INCREMENTAL VERSUS BEST LINE CURVE APPROXIMATION APPROACHES APPLIED TO THE GENERATION OF CNC TOOL PATH [+]

Wissam K. Hamdan* **Hussam L. Alwan*** **Baraa M. Hussein***

Abstract:

Minimization of machining time is one of the most important tasks in the modern manufacturing technology. This paper aims to compare between two different types of curves approximation techniques namely Best Line Curve Approximation (BLCA) technique and Incremental Line Curve Approximation (ILCA) technique. These two approaches are based on approximation of any curve segment by a sequence of linear segments. Throughout building robust mathematical model via computer iteration programs ILCA technique and using the simple iteration method as a numerical approximation tool to achieve these tasks. The "Best Line" is the tangent line that touches the maximum number of points, while the incremental line is a chord line that intersects the interior curve throughout a small interval so that the produced deviation is still within the required tolerance. The proposed approaches are applied to numerical example for the purpose of maneuverability. The results show that BLCA approach requires further modification to control the tool path, but it gives a wide reduction in the length of tool path compared with ILCA approach which tends to maximize the tool path length but it requires a very simple modification to control the tool.

المستخلص

من الاولويات المهمة في عمليات التصنيع الحديثة هي تقليل زمن التشغيل. يهدف هذا البحث الى الموازنة بين طريقتين مختلفتين من طرق تقريب المنحنيات وتحويلها الى مجموعة من الخطوط المستقيمة وهي طريقة تقريب المنحني بأفضل خط مستقيم (BLCA)، وطريقة تقريب المنحني بمستقيمات تزايدية (ILCA). الطريقتان كلتاهما تعتمدان على تحويل المنحني الى مجموعة من الخطوط المستقيمة الصغيرة. من خلال بناء نموذج رياضي واستخدام البرامج الحاسوبية التكرارية بالنسبة لتقنية (ILCA) واستخدام طريقة التكرار البسيط كأداة للتحليل الرياضي العددي بالنسبة لتقنية (BLCA).أفضل مستقيم (Bes Line) هو المستقيم المماس والذي يمس المنحني بأكثر عدد ممكن من النقاط بينما المستقيم التزايدي هو قطعة المستقيم التي تقطع المنحني من الداخل الفترة صغيرة.ولغرض تقييم الطريقتين تم تطبيقهما على احد الامثلة العملية ، وقد أثبتت النتائج ان تقنية (BLCA) تحتاج الى عمليات تعديل اضافية للسيطرة على حركة العدة ضمن المسار الصحيح ولكن طول مسار العدة الناتج من هذه الطريقة هو اقصر بكثير من طول مسار العدة الناتج من تطبيق تقنية (ILCA) ، من ناحية أخرى فأن التقنية الاخيرة تحتاج الى عملية تعديل بسيطة جداً للسيطرة على حركة العدة.

[+]Received on 5/4/2005 , Accepted on 13/9/2005
* Department of Production Engineering and Metallurgy / University of Technology,

Keywords: curve approximation, simple iteration, length of tool path, chordal deviation.

Nomenclature:

Δx	Increment of x-axis domain.
P_{ML}	Mid point of linear segment.
P_{MC}	Mid point of curved segment.
e	Chordal deviation.
T	Tolerance.
x_{PML}	X-coordinate of P_{ML}.
y_{PML}	Y-coordinate of P_{ML}.
m	Slope of normal line.
c	Constant.
c_1,c_2,c_3	Constants for curve equation.
φ	$(c_2-m)/c_1$
λ	$(c_3-c)/c_1$
N	Number of intervals.
x_1	Start point of the curve.
x_f	End point of the curve.
i	Independent variable (i=1,2,3,4,…,n+1)
ε	Independent variable (ε=i-1)
x_i, y_i	Independent X&Y coordinates respectively.
$f'(x)$	First derivative of the curve.
y_t	Equation of tangent line.
y_n	Equation of normal line.
y_s	Equation of secant line.
G	$G = \left(\frac{c_2}{c_1} - \frac{f'(x_1)}{c_1}\right) c_2/c_1$
ψ	$\Psi = \frac{c_3 + f'(x_1)x_2^1 + y_2^1}{c_1}$
m	Independent variable & m≥2
n_p	Number of fixed points.

1. Introduction:

Curve and surface construction is an important topic in computer graphics, computer-aided design geometric modeling and visualization courses [1].

In CAD systems, tool paths for geometry, which is not defined by linear or circular segments, are always generated by breaking down this geometry into linear or circular segments. Breaking down process of these geometries is termed decomposition process. As illustrated in Figure (1), the decomposition process of such curves results in some deviation between the actual and piecewise line segments. This deviation is called tolerance between the actual curve and the approximated curve (see Figure 1) [2, 3].

As the tolerance approaches zero, the number of linear segments becomes infinite, resulting in large machining time, long machining program, which in turn result in increasing of cycle time and reducing the productivity. on the other hand, increasing tolerance will reduce the number of line segments and shortening the cycle

time, but this approach destroys the analytical continuity of the curve. Consequently, the most important factor affecting the machining time and productivity is the way of decomposition process. Fig.2 shows the relation between the number of linear segments and produced error [2].

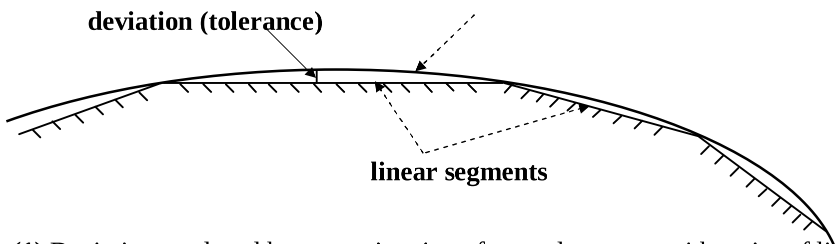


Figure (1) Deviation produced by approximation of curved segment with series of linear segments [3]

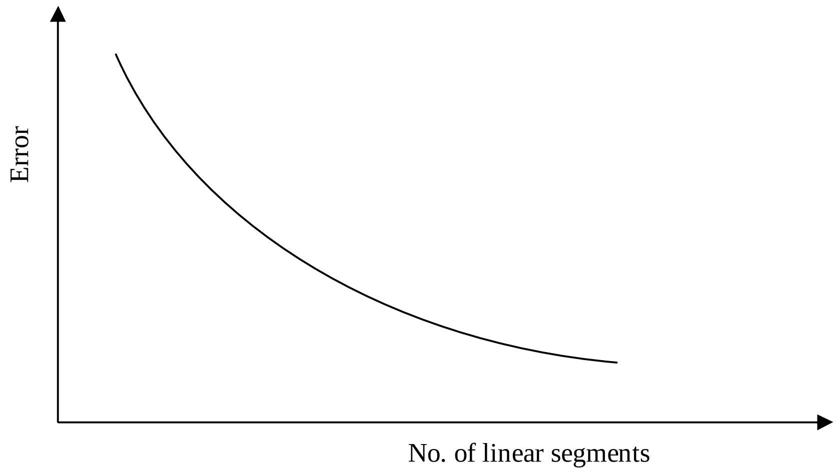


Figure (2) Relation between No. of linear segments and produced error, as the linear segments increases the produced error decreases [2].

From the open literature, many researches were interested in curve approximation and tool path generation. In Kim's algorithm [4], ball-end milling cutter is used to calculate the tool path for three dimensional space curve through approximation of these curves by circular arcs and the approximation of these arcs by linear segments.

A linear tool path generation for three dimensional space curves and surfaces using three different types of milling cutters was presented by Seuny Ryol et.al. [5], the tool path was generated using the Z-Map representation, by representing an object as a set of Z-axis aligned vectors.

Jachob's algorithm [6] discusses a simulation system for verifying 3-axis NC tool Path. The proposed simulation system consists of five stages, surface

converter, tool path generator, a generic APT-Like cutter model, an intermediate surface calculation and an animator/simulator system.

Susan X. Li and Robert B. [7] present an algorithm for non-isoparametric tool path generation of sculptured curves and surfaces throughout subdivision of the surface to train using ball end milling cutter.

Our research is focused on the investigation of two different methods of curve decomposition process, which can decrease the machining time. The first method is termed Incremental Line Curve Approximation (ILCA) and the second is called Best Line Curve Approximation (BLCA). The ILCA method depends on computer iteration program, but BLCA method depends on simple iteration method for linear approximation.

The paper is organized as follows. First, a description of the specification of the geometry of a planar contour. Then, describing the a description of our two methods for curve approximation throughout the generation of tool path for the two methods. Then a report on several machining parameters such as length of tool path, produced error and the specification of the two different methods is discussed. This paper is concluded with indicating the influence of these machining parameters on the machining time and productivity.

Curve Approximation Formulation:

In the following part the discussion on the curve approximation throughout the decomposition of a given curve into a definite series of linear segments is introduced. The decomposition procedure has a great importance in the CAM systems; therefore, it is very attractive for the process plnners to examine it along with studying its effect on the productivity. To increase productivity, the total machining time should be decreased; a perfect situation exists when the number of linear segments is minimized that is the produced error is exactly equal to the intended tolerance. Unfortunately, this situation rarely occurs. Therefore, the selection of decomposition procedure is an important issue in NC machining problems.

The discussion and analysies of two approximation approaches namely incremental and best line curve approximation are as follows.

Incremental Line Curve Approximation Procedure (ILCA):

This approach can be classified under quick and dirty approach. Because once the equation of the curve [$y = f(x)$] is introduced, the x–domain of the equation is divided into number of intervals with an equal increment (Δx = constant) as shown in Figure (3). After that, the x and y coordinates of the points at each interval on the curve are calculated, then a series of linear segments are inserted between every two adjacent points. At this stage the curve is divided into series of sub curved segments with their corresponding linear segments.

The chordal deviation (error) for each curved segment is calculated and the maximum one is extracted. When the maximum chordal deviation is greater than the required tolerance, the increment (Δx) is reduced to find a new maximum chordal deviation and the calculation is retrieved until the maximum chordal deviation reaches the required tolerance.

Since the curvature of the curve is highly different along it, an important drawback of this approach is obvious, that is the chordal deviation for each curved segment is different from the other, since that this approach is not based on the curvature variance; therefore referring to Figure (3) some curved segments may have a chordal deviation less than the required tolerance, but the others may have a chordal deviation greater than it. This leads to a recursive decomposition resulting in a large number of curved versus linear segments, which in turn result in a long tool path(s) which can be translated into a long machining time and finally a wide reduction in productivity will appeared.

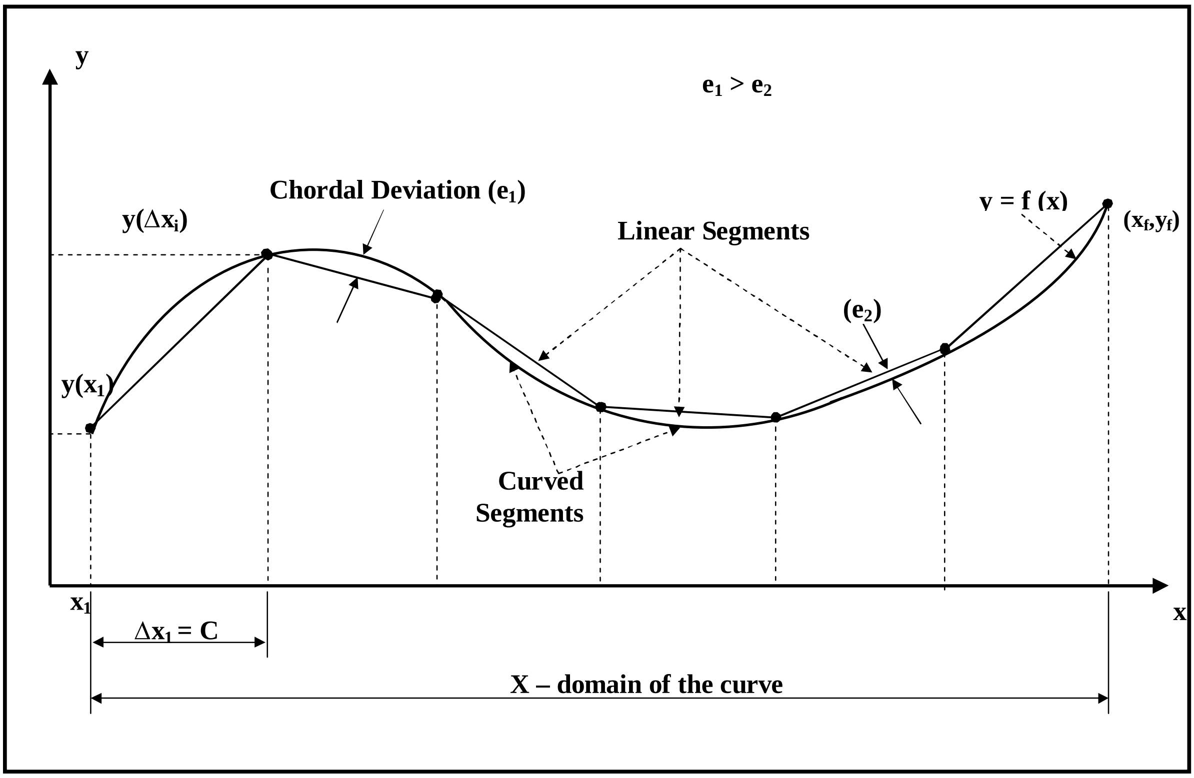


Figure (3) Schematic representation of ILCA procedure, shows a fluctuation of chordal deviation from high to low along the X-domain of the curve.

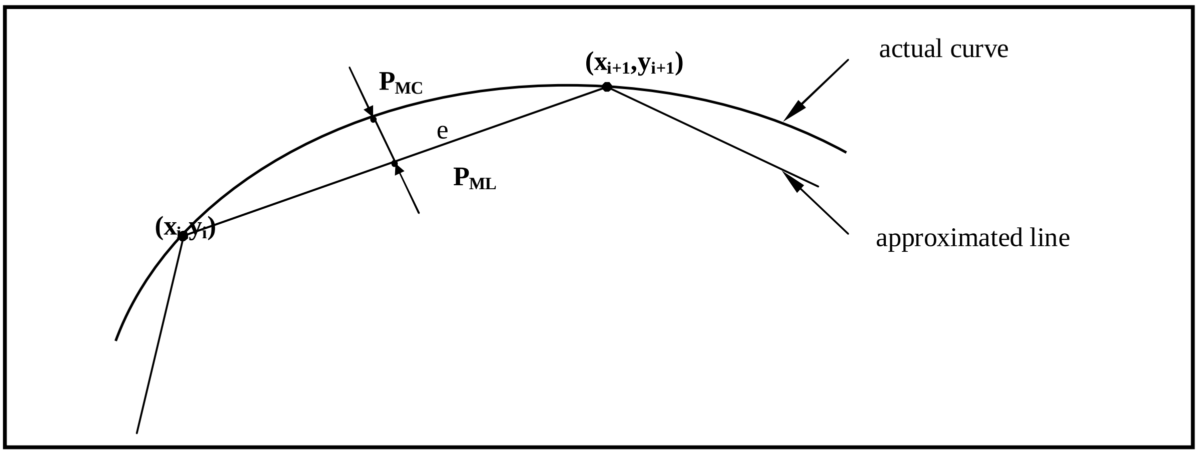


Figure (4) (Exaggerated) chordal deviation between the actual curve and approximated line

Calculation of Chordal Deviation (e)

Assuming that the x–axis increment (Δx) is small, the chordal deviation may be regarded as the distance from the mid point (P_{ML}) on the straight line segment to the mid point (P_{MC}) on the curved segment as shown in Figure (4).

where:

$$x_{pmL} = \frac{x_i + (x_{i+1})}{2} \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

$$y_{pmL} = \frac{y_i + (y_{i+1})}{2} \quad \ldots\ldots\ldots\ldots\ldots\ldots (2)$$

The equation of the normal line passing through (P_{ML}) can be calculated as follows:

$y = m \,.\, x + c$ ……………………… *(3)*

m: is the slope of the normal line;

$$m = \frac{x_i - (x_{i+1})}{(y_{i+1}) - y_i} \quad \ldots\ldots\ldots\ldots\ldots\ldots (4)$$

c: constant;

$c = y_{PML} - m \,.\, x_{PML}$ ……………….*(5)*

The standard form of the curve equation may be given by:

$y = c_1\, x^2 + c_2\, x + c_3$ ……….. *(6)*

Equating equation (3) and equation (6) we get:

$c_1\, x^2 + c_2\, x + c_3 = m.x + c$

Re-arranging this equation yields:

$x^2 + \varphi\, x + \lambda = 0$ ………………….*(7)*

Solving equation (7) yields:

$$x_{PMC} = \frac{-j \pm \sqrt{j^2 - 4l}}{2} \quad \ldots\ldots(8)$$

Since the curve is defined by the analytic form *y=f(x)*, the x_{pc} can be determined by substituting equation (8) in the curve function as follows:

$y_{PMC} = f\,(x_{pc})$ ……………………*(9)*

Accordingly chordal deviation can be calculated as follows:

e = ½ *power of* $\left[(x_{PML} - x_{PMC})^2 + (y_{PML} - y_{PMC})^2\right]$ ………*(10)*

Algorithm for ILCA Approach:

The procedure of ILCA approach can be integrated into an iterative solution as follows:

1. Set an initial Δx.
2. Set required tolerance (T).
3. Divide the x – domain into (n) number of intervals where:

   $$n = \frac{x_f - x_1}{\Delta x} \quad \ldots\ldots\ldots\ldots\ldots\ldots (11)$$

4. Calculate the coordinate (x_i) for each interval starting with (x_1) and ending by (x_f) as follows:

   $x(i) = x_1 + e.\Delta x$ …………… *(12)*

5. Calculate the coordinate (y_i) corresponding to each value of (x_i):

   $y(i) = f\,[x(i)]$

6. Join the calculated points with straight-line segments end with end.
7. Calculate the chordal deviation (e) for each interval (equation 10).
8. Find the maximum chordal deviation (e_{max}).

9. If $e_{max} > T$, then reduce Δx and go to step (3).
Otherwise:
10. End.

. Best Line Curve Approximation Procedure (BLCA)

This method is a curvature – based curve approximation; therefore, this approach is more efficient compared with (ILCA) approach, though the analyses of this approach are more complex. This approach is based on tangent and normal lines. This tangent line to the curve is regarded as aguideline for sliding (translation) the normal line along it, where the length of normal line represents a required tolerance. In this approach;
First, we find the tangent line of the curve at the point (P_1) as shown in Figure (5), the equation of tangent line is given by [8]

$$y_t = f'(x_1).x + (y_1 - x_1) \ldots\ldots\ldots(13)$$

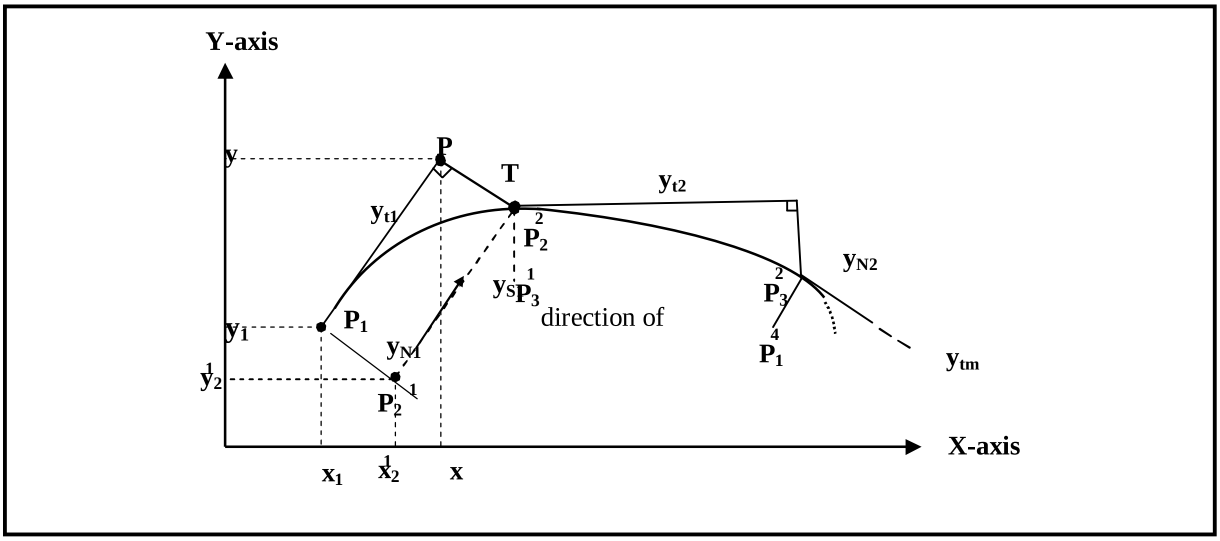


Figure (5) Decomposition procedure of (BLCA) approach, shows the translation stages of normal line (y_N) along to the guide line (tangent line)(y_t)

Second, calculation of the equation of the normal line at (P_1), the length of this line should equal to the required tolerance (T), the equation of this normal line can be given by [8]:

$$y_n = -\frac{1}{f'(x_1)} x + (y_1 - x_1) \ldots\ldots\ldots\ldots\ldots (14)$$

Third, the point P_2^1 at the end of the normal line must be calculated according to the following condition:

$$\| P_1 \; P_2^1 \| = T \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (15)$$

Fourth, the normal line joining $P_1 \; P_2^1$ is translated (moved) along the guide line (tangent line) until the point (P_2^1) is coincident with the curve at the point (P_2^2) see Figure (5).
Since the line y_{t1} is parallel to the secant line (y_{s1}) joining points $P_2^1 \,\&\, P_2^2$, a simple modification to Equation (13) can be made to find the equation of secant line as follows:

$$y_s = f'(x_1)(x - x_2^1) - y_2^1 \ldots\ldots\ldots\ldots (16)$$

Accordingly, the general intersection point (P_2^m) between the curve and any secant line can be determined by equating equation 6 and equation 16 as follows:

$$c_1 x^2 + c_2 x + c_3 = f'(x_1)(x - x_2^1) - y_2^1$$

$$c_1 x^2 + c_2 x + c_3 - f'(x_1)x + f'(x_1)x_2^1 + y_2^1 = 0$$

$$x^2 + \frac{c_2}{c_1}x + \frac{c_3}{c_1} - \frac{f'(x_1)}{c_1}x + \frac{f'(x_1)}{c_1}x_2^1 + \frac{y_2^1}{c_1} = 0$$

$$x^2 + \left(\frac{c_2}{c_1} - \frac{f'(x_1)}{c_1}\right)x + \frac{c_3 + f'(x_1)x_2^1 + y_2^1}{c_1} = 0$$

The above equation con be reduced to the following form

$$x^2 + \Gamma x + y = 0$$

Solving this equation we get:

$$x_2^m = \frac{-\Gamma \pm \sqrt{\Gamma^2 - 4y}}{2} \quad \ldots\ldots\ldots\ldots (17)$$

$$y_2^m = f(x_2^m) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (18)$$

where $m \geq 2$

When we find the point (P_2^m), at this moment the point (P_2^m) is fixed and the procedure from first to fourth stage is returned to find the next tangential line (y_{t2}) at (P_2^2) and so on as shown in the previous Figure (5).

Length of Tool Path (LTP)

To deduce, with confidence, that the generated tool path using (BLCA) approach and (ILCA) approach is highly different, a length of generated tool path may be a good criterion to assess the two presented approaches.

The total length of tool path (LTP) using ILCA approach can be calculated as follows:

$$LTP = \sum_{i=1}^{n} \text{½ power of } [(x_{i+1} - x_i)^2 + (y_{(xi+1)} - y_{(xi)})^2] \ldots\ldots (19)$$

On other hend the total length of tool path using BLCA approach is calculated from the following equation

$$LTP = \sum_{j=1}^{np} \| p_j^1 p_j^2 \| + MT \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (20)$$

Results and Discussion

In this section, two numerical experiments are presented to verify the proposed curve approximation strategies:

Example I

The following experiment present the input data of Example I :

1. The equation of the curve:
   $y = -0.043x^2 + x + 0.01$

2. X-Coordinate of the start point:
   $x_1 = 2$
3. X-Coordinate of the end point:
   $x_f = 11$

4. Required tolerance:
   t=0.05

5. The initial length of x-axis interval
   $\Delta x_1 = 3$ as a start of iteration of computer program.

The output data of ILCA program:

1. The final calculated length of x-axis interval
   $\Delta x_f = 0.04$

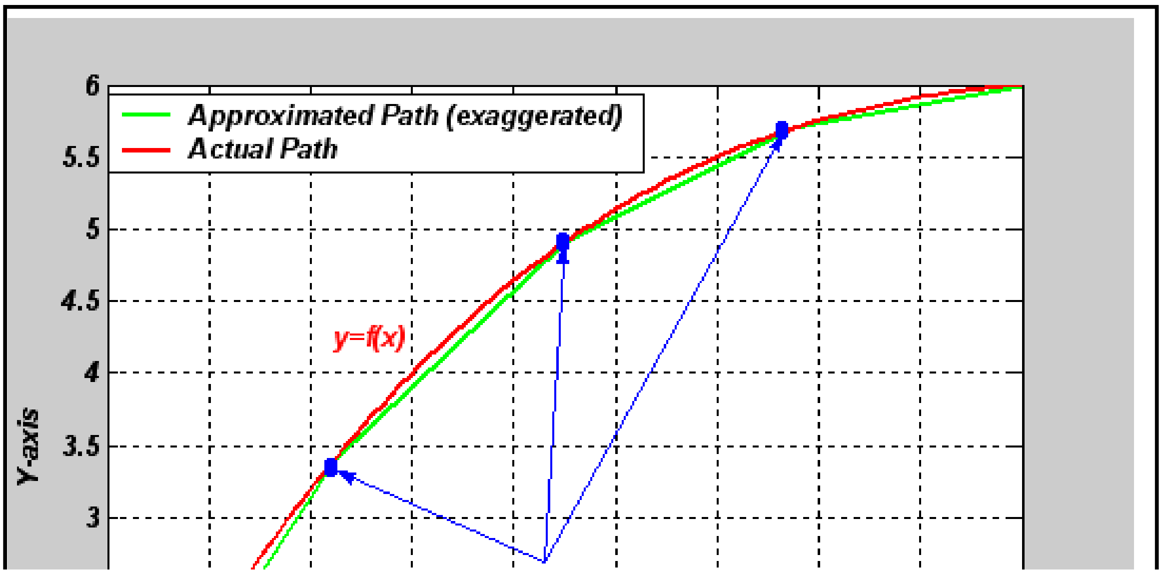
2. The Max. calculated chordal deviation (error)
   $e_{Max} = 0.0493$

3. The total length of tool path
   $LTP_{total} = 63.4$ Unit Length

4. The number of linear segments = 226

Figure (6) shows the exaggerated of approximated curve of ILCA approach using computer program implemented on IBM compatible PC. This program is running using MATLAB programming language[9]

Figure (7) shows the exaggerated of approximated curve BLCA approach.



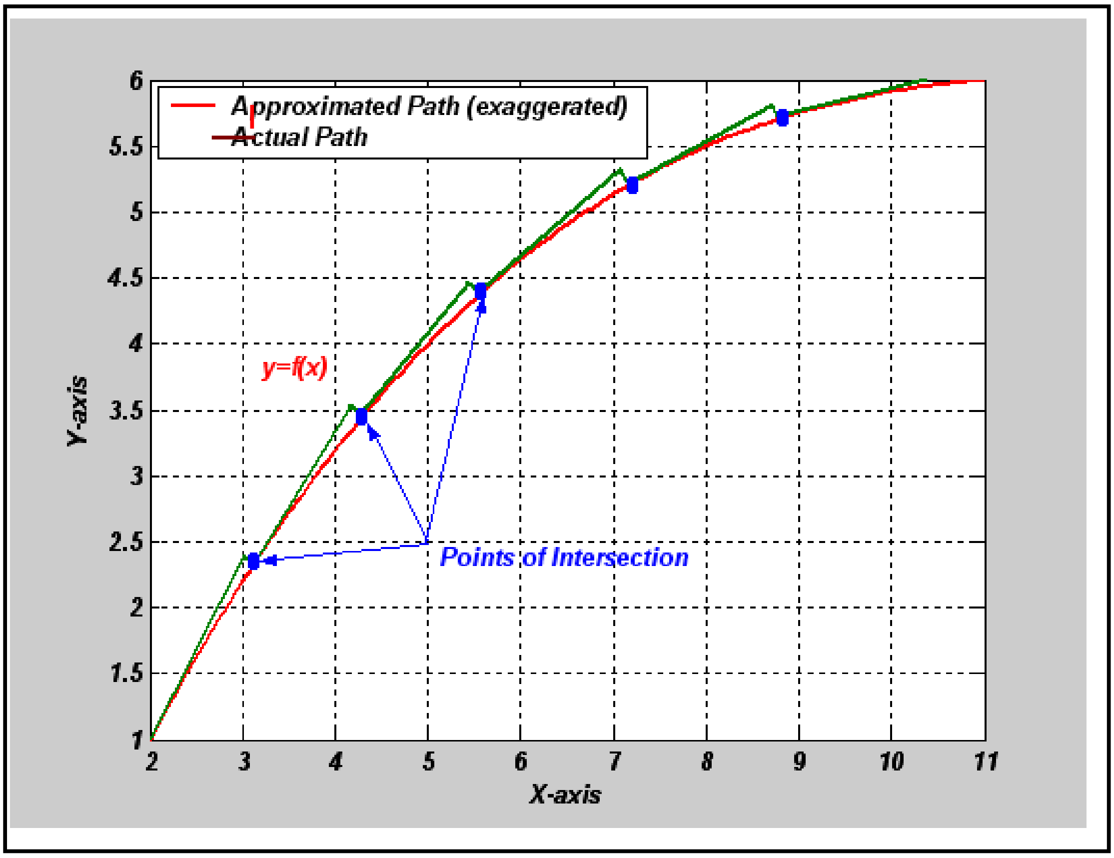


Figure (7) Exaggerated tool path of BLCA technique (Example I)

1. Max. chordal deviation
   $e_{Max} = 0.05$

2. The total length of tool path
   LTP_{total} = 41.844 Unit length

3. No. of linear segments = 90

Example II

The following input data are used for Example II:

1. The equation of the curve:
   $y=2x^2-x+1$

2. X-Coordinate of the start point:
   $x_1 = 1$

3. X-Coordinate of the end point:
   $x_f = 10$

4. Required tolerance:
   t=0.045

5. The initial length of x-axis interval
   $\Delta x_1 = 2$ as a start of iteration of computer program.

The output data of ILCA program:

1. The final calculated length of x-axis interval
   $\Delta x_f = 0.02$

2. The Max. calculated chordal deviation (error)
   $e_{Max} = 0.039$

3. The total length of tool path
   $LTP_{total} = 425.25$ Unit Length

4. The number of linear segments = 451

The output data of BLCA program:

1. Max. chordal deviation
   $e_{Max} = 0.045$

2. The total length of tool path
   $LTP_{total} = 282.15$ Unit length

3. No. of linear segments = 251

Figure (8) and figure (9) show the exaggerated approximated curve of ILCA and BLCA approach.

For the purpose of completeness, the discussion on obtained results is based on the following criteria:

- Number of linear segments
- Total length of tool path
- Over-cut and under-cut areas

Since the strategy of curve decomposition for BLCA is different from that of ILCA, hence the resulting number of linear segments will differ, i.e. the curve decomposition of BLCA depends on a tangential line, while the ILCA depends on secant line. Figure (10) shows the produced error of ILCA which is smaller than the required tolerance over a semi-log graph versus number of linear segments, this figure shows that as the produced error is reduced, the number of linear segments is increased. Accordingly, the total length of tool path will also increase. The obtained results confirm this truth, because the produced error of ILCA (e_{max} =

0.0493) results in a 226 linear segments and a total length of tool path about (63.4 unit length). on the other hand, the produced error of BLCA (e_{max} = 0.05) results in a 90 linear segments and a total length of tool path about (41.844) i.e., a small increase in the produced error (0.05 – 0.0493 = 0.0007) will give a reduction in the total length of tool path about (63.4-41.844 = 21.556). The over-cut and under-cut areas may be a very important task in the generation and planning of tool paths. The over-cut problem may be defined as moving the tool out off the boundary of the work-piece while the under-cut problem is moving the tool inside the boundary of the work-piece.

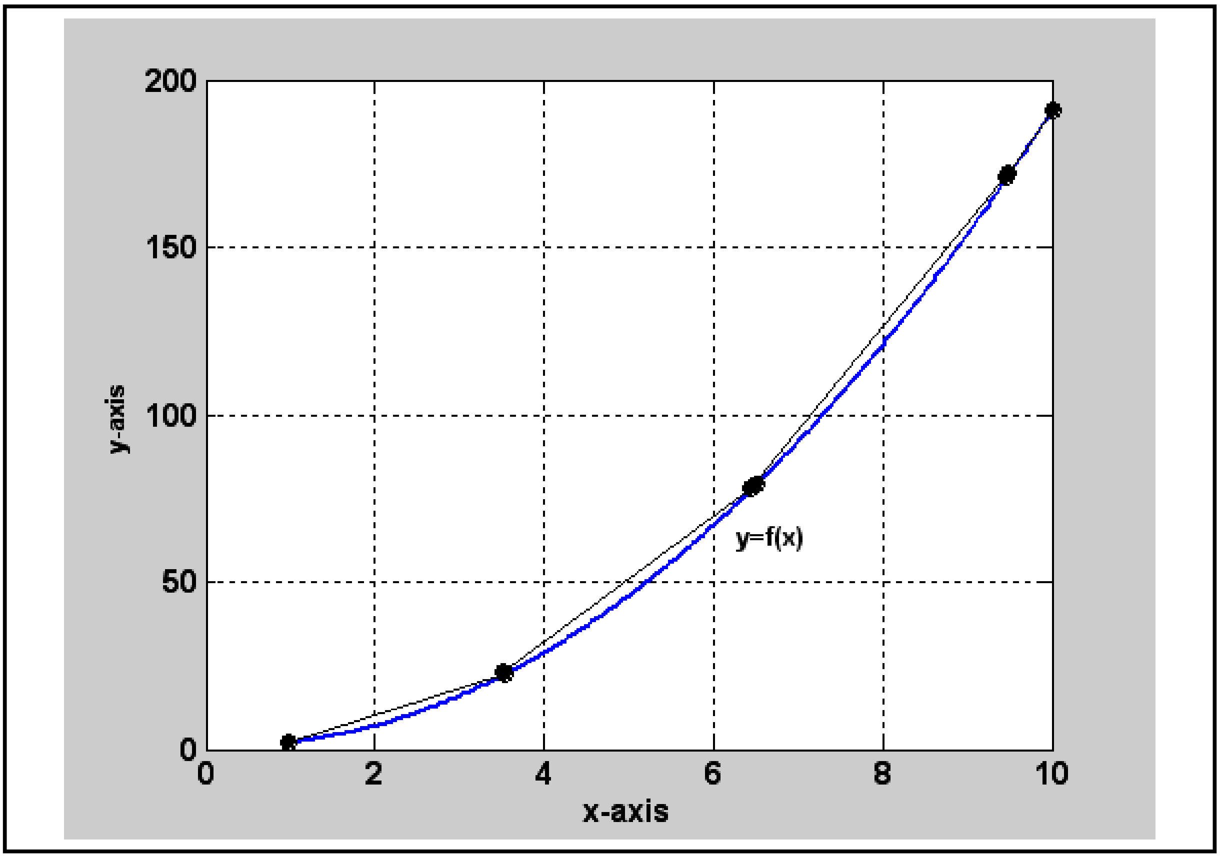
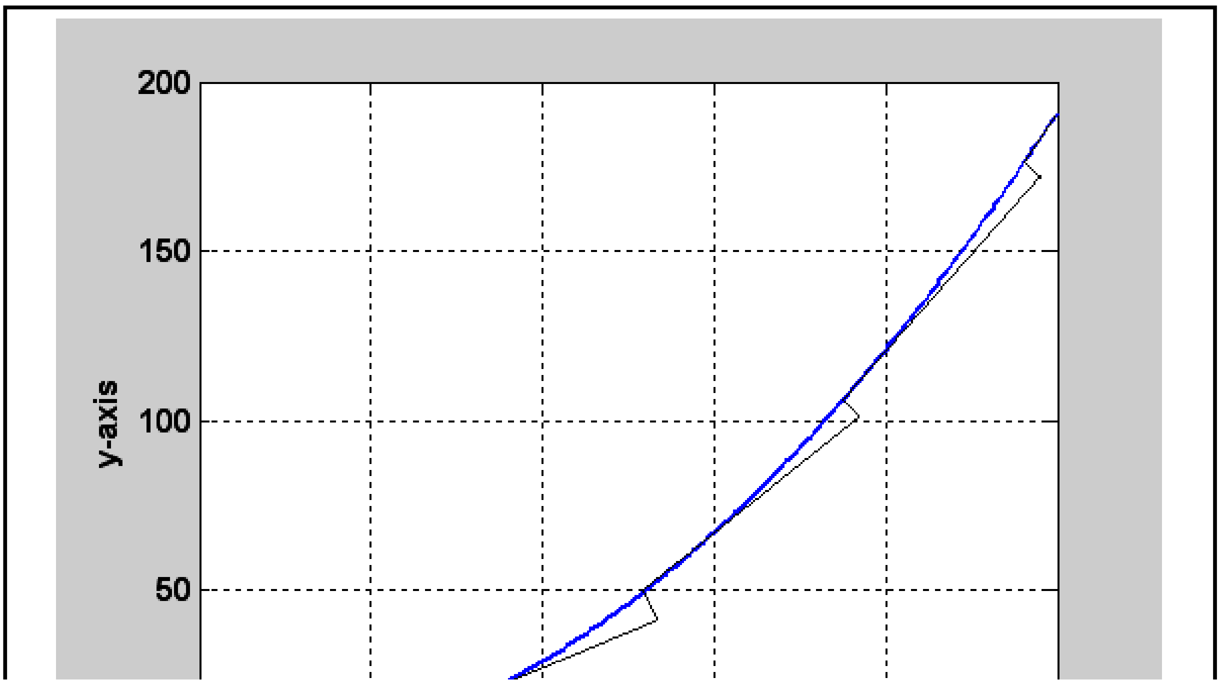
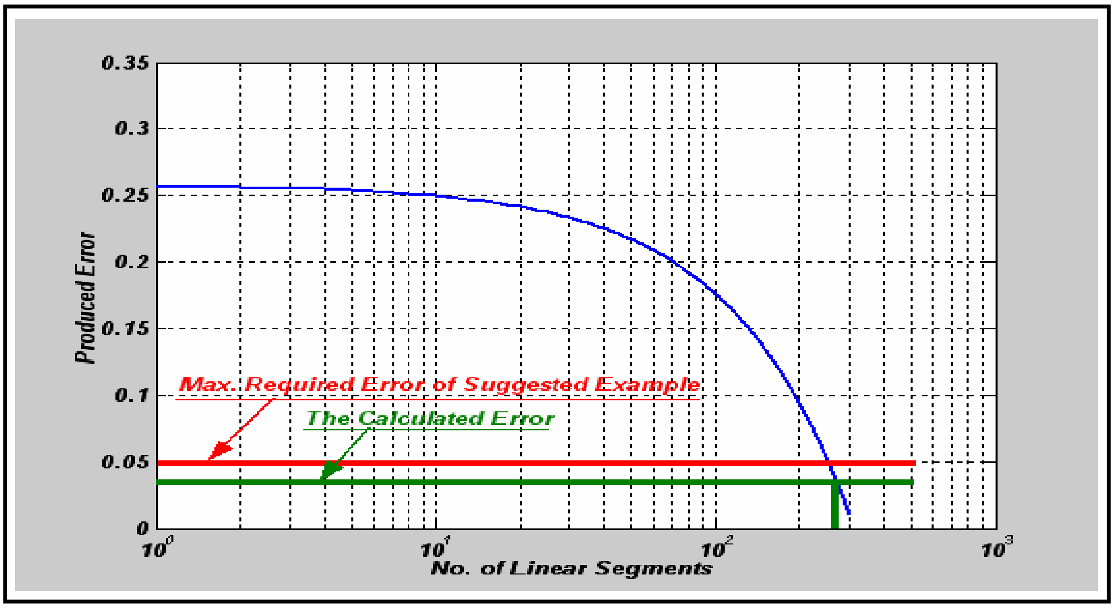


Figure (8) Exaggerated tool path of ILCA technique (Example II)



From the previous Figure (7) it is obviously noted that tool path generated from BLCA

approach suffes from an important drawback that is over-cut metal left by the cutter during the translation from the end path to the next one. This over cut metal depends on the cutter diameter and the angle of lab as shown in Figure (11).



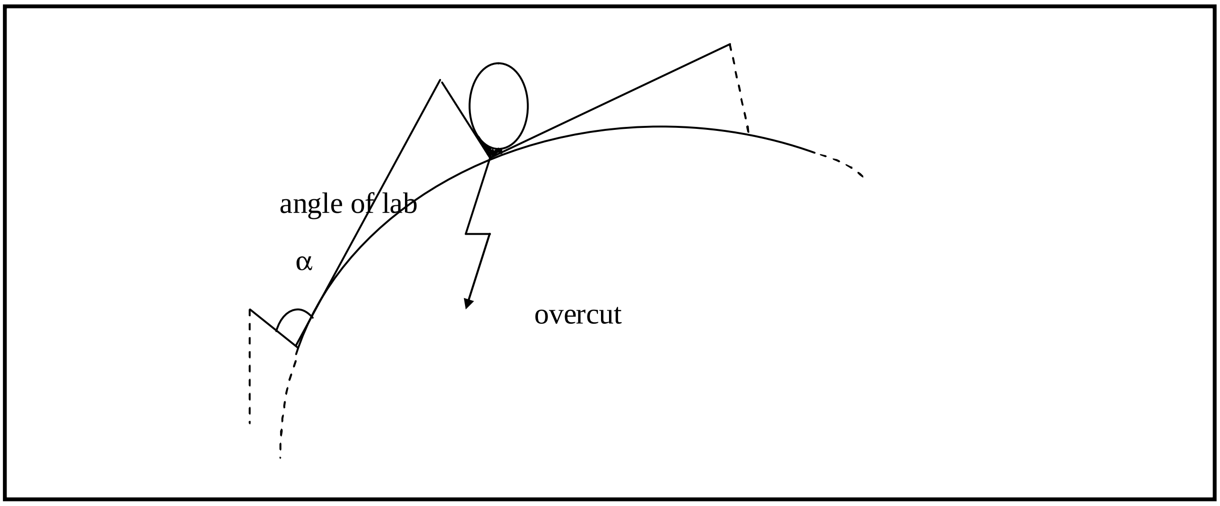


Figure (11) over-cut area as the drawback of BLCA approaches

he other hand, the problem of over-cut metal is absent from ILCA approach, but the problem of under cut is introduced here.

When the cutter is translated from the location P1 to P2 (see Figure 12-a), a specific area (dark area in Figure 12-a) will be removed.

This problem can be eliminated by inserting a specific point (P_{of}) between P_1 and P_2 as show in Figure (12-b). In this case the cutter will be translated according to the following sequence (P_1 **à** P_2 **à** P_3) to overcome the problem of undercut areas.

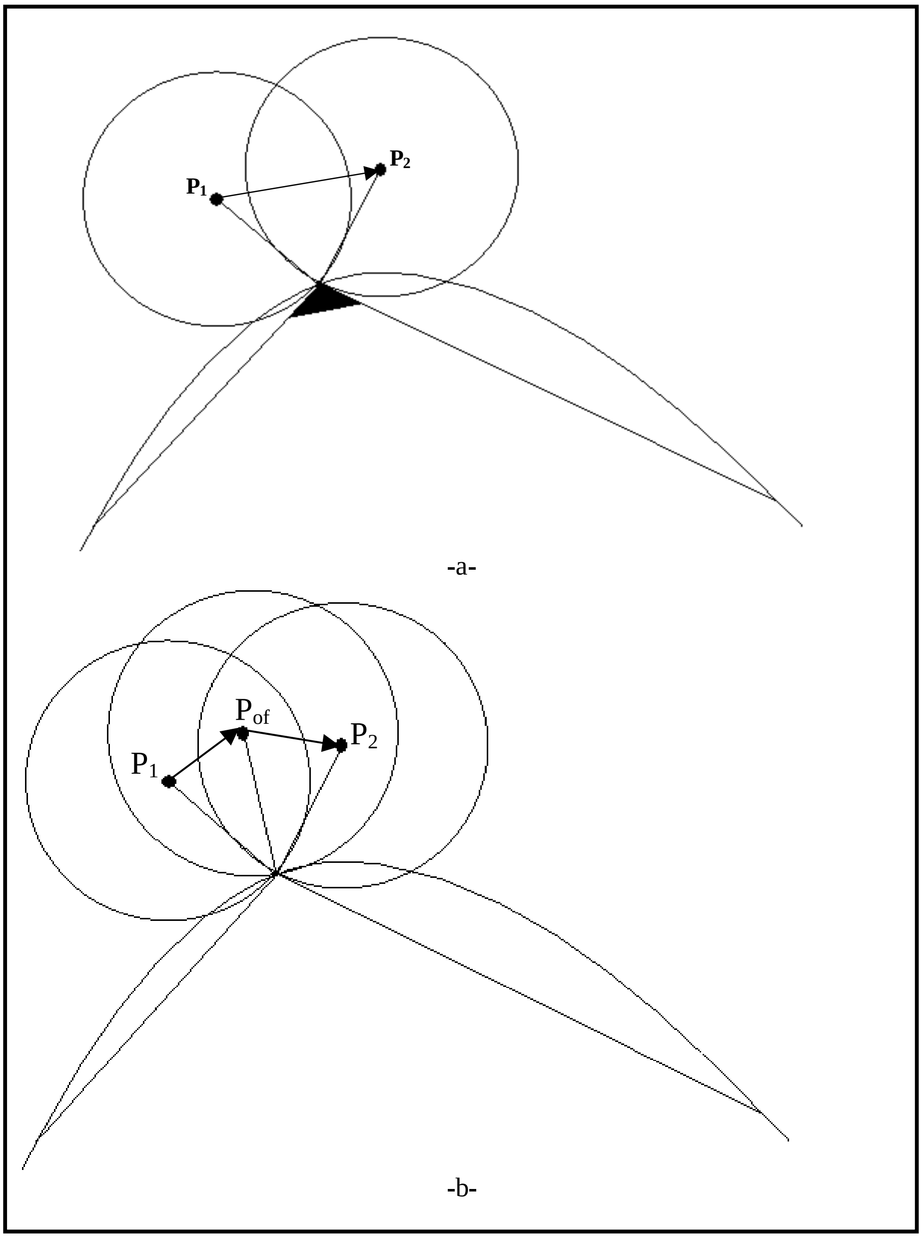


Figure (12) The problem of under-cut area of ILCA approach
(a) Under-cut area as the cutter translated directly from P_1 to P_2.
(b) Removing of under-cut area as the cutter translated from P_1 to P_2 and the finally to P_2.

Conclusions

From the presented algorithms, the following remarks can be concluded and stated as follows:

1. The number of linear segments of BLCA approach is less than that of ILCA approach.

2. The BLCA approach gives a wide reduction in the total length of tool path approximately up to 66% smaller than the total length of tool path of ILCA approach.

This reduction in length of tool path can be translated to a machining time, which results in a wide increasing of productivity. Consequently, the BLCA approach is better from the other approach, but it requires further modifications to overcome the problem of over-cut areas.

References:

1. Yan Zhou and Chining-Kuang Chene, "*Curve and Surface Interpolation and Approximation: Knowledge Unit and Software Tool*", http://www.cs.mtu.edu/~shene/NSF-2, 1999.
2. Qiyi G. Zhang Greenway R. Bryan, "Development & Implementation of a NURBUS Curve Motion Interpolation", *Robotics & Computer – Integrated Manufacturing*, Vol. 14, pp. 27-36, 1998.
3. Roogers and Sons, *Numerical Control & Computer Aided Manufacturing*, John Wiley & Sons. Inc., 1986.
4. Kim Kwongsoo and John E. Biegl, "A Path Generation Method for Sculptured surface Manufacturing", *Computer Ind. Eng*. Vol. 14, No. 2 , pp.95-101, 1988.
5. Seuny Ryol Maeng, Nakhoon Beak, Sung Yong Shing & Byong Kyu Choi, "A Z-Map update Method for Linearly Moving Tool", *Computer Aided Design*, WWW.elesvier.com/locate/cad, 2000.
6. Jacob A. Crossman & Dovid H. Yoon, "*A Cutter Motion Simulation System*", *Transaction of the SDPS*, Vol. 4, No. 1, pp.25-35, March 2000.
7. Susan X. Li & Robert B. Jerard, "*Non-isoparametric Three Axis NC Tool Path Generation for Finish Machining of Sculptured Surfaces*", 2000.
8. Howard Auton, Irl Bivens & Stephen Davis, *Calculus*, John Wiley & Sons. Inc., 2002.
9. Brian D. Hahn, *Programming Style with MATLAB*, Math-works, Inc., 1997.