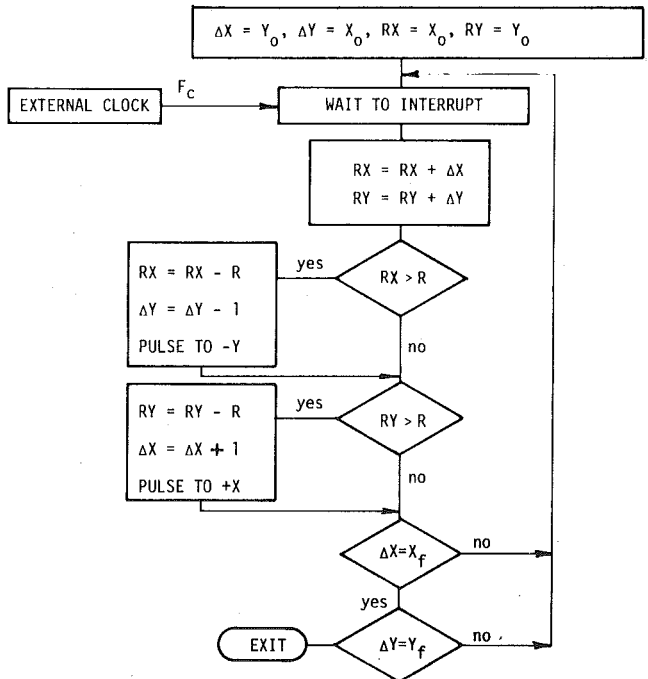


Fig. 4 Flowchart of software DDA method

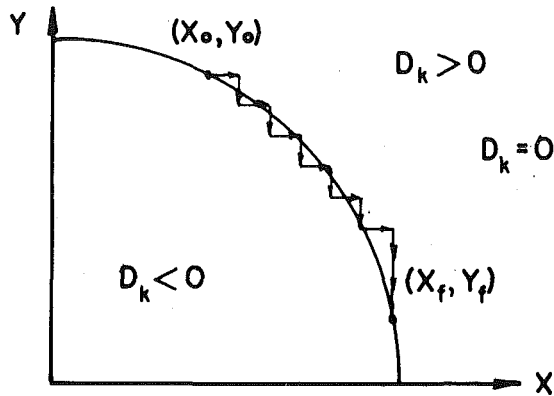


Fig. 5 Circle generated by the Stairs Approximation method

basic assembler instructions: addition, subtraction, comparison, etc. However, the comparison of the contents of RX or RY , with the radius R can cause some difficulty. The radius is not given explicitly in the NC format [2] and must be calculated from the initial values of X_0 and Y_0 (denoted as I and J in reference [2]). Employing the exact formula requires the use of a Floating Point Unit (FPU) to calculate the radius from:

$$R = \sqrt{X_0^2 + Y_0^2}$$

In order to eliminate the necessity of a FPU, R might be determined by the approximation

$$R = \max(X_0, Y_0) + \frac{1}{2} \min(X_0, Y_0) \quad (2)$$

which is derived by assembly language. The implementation of this approximation results in an error of less than 11%. This error only decreases the feedrate without affecting the accuracy of the algorithm. Alternatively, the radius can be added explicitly to the part-data program on the perforated tape. However, the circle accuracy is slightly affected by the initial conditions of RX and RY . Note that inserting the initial values as $RX = X_0$ and $RY = Y_0$ is preferred to setting the registers initially to zero.

As is seen from Fig. 4, the routine includes a comparison of the contents of RX and RY to R . In the worst case, RX or RY is equal to $(2R - 1)$. Therefore, in an n -bit computer with one bit reserved for the sign, the maximum allowable radius is:

$$R_{\max} = 2^{n-2} - 1 \quad (3)$$

The number of iterations required to complete a quarter of a circle with a software DDA is given by [8]:

$$N = (\pi/2)R \quad (4)$$

Since the number of iterations is identical to the length of the arc in BLU's, the interrupt pulse frequency F must be equal to the required velocity V given in BLU's/sec (see equation (1)). This feature is especially useful for providing acceleration or deceleration along the axis-of-motion.

The DDA method has been applied in a CNC system using an HP-2100 computer [8, 9]. The circular interpolator routine occupied 40 words of 16-bit, with a maximum execution time of 74 μ sec per iteration. This enables a maximum feedrate of approximately 13000 BLU's/sec (8 m/min with a BLU = 0.01 mm) when using the HP-2100 as the CNC computer.

Stairs Approximation Method

The Stairs Approximation Method (known also as the Incremental Interpolator) is based on proceeding along the circular arc by BLU step increments either in the X - or in the Y -direction [7, 10], as shown

in Fig. 5. To illustrate its operation, assume that upon completing k iterations the point (X_k, Y_k) is reached. The algorithm requires the determination of a variable D_k given by

$$D_k = X_k^2 + Y_k^2 - R^2 \quad (5)$$

The stepping policy to be used depends on the sign of D_k , as well as on the required direction of motion (CW or CCW) and the quadrant of the circle. For example, consider a CW motion in the first quadrant. The Stairs Approximation method is executed as follows:

- If $D_k < 0$ (the point is within the circle), one step in the $+X$ direction is performed.
- If $D_k > 0$ (the point is outside the circle), one step in the $-Y$ direction is performed.

The case $D_k = 0$ is included in one of the above cases. Upon completing a step, the value of X or Y is updated, and the algorithm proceeds until reaching the final point (X_f, Y_f) . The stepping policy for the eight possible cases (four quadrants and two directions) is given in Table 1.

The stepping algorithm contains a small number of simple arithmetic operations, with a small storage requirement. The number of iterations to be executed, which is relatively large, is given by

$$N = |X_0 - X_f| + |Y_0 - Y_f| \quad (6)$$

where (X_0, Y_0) are the coordinates of the initial point. Therefore, the number of iterations required to generate a quarter of a circle is $N = 2R$, where R is given in BLU's. In order to obtain the required velocity V in BLU's/sec, the iterations are executed at a frequency F calculated by equation (1):

$$F = \frac{2R}{\pi R/2} = \frac{4V}{\pi} \quad (7)$$

The factor π in equation (7) causes some computational difficulties.

Although the algorithm is simple, the execution of equation (5) requires a multiplier, which may be a disadvantage. Moreover, the computer word must contain the value of R^2 , which in practice means operating with double-precision, or even triple-precision. The maximum permitted radius with double-precision is given by:

$$R_{\max} = \sqrt{2^{2n-1} - 1} \approx \sqrt{2} 2^{n-1} \quad (8)$$

In order to offset these unattractive features, improvements of the algorithm were developed [11, 12, 13] which are shown here for one case from Table 1 (CW direction, first quadrant). By adding one more index, equation (5) for the Improved Stairs method can be rewritten as:

$$D_{i,j} = X_i^2 + Y_j^2 - R^2 \quad (9)$$

where i and j are the number of steps accomplished in the X - and Y -direction, respectively. Upon executing an additional step in the X -direction, equation (9) becomes

$$D_{i+1,j} = (X_i + 1)^2 + Y_j^2 - R^2 = D_{i,j} + 2X_i + 1 \quad (10)$$

Table 1 Stepping policies in generating circles

No.	Quadrant	Direction	D<0	D>0
1	1	CW	+X	-Y
2	1	CCW	+Y	-X
3	2	CW	+Y	+X
4	2	CCW	-X	-Y
5	3	CW	-X	+Y
6	3	CCW	-Y	+X
7	4	CW	-Y	-X
8	4	CCW	+X	+Y

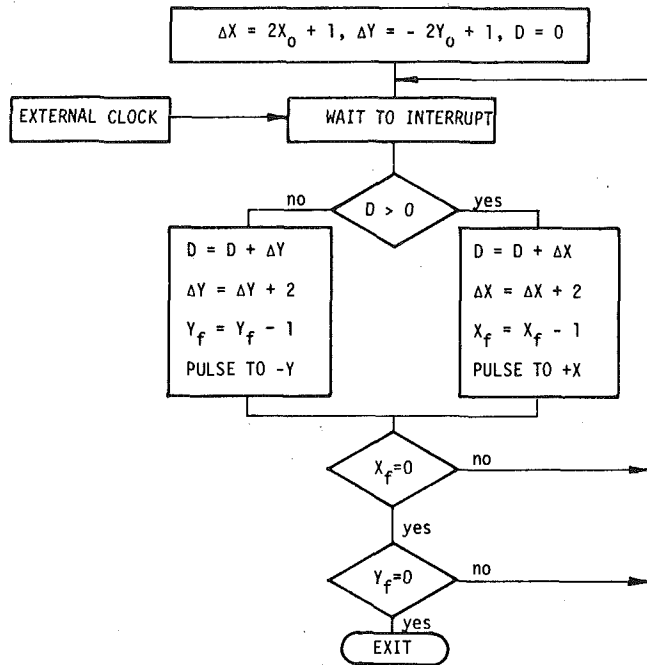


Fig. 6 Flowchart of the Stairs Approximation method

Defining

$$\Delta X_i = 2X_i + 1 \quad (11)$$

Equation (10) now becomes

$$D_{i+1,j} = D_{ij} + \Delta X_i \quad (12)$$

In the next step in the X -direction, D is increased to

$$D_{i+2,j} = D_{i+1,j} + \Delta X_{i+1} \quad (13)$$

where

$$\Delta X_{i+1} = 2X_{i+1} + 1 = 2(X_i + 1) + 1 = \Delta X_i + 2 \quad (14)$$

Similarly, the iterative equations for steps in the Y -direction are

$$D_{i,j+1} = D_{i,j} + \Delta Y_j \quad (15)$$

$$\Delta Y_{j+1} = \Delta Y_j + 2 \quad (16)$$

where

$$\Delta Y_j = -2Y_j + 1 \quad (17)$$

The initial conditions for equations (14) and (16) are derived from (11) and (17):

$$\Delta X_o = 2X_o + 1 \quad (18a)$$

$$\Delta Y_o = -2Y_o + 1 \quad (18b)$$

To summarize, the Improved Stairs Method is executed as follows: at each iteration the determination of a single variable, D , is required. The new D is evaluated from the previous D and an increment ΔX or ΔY , which is calculated simply by an additional instruction according to equations (14) or (16). Since the multiplication is eliminated, the computation speed is increased. Furthermore, the maximum contents of a word are now $2R + 1$, rather than R^2 . Therefore, the maximum radius expressed in an n -bit word is

$$R_{\max} = 2^{n-2} - 1 \quad (19)$$

The accuracy obtained (1 BLU) as well as number of iterations and the interrupt clock frequency F_c given again by equations (6) and (7).

A flowchart of the Improved Stairs Method is given in Figure 6. On a PDP-8 computer with a double-precision word [13], it occupied 80 (12 bit) words and had an execution time of 60 μ sec per iteration.

Direct Search Method

One drawback of the Stairs Approximation Method and its various modifications is that the axes-of-motion do not move simultaneously, so that more iterations are required to produce the circle than with the DDA technique. Furthermore, the basis for selecting an interpolated point is not related to any error criteria. By contrast, the Direct Search Method (DSM) is an optimization policy which is based upon searching in three directions for the minimum path error [14, 15]. The DSM applies the calculation strategy of the Improved Stairs Method but permits simultaneous motion of both axes, which consequently results in an interpolated circle closer to the required one.

The DSM is based upon evaluating the term $D_{i,j}$ in equation (9), which is proportional to the radial error $E_{i,j}$ [14]:

$$D_{i,j} \approx (2R)E_{i,j} \quad (20)$$

Therefore, it is only necessary to test the value of $D_{i,j}$ rather than $E_{i,j}$.

The strategy at each step proceeds by evaluating the variable $D_{i,j}$, which is defined in equation (9), at three points. The choice of these points depends upon the quadrant and the direction-of-motion as given in Table 1. For the case of CW motion to produce an arc in the first quadrant, three trials are made to determine the direction of motion for the smallest error. In the first trial, a value of 1 BLU is added to the coordinate X_i and the corresponding value $D_{i,j}$ is calculated according to equation 10. With the second trial, 1 BLU is subtracted from Y_j (with $X = X_i$) and the corresponding $D_{i,j}$ is obtained. For the final trial, 1 BLU is added to and subtracted from X_i and Y_j , and a third value of $D_{i,j}$ is obtained. The three $|D_{i,j}|$ values are compared in order to find the one giving the least error. The actual stepping is then performed in the appropriate direction and $D_{i,j}$, ΔX and/or ΔY , are updated. The flowchart of the DSM for CW motion, in the first quadrant is shown in Fig. 7.

The DSM method is more accurate than the methods described above. The maximum error with this method is only $\frac{1}{2}$ BLU. The number of iterations to produce a quarter of a circle is $N = \sqrt{2R}$ which is approximately 30% less than with the Stairs Approximation Method and 10% less than with the DDA Method. The maximum radius which can be obtained is the same as with the Stairs method and is given by equation (19). The interrupt frequency is calculated from equation (1):

$$F = \frac{2\sqrt{2}}{\pi} V \quad (21)$$

Improved Direct Search Method

The largest value of DX or DY , according to equations (11) and (17) is $2R + 1$, which limits the maximum permitted radius with the DSM method to the value given by equation (19). In order to increase the allowable radius, equations (11) and (17) are divided by 2, resulting in new definitions for ΔX and ΔY :

$$\Delta X_i = X_i; \quad \Delta Y_j = Y_j \quad (22)$$

(The residue of $\frac{1}{2}$ cannot be expressed when assembly language is used.) As a consequence, equations (14) and (16) become:

$$\Delta X_{i+1} = \Delta X_i + 1 \quad (23a)$$

$$\Delta Y_{j+1} = \Delta Y_j + 1 \quad (23b)$$

The maximum value of the new ΔX and ΔY is only R . Therefore, the maximum radius which can be expressed by an n -bit word is:

$$R_{\max} = 2^{n-1} - 1 \quad (24)$$

In order to avoid the accumulation of the residue $\frac{1}{2}$ which has been neglected in equation (22) at each step, the following policy is used:

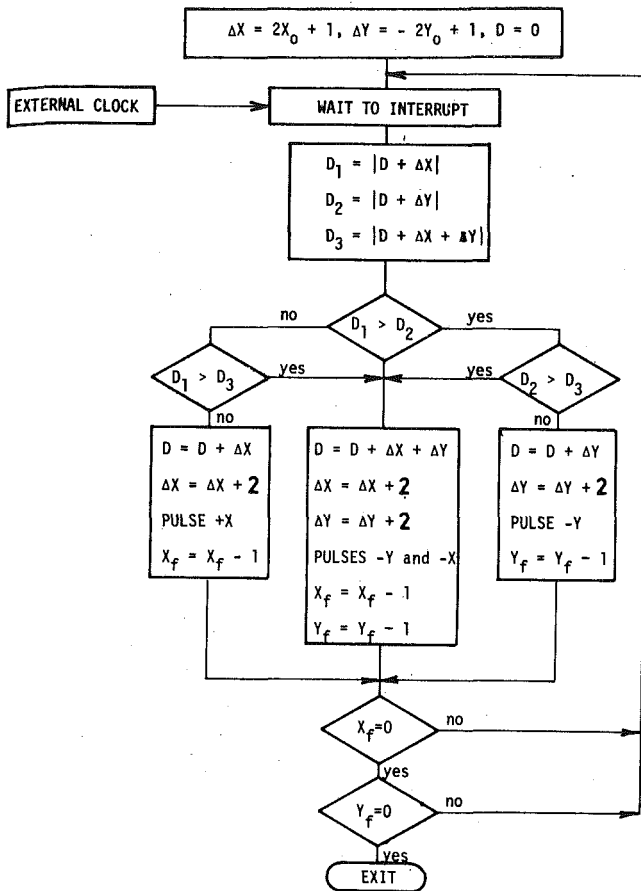


Fig. 7 Flowchart of the Direct Search method

- (a) If a simultaneous motion of both axes is carried out, add 1 to $D_{i,j}$.
 (b) If a single axis motion is performed, add 1 to $D_{i,j}$ every second iteration in which the policy is employed.

As a result of this policy the maximum radial error remains $\frac{1}{2}$ BLU. The number of iterations required to produce a quarter of a circle and the interrupt frequency F also remain as with the Direct Search Method.

Selection of Interpolation Method

Four Reference-Pulse interpolation methods for CNC systems have been described. In this section these methods are compared in order to provide a basis for selecting the appropriate interpolation method for a particular application.

Some features of the various algorithms are summarized in Table 2. It can be seen that the maximum radius that can be obtained with the Improved DSM is about twice as big as with the other methods with a maximum radial error of only $\frac{1}{2}$ BLU. However this feature is significant only with single-word programming (16-bit word) and does not impose a practical limitation on the system performance in other cases. For example, a system with a resolution of BLU = 0.01 mm using the Improved DSM on a 16-bit minicomputer permits a maximum radius of $0.01 \times 2^{15} \approx 328$ mm, while with the other methods the maximum radius is only 164 mm. If however a double-word technique ($n = 32$) is used, the allowable radii are 2×10^7 mm and 10^7 mm, respectively, which greatly exceed practical requirements.

In many manufacturing systems the overriding factor in selecting the interpolation method may be the uniformity of feedrate along the circular path. In a laser-beam cutter, for example, the feedrate prescribes the kerf (width-of-cut) and consequently, feedrate variations along the path also affect the accuracy of the produced part. The

Table 2 Comparison of reference pulses interpolators

Method	$R_{\max+1}$	E_{\max}	$\#$	\bar{V}	V_{\max}	V_{\min}
DDA	2^{n-2}	1	$\frac{\pi}{2}R$	F	F	F
Stairs	2^{n-2}	1	2R	$(\pi/4)F$	F	$(1/\sqrt{2})F$
DSM	2^{n-2}	$\frac{1}{2}$	$\sqrt{2}R$	$(\pi/2\sqrt{2})F$	$\sqrt{2}F$	F
Improved DSM	2^{n-1}	$\frac{1}{2}$	$\sqrt{2}R$	$(\pi/2\sqrt{2})F$	$\sqrt{2}F$	F

feedrate along the path is obtained from equation (1):

$$V = \frac{L}{N} F \quad (25)$$

where L is the length of the arc α given by $L = \alpha R$, and N is the number of iterations to complete the arc α and is given in Table 2 for a quarter of a circle. The average feedrate \bar{V} for a quarter circle has been calculated for the various methods and is also given in Table 2.

The DDA Method is based upon the solution of differential equations, so the feedrate along the path is expected to be constant. With the other methods, however, big variations can occur along the circular path. The biggest difference in the feedrate is between the circular segments in which $X \approx Y$ and $X \approx 0$; $Y \approx R$ (or alternatively, $X = R$; $Y = 0$).

In the neighborhood of ($X \approx 0$; $Y \approx R$) both the Stairs and DSM methods will produce at each iteration a pulse in the X-direction only, with the length of the path (in BLU's) equal to the number of iterations. In this case, according to equation (25), $V = F$. In percentage this feedrate for a quarter circle is 27% greater than the average in the Stairs method and 10% smaller than the average in the Direct Search Method. On the contrary, in the neighborhood of $X = Y$ the Stairs method will produce a pure "staircase" shape (see Fig. 5), namely, a pulse in X, then a pulse in Y, etc. As a consequence, the length of the corresponding arc (in BLU's) is $L \approx N/\sqrt{2}$, and according to equation (25) a feedrate of $V = F/\sqrt{2}$ is generated, which is the minimum and 10% below the average. By contrast, in the same region the DSM generates its maximum feedrate. At $X \approx Y$ this method produces at each iteration a pulse in X and a pulse in Y simultaneously, and therefore the generated path is $\sqrt{2}$ BLU per iteration and the maximum feedrate is $\sqrt{2}F$, which is 27% greater than the average. These results are also included in Table 2.

The maximum feedrate which can be achieved by a system may be another factor in selecting an interpolation method. For example, a digital plotter requires fast interpolation while the uniform velocity may not be required. The maximum allowable feedrate \bar{V}_{\max} is proportional to the maximum interrupt frequency F which can be provided. This frequency depends upon the computing speed of the algorithm, or

$$F_{\max} = 1/T_i \quad (26)$$

where T_i is the time required to execute one iteration. This time depends on the interpolation method and on the computing speed of the CNC computer.

In order to compare the computing speed of the various methods, they were programmed in assembly language by the same programmer on a PDP-11/40 minicomputer. The results are shown in Table 3. The iteration time is calculated by totaling the execution times of the individual instructions in the interpolation routine. Multiplying the iteration time by the number of iterations required to complete a quarter of a circle gives the quarter-circle execution time which is inversely proportional to the maximum obtainable feedrate in the system. In order to double-check the results, the total time duration of the generated pulses for a quarter circle with different radii was measured directly. Both tests were in agreement.

Table 3 Comparison on a PDP—11

Method	Iteration time [μsec]	Max. average feedrate [BLU/sec]	No. of iterations N	1/4 Circle exec. time [μsec]	Program length [Words]
DDA	52	19230	1/4πR	81R	46
Stairs	25	31420	2R	50R	30
DSM	58	19150	√2R	82R	67
Improved DSM	73	15220	√2R	103R	75

The maximum feedrates are calculated from \bar{V} in Table 2 and equation (26). It is seen that the maximum feedrate can be achieved with the Stairs Method, while the Improved DSM gives the slowest feedrate. It is worthwhile to note that the DDA method provides a maximum feedrate which is reasonable for most of today's manufacturing systems. For a BLU = 0.01 mm/pulse, the maximum feedrate with a PDP-11 is: $19230 \times 0.01 \times 60 = 11538$ mm/min. (The actual feedrate is slightly smaller since a feedrate-subroutine is also contained in the CNC control program.) In addition, Table 3 provides the number of computer words (of 16-bit) required by each method. As expected, all methods occupy a small number of memory locations, and so the storage requirement can not be considered as an evaluation criterion.

One measure of the quality of the circle is the maximum error in the radius E_{max} . The circle quality can also be expressed by the standard deviation S, defined by:

$$S = \sqrt{\frac{1}{N} \sum_{i=1}^N E_i^2} \quad (27)$$

which is a measure of the smoothness of the generated circle. In Table 4 it can be seen that the Direct Search Methods gave the best circle.

The four interpolator methods were simulated in producing a quarter of a circle with a radius of $R = 10000$ BLU's. The results, presented in Table 4, are in agreement with the theoretical ones.

In summary, the selection of the Reference-Pulse interpolator depends on the requirements of the system as follows:

1. A uniformity of the feedrate along a circular path is obtained only with the DDA method. (Note that all the methods discussed here provide uniform feedrate in linear motions.)
2. If the maximum feedrate is the only dominant factor in the system, the Stairs Method should be used.
3. The Improved Direct Search Method permits radii twice as big as with the other methods with the same computer word-size. The radial errors with this method are half as big with the other methods and also the best smoothness is obtained. Therefore, the Improved

Table 4 Results with R = 10000 BLU's

Method	N	E_{max}	S	V_{max}/V_{min}
DDA	15708	1.00	0.379	1.02
Stair	20000	1.00	0.472	1.40
DSM	14142	0.50	0.261	1.41
Improved DSM	14142	0.50	0.261	1.41

DSM is recommended for manufacturing systems requiring large radii with single-word programming, accuracy and smoothness.

Acknowledgment

The authors would like to thank Professor S. Malkin for his detailed comments and helpful suggestions.

References

- 1 Koren, Y., "Computer Control of NC Systems," *Spectrum*, Vol. 14, No. 3, Mar. 1977.
- 2 EIA Standard RS-274-B: Interchangeable Perforated Tape Variable Block Format for Contouring and Contouring/Positioning NC Machine Tools. May 1967.
- 3 Koren, Y., and Ben-Uri, J., *Numerical Control of Machine Tools*. Kahan Publishers, Delhi, India, 1978.
- 4 Koren, Y., "Design of Computer Control for Manufacturing Systems," *ASME-78-WA/PROD-14*, Dec. 1978.
- 5 Poo, A. N., and Bollinger, J. B., Digital-Analog Servo System Design for CNC. *IEEE-IAS, 9th Annual Meeting Proceedings*, Oct. 1974, pp. 621-630.
- 6 Koren, Y., Shani, A., and Ben-Uri, J., "Numerical Control of a Lathe," *IEEE Ind. and Gen. Appl.*, Vol. 6, No. 2, pp. 175-179, March 1970.
- 7 Danielson, P. E., "Incremental Curve Generation," *IEEE Trans. on Computers*, Vol. C-19, No. 9, Sept. 1970, pp. 783-793.
- 8 Koren, Y., Interpolator for a CNC system. *IEEE Trans. on Computers*, Vol. 25, No. 1, Jan. 1976, pp. 32-37.
- 9 Koren, Y., Design Concepts of a CNC system. *IEEE Ind. Appl. Soc., Annual Meeting, Atlanta, Proceedings*, Oct. 1975, pp. 275-282.
- 10 Kaiwa, T., and Inaba, S., "Latest Japanese Numerical Control Features," *Control Engineering*, Oct. 1961, pp. 88-91.
- 11 Gorman, J. E., and Raamot, J., "Integer Arithmetic Technique for Digital Control Computers," *Computer Design*, July 1970, pp. 51-57.
- 12 Sudarshan, P. A., "Electronic Techniques for NC Contour Machining," *6th MTDR Conference*, Dec. 1973.
- 13 Angelov, A. S., Mihailov, A. D., and Nachev, G. N., "Simple Algorithm for Curvilinear Interpolation," *11th Annual Meeting and Technical Conference Numerical Control Society*, Proceedings, March 31-April 3, 1974, pp. 338-348.
- 14 Masory, O., and Koren, Y., "The Direct Search Method in CNC Interpolators," *ASME-78-WA/PROD-40*, Dec. 1978.
- 15 Jordan, B. W., Lenon, W. J., and Holm, B. D., "An Improved Algorithm for Generation of Nonparametric Curves," *IEEE Trans. on Computers*, Vol. C-22, No. 12, Dec. 1973, pp. 1052-1060.