NISTIR 6517

A Spline Algorithm for Modeling Cutting Errors on Turning Centers

David E. Gilsinn
Herbert T. Bandy
Alice V. Ling

U.S. DEPARTMENT OF COMMERCE
Technology Administration
Automated Production Technology
Division
National Institute of Standards
and Technology
Gaithersburg, MD 20899

NIST
National Institute of Standards and Technology
Technology Administration
U.S. Department of Commerce

NISTIR 6517

A Spline Algorithm for Modeling Cutting Errors on Turning Centers

David E. Gilsinn
Herbert T. Bandy
Alice V. Ling

U.S. DEPARTMENT OF COMMERCE
Technology Administration
Automated Production Technology
Division
National Institute of Standards
and Technology
Gaithersburg, MD 20899

May 2000

U.S. DEPARTMENT OF COMMERCE
William M. Daley, Secretary

TECHNOLOGY ADMINISTRATION
Dr. Cheryl L. Shavers, Under Secretary
of Commerce for Technology

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Raymond G. Kammer, Director

Abstract

Turned parts on turning centers are made up of features with profiles defined by arcs and lines. An error model for turned parts must take into account not only individual feature errors but also how errors carry over from one feature to another. In the case where there is a requirement of tangency between two features, such as a line tangent to an arc or two tangent arcs, any error model on one of the features must also satisfy a condition of tangency at a boundary point between the two features. Splines, or piecewise polynomials with differentiability conditions at intermediate or knot points, adequately model errors on features and provide the necessary degrees of freedom to match constraint conditions at boundary points. The problem of modeling errors on features becomes one of least squares fitting of splines to the measured feature errors subject to certain linear constraints at the boundaries. The solution of this problem can be formulated uniquely using the generalized or pseudo inverse of a matrix. This is defined and the algorithm for modeling errors on turned parts is formulated in terms of splines with specified boundary constraints.

Key Words: error modeling; generalized inverse; least squares; machine tool; pseudo inverse; spline

1.0 Introduction

Errors in a machined part are due to several sources. There are errors inherent in the machine itself due, for example, to misalignment of slide ways and other geometric errors. There are errors due to thermal deformations of the machine while operating. There are also errors caused by inaccurately specified tool dimensions, tool wear, tool and/or part deflection, and so on. We will call these types of errors the "process related errors". It is the modeling of these process related errors for a turning center that will concern us in this report.

The object of developing process error models is to apply them in error compensation strategies [1, 2]. Process errors can be measured during machining [3] or by process intermittent gauging. Process-intermittent gauging has an advantage in that a simple measurement device, such as a touch-trigger probe, can be inserted into the tool changer. This form of probe is less intrusive than apparatus required for measurement during machining. Process-intermittent gauging of process-related part errors usually takes place between semi-finish and finish machining processes. This permits on-line modeling of process-related errors, the results of which are then used to anticipate and compensate these errors in the finish process. For a discussion of process-intermittent probing and real-time error compensation, see [4, 5].

One form of error compensation strategy requires interpreting a part as consisting of separate features. Such a decomposition of a part is useful for establishing correspondence between design information and manufacturing operations [6]. Part features can be defined very generally. For a turning center, however, in which part geometry is defined in two dimensions, the features of concern are the arcs and lines that comprise the CAD profile of the part. CAD-based methods facilitate the creation of pre-process data such as feature geometry, nominal coordinates of gauging points, and surface normal vectors.

Any error model for turned parts must take into account not only errors on individual features but also how the errors carry over from one feature to another. This just reflects the physical fact that as a tool cuts a feature of a part it transitions in a smooth manner to cutting the next feature. This implies that there should not be any unintentional changes in slopes between features. Therefore, a feature error model must take into account slope constraints at the ends of the features.

Another aspect of modeling machine tool errors is the need to create model function forms that can be computed rapidly when the models are implemented in error compensation strategies. This often means that functional forms need to be low order polynomials. However, low order polynomials may or may not model all the errors on a machined feature. If the geometry of a feature is broken into smaller parts, the errors on those smaller parts can often be modeled by low order polynomials. If the low order polynomials are chosen in such a way that the slopes are made equal at the feature part transition points, the combined piecewise polynomial is called a *spline* (see [7]).

The error-modeling algorithm described in this report combines the use of splines, to model the errors within a feature, with boundary slope constraints at the ends of the features. The general modeling technique involves a least squares fitting of a spline to process-intermittent, measured, machine-error data but with an extra requirement that slopes at the end of features be equal. These are usually linear constraints so the algorithm can be classified as a least squares fitting of a linear model with linear end constraints.

2.0 Modeling Errors on Features with Linear Profiles

When linear features join each other the modeling does not necessarily require splines but splines could be used. Regardless of whether splines are used, at least two cases of errors usually occur. First, if an analysis of the part errors indicates the existence of feature size errors only, a constant offset for either axis is sufficient to compensate the errors. In this case, the compensation software inserts the appropriate values in the tool offset update command in the numerical control (NC) program segment for the finish cut and all coordinates in the NC program are left at their nominal values. As an alternative means of compensating such errors, the compensation software also writes the axis offsets to a file, which is used for real-time compensation. Second, if errors are essentially linear but the slopes are different from those of the nominal features, the compensation software can adjust the finish cuts for each feature. Adjustments for features with nominally linear profiles are usually calculated by fitting linear functions through the error vectors computed at the gauge points for each cut of the part. The intersections of the linear equation for a cut with similar linear error equations for the neighboring cut on each side give the errors at the endpoints of the cut. These endpoint errors are used to adjust the points that are then entered into NC program for the finish cut and are written to a file that is used to provide data to generate real-time cut adjustments. Elaboration of these procedures may be found in [8, 9, 10].

3.0 Modeling Errors on Features with Curved Profiles

If a part contains a feature whose nominal profile is not linear, the adjustments are more complex. For example, when an arc smoothly meets a line or another arc, not only do the compensation curves intersect but the two curves must usually be tangent to each other. The treatment of a circular arc profile is explained in this section. The principles, however, can be extended to non-circular curves.

Some earlier work in compensating errors on a hemispherical nose of a turned part showed that error compensation on arcs was feasible [11]. No attempt, however, was made in this previous work to maintain tangencies at feature boundaries. Although the previous work showed that process-intermittent errors in curved features could be compensated, the application was limited to a turned hemisphere generated by a nominal arc cut, because a circle could be fitted to the probed data. However, turning centers can generate other types of curved cuts, which are better fitted by spline modeling.

Furthermore, the previous work did not consider what would happen at the interface between two features such as a linear feature tangent to a curved feature. If two curves are fit separately to probed values on each feature, then the resulting curves might have a discontinuity at the nominal point of tangency. In the finish cut, this could lead to a significant step in the part. Therefore, another data fitting procedure had to be investigated to compensate errors on general, turned, curved features that might have various interface angles to neighboring features. That is, a least squares technique with prescribed boundary conditions had to be developed. This problem cannot be treated as a standard least squares problem since the boundary conditions restrict the selection of the fitting parameters.

Polynomials are useful as approximation functions to unknown and possibly very complex nonlinear relationships. However, the literature on least squares regression models [12, 13] warns that it is important to keep the order of the polynomial models as low as possible. In an extreme case it is possible to pass a polynomial of order n-1 through n points so that the polynomial of sufficiently high degree can always be found that provides a "good" fit to the data. The behavior of the polynomial between the data points may be highly oscillatory, though, and not provide good data interpolation. Figure 1 is a good example of the oscillatory behavior of a high order interpolating polynomial. The probe data in the figure represents micrometer errors measured on the Z-axis of travel on a turning center. Notice that the interpolating polynomial goes through each of the data points, but only produces a good fit between the data points within the mid-range of the data. The interpolating polynomial, however, performs large excursions near the ends of the data set. This is a typical behavior of a high order interpolating polynomial.

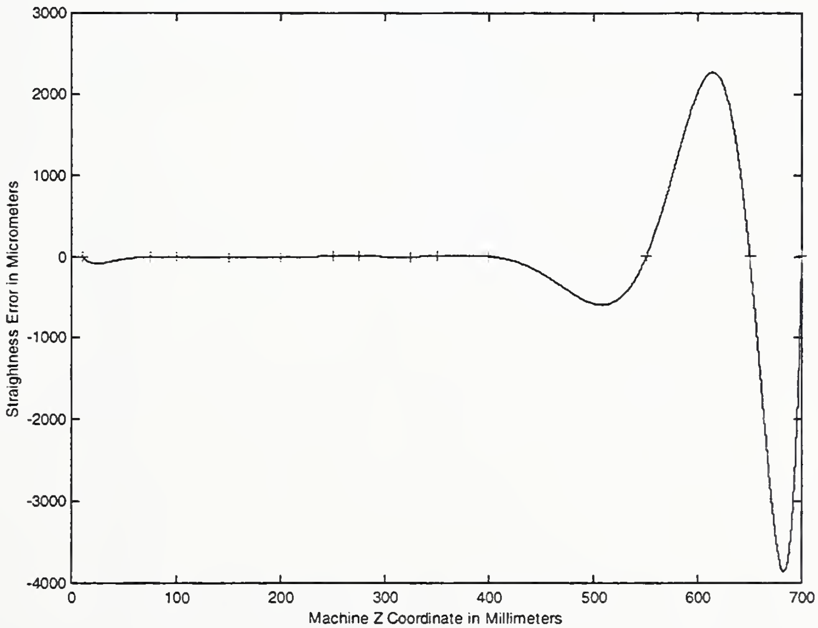

Figure 1: The results of interpolating with a polynomial of order 12. Note the large oscillation at the right end.

When a function behaves differently in different parts of the range of the independent variable, the usual approach is to divide the range of the independent variable into segments and fit an appropriate curve to each segment. Spline functions offer a way to perform this type of piecewise polynomial fitting and provide smooth transitions, if desired, between neighboring segments.

Splines are generally defined to be piecewise polynomials of degree n. The function values and first n-1 derivatives are set to agree at the points where they join. The abscissa coordinates of these joining points are called *knots*. Thus, a spline is a continuous function with n-1 continuous derivatives. Polynomials may be considered a special case of splines with no knots, and piecewise polynomials with fewer than the maximum number of continuity restrictions may also be considered splines. The number and degrees of the polynomial pieces and the number and position of the knots may vary in different situations.

Figure 2 shows the results of interpolating the same probe data as in Figure 1 but using a clamped spline (note that the scale of the ordinate axis is different from Figure 1). A clamped spline means one with prescribed derivative conditions specified at the end points of the data. This figure shows vividly the benefits of interpolating with spline functions. The ability to interpolate with piecewise polynomial allows a tighter control on the interpolation errors.

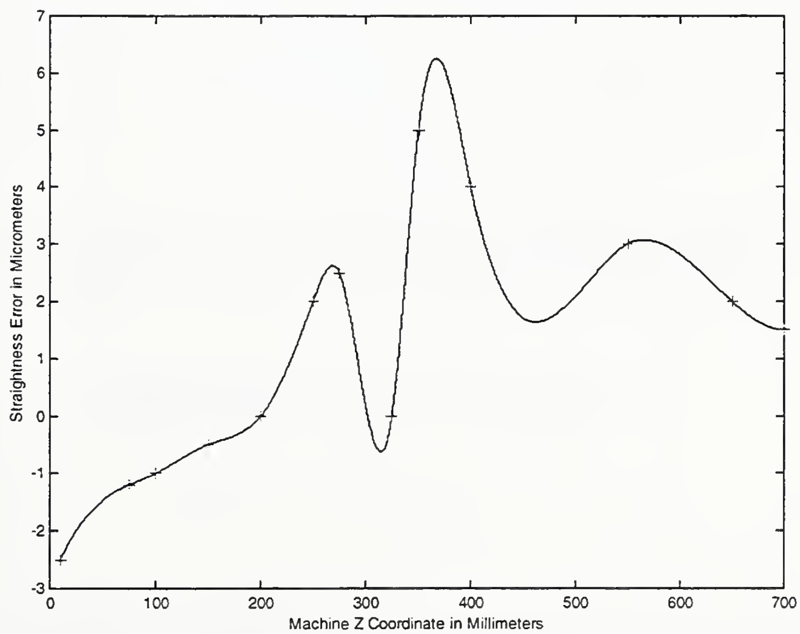

Figure 2: Interpolating the same data from Figure 1 using a clamped cubic spline. Note the close modeling of the data.

4.0 Constructing Interpolating Splines with End Constraints

It is possible to construct a *basis*, or sequence of functions, such that every spline of interest can be written in one and only one way as a linear combination of these functions [14]. General cubic splines ($n = 3$) will be used since they have been shown to be adequate for most practical problems. They can be written in terms of basis functions as follows:

Let an ordered sequence of k knots be given. These can be nominal probe points, but do not necessarily have to be

$$a \leq t_1 < t_2 < \dots < t_k \leq b \qquad (1)$$

A cubic spline with these k knots can be written as

$$y(x) = \sum_{j=0}^{3} c_j \, x^j + \sum_{j=1}^{k} c_{j+3}(x - t_j)_+^3 \qquad (2)$$

The $c_j,\ j = 1, \cdots, k+3$ are constants and

$$(x - t_j)_+^3 = \begin{cases} (x - t_j)^3 & x > t_j \\ 0 & x \leq t_j \end{cases} \qquad (3)$$

This cubic spline representation has continuous first and second derivatives. See [12, 13] for good general discussions of the use of splines in statistical data analysis.

Assume that there are s sampled points in the plane given by the pairs $(x_1, y_1), \cdots, (x_s, y_s)$ and suppose that the x-values are ordered by

$$a \leq x_1 < x_2 < \dots < x_s \leq b, \qquad (4)$$

where a and b are bounds for the sequence of x-values. Since the sampled points might show undesired oscillations or "noise", some form of smoothing will be obtained by not selecting knots at each point. In fact, guidelines in the literature [12, 13] suggest 4 to 5

points between knots. Since this will not always be possible one can select this value as a variable, say r, and set a knot at every r-th point. This would mean that one first selects an integer k so that $kr \leq s$. This selection of knots partitions the sampled points into the sets

$$\begin{gathered} x_1 < x_2 < \ldots \\ < x_r(=t_1) < x_{r+1} < \ldots < x_{2r}(=t_2) < x_{2r+1} \ldots < x_{kr}(=t_n) \qquad (5) \\ < x_{kr+1} < \ldots < x_s . \end{gathered}$$

The standard least squares problem of fitting a spline of the form (2) through the sample points can be formulated in matrix terms. To start, with define the residual at the q-th sample point q = 1, 2, ...,s, as

$$R_q = \begin{cases} y_q - \sum_{j=0}^{3} c_j x_q^j & q < r \\ y_q - \sum_{j=0}^{3} c_j x_q^j - \sum_{j=1}^{t} c_{j+3}(x_q - t_j)^3 & q \geq r \end{cases} \qquad (6)$$

where, for a given q-th point, t is the smallest integer so that $q \leq tr \leq s$. To begin formulating the matrix version of the least squares problem define the vectors

$$c = (c_0, \ldots, c_{k+3})^T, \quad y = (y_1, \ldots, y_s)^T$$

where the superscript T indicates a transposed vector. The least squares sum is usually formulated as

$$LS(c) = \sum_{q=1}^{s} (R_q)^2 . \quad (8)$$

However, referring to (6) one can define the following matrix

$$
A=\begin{bmatrix}
1 & x_1 & x_1^2 & x_1^3 & 0 & 0 & 0 & \cdots & 0 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
1 & x_r & x_r^2 & x_r^3 & 0 & 0 & 0 & \cdots & 0 \\
1 & x_{r+1} & x_{r+1}^2 & x_{r+1}^3 & (x_{r+1}-t_1)^3 & 0 & 0 & \cdots & 0 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
1 & x_{2r} & x_{2r}^2 & x_{2r}^3 & (x_{2r}-t_1)^3 & 0 & 0 & \cdots & 0 \\
1 & x_{2r+1} & x_{2r+1}^2 & x_{2r+1}^3 & (x_{2r+1}-t_1)^3 & (x_{2r+1}-t_2)^3 & 0 & \cdots & 0 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
1 & x_{nr+1} & x_{nr+1}^2 & x_{nr+1}^3 & (x_{nr+1}-t_1)^3 & (x_{nr+1}-t_2)^3 & (x_{nr+1}-t_3)^3 & \cdots & (x_{nr+1}-t_k)^3 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\
1 & x_s & x_s^2 & x_s^3 & (x_s-t_1)^3 & (x_s-t_2)^3 & (x_s-t_3)^3 & \cdots & (x_s-t_k)^3
\end{bmatrix}
\tag{9}
$$

The matrix **A** is s rows by $k + 4$ columns where

$$k\,r \leq s. \tag{10}$$

The least squares problem can now be formulated in matrix notation as

$$\min\|Ac - y\| \tag{11}$$

where the minimum is taken over all vectors c and the norm is the standard Euclidean norm.

The splines in this application are not unrestricted at their ends, however, and this changes the least squares problem in this case. In order to make the curved features match with neighboring features, restrictions must be placed on how the splines behave at the endpoints a and b of the interval. In particular, we will require that the splines go through specific points with specific slopes. Therefore, we will require the following conditions be satisfied:

$$
\begin{aligned}
y(a) &= y_0 \\
y(b) &= y_{s+1} \\
\frac{dy}{dx}(a) &= y_0^{(1)} \\
\frac{dy}{dx}(b) &= y_{s+1}^{(1)}
\end{aligned}
\qquad (12)
$$

where the right hand sides are prescribed by the matching requirements. These conditions can also be formulated as matrix equations. To do this, first write each of the conditions as

$$
\begin{aligned}
&\sum_{j=0}^{3} c_j a^j = y_0 \\
&\sum_{j=0}^{3} c_j b^j + \sum_{j=1}^{k} c_{j+3}(b - t_j)^3 = y_{s+1} \\
&\sum_{j=1}^{3} j\, c_j a^{j-1} = y_0^{(1)} \\
&\sum_{j=1}^{3} j\, c_j b^{j-1} + 3\sum_{j=1}^{k} c_{j+3}(b - t_j)^2 = y_{s+1}^{(1)}
\end{aligned}
\qquad (13)
$$

then define

$$
B = \begin{bmatrix}
1 & a & a_2 & a_3 & 0 & \cdot\ \cdot\ \cdot & 0 \\
1 & b & b_2 & b_3 & (b - t_1)^3 & \cdot\ \cdot\ \cdot & (b - t_k)^3 \\
0 & 1 & 2a & 3a_2 & 0 & \cdot\ \cdot\ \cdot & 0 \\
0 & 1 & 2b & 3b_2 & 3(b - t_1)^2 & \cdot\ \cdot\ \cdot & 3(b - t_k)^2
\end{bmatrix}
\qquad (14)
$$

and let

$$
f = (\, y_0, y_{s+1}, y_0^{(1)}, y_{s+1}^{(1)})^T . \qquad (15)
$$

The constraint equation becomes

$$B\,c = f. \qquad (16)$$

The constrained least squares problem is then the combined relations (11) and (16). The solution of this problem requires defining a generalized notion of an inverse of a matrix.

If **A** is an n x n nonsingular matrix then the solution of the matrix problem $\mathbf{Ac} = \mathbf{y}$ is given uniquely by $\mathbf{c} = \mathbf{A}^{-1}\mathbf{y}$. But in the least squares problem where **A** has m rows and n columns and m and n are not the same value, the question arises whether there is an n x m matrix **Z** so that $\mathbf{c} = \mathbf{Zy}$ where **c** is the unique minimum length solution of the least squares problem (11). In fact the answer is yes, and the matrix **Z** is uniquely determined by **A**. It is called the *generalized* or *pseudo inverse* of **A,** and is denoted by $\mathbf{A}^{+}$ [15].

It is not difficult to find the generalized inverse of a matrix **A** if **A** is properly decomposed. For this application one can introduce the decomposition of A called the *singular value decomposition* [15]. Any m x n matrix **A**, whose number of rows m is greater than or equal to the number of columns n, can be written as the product of an m x n column orthogonal matrix **U**, an n x n diagonal matrix **D**, and the transpose of an n x n orthogonal matrix **V**. Symbolically

$$A = U\,D\,V^{T} \qquad (17)$$

where

$$\begin{aligned} U^{T}U &= I \\ V^{T}V &= I \end{aligned} \qquad (18)$$

and I is the n x n identity matrix and **D** is the diagonal matrix

$$D = diag\ [\,d_{11},\ d_{22}, \ldots, d_{nn}\,] \qquad (19)$$

where d_{ii} could be zero for several i's. The generalized inverse of **A** can be written as

$$A^{+} = V\,D^{+}\,U^{T} \qquad (20)$$

where

$$D^{+} = diag\ [\ d_{11}^{+}, d_{22}^{+}, \ldots, d_{nn}^{+}\,] \qquad (21)$$

and

$$d_{ii}^{+} = \begin{cases} \dfrac{1}{d_{ii}} & d_{ii} > tol \\ 0 & d_{ii} \leq tol \end{cases} \qquad (22)$$

and *tol* is a tolerance that is often set in such a way that it is related to the reciprocal of the maximum allowed condition number (i.e. ratio of the largest eigenvalue to the smallest) for the matrix **D**.

One can now formulate the result that gives the solution to the constrained least squares problem (11), (16). The principal reference for this result is [15].

Given an m x n matrix **B** of rank k, an m-vector **y**, an r x n matrix **A**, and an r-vector **y** the linear least squares problem with equality constraints becomes one of finding an n-vector **c** that minimizes

$$\|Ac - f\| \qquad (23)$$

and satisfies the linear equalities

$$B\,c = f. \qquad (24)$$

This is just a general restatement of the problem described by (11) and (16) above.

Assuming that (24) is consistent, there is a unique solution that minimizes (23) subject to (24) [13]. It is given by

$$c = B^{+} f + (A\,Z\,)^{+}(\;\; y - AB^{+} f\;) \qquad (25)$$

where

$$Z = I_n - B^{+} B \qquad (26)$$

and I_n is the n x n identity matrix. For many usual cases one would have $n > r = k$. The generalized inverses are computed by the singular value decomposition technique.

In order to use the spline representation of the surface errors on a part it is easier to evaluate the spline in its individual cubic components between knots. To do this requires compacting the representation of the spline polynomial as the underlying variable passes each knot. The algorithm is straightforward and begins by assuming that there are k knots. First add two knots for the end points to make k+2 knots. Thus,

$$a = t_0 < t_1 < \ldots < t_k < t_{k+1} = b. \tag{27}$$

For k internal knots there will be k+4 spline coefficients. But when these are combine to form groups of four coefficients for each interval there will be 4k+4 coefficients. These will be defined as follows: For

$$a = t_0 \leq x \leq t_1 \tag{28}$$

the polynomial is given by

$$y(x) = c_1^* + c_2^* \, x + c_3^* \, x^2 + c_4^* \, x^3 \tag{29}$$

where

$$\begin{aligned} c_1^* &= c_1 \\ c_2^* &= c_2 \\ c_3^* &= c_3 \\ c_4^* &= c_4 \end{aligned} \tag{30}$$

That is, the first four coefficients of the new array, identified by the superscript asterisk, are the same as the spline coefficients. The other groups of four coefficients are computed as follows. For j = 1, 2, ..., k one has for

$$t_j \leq x \leq t_{j+1} \tag{31}$$

the polynomial

$$y(x) = c_{4j+1}^* + c_{4j+2}^* \, x + c_{4j+3}^* \, x^2 + c_{4j+4}^* \, x^3 \tag{32}$$

where

$$
\begin{aligned}
c^*_{4j+1} &= c^*_{4(j-1)+1} - c_{j+4}\ t_j^3 \\
c^*_{4j+2} &= c^*_{4(j-1)+2} + 3\ c_{j+4}\ t_j^2 \\
c^*_{4j+3} &= c^*_{4(j-1)+3} - 3\ c_{j+4}\ t_j \qquad (33) \\
c^*_{4j+4} &= c^*_{4(j-1)+4} + c_{j+4}
\end{aligned}
$$

5.0 Model Application

The part used to demonstrate the use of the algorithm is shown in Figure 3. It has a step portion on the largest diameter area, a long taper, a cylindrical section and a hemisphere. The software, in which the algorithm described in this report is embedded, called Process Intermittent Error Compensation Software (PIECS) (see references [6]), is used to compensate machining errors on all of these surfaces. The tool tip selected to turn the part in Figure 3 was chosen from a batch that were known to be worn but the exact nature of the wear was unknown at the time of selection. The authors thought that this would be a good test of the algorithm, since the errors generated at the semi-finish cut would be unknown beforehand to the operator. The resulting semi-finish part showed errors that indicated the worn spot lay at approximately a 45^0 angle on the tool tip. This is indicated by the errors plotted in Figure 4. The errors are shown as scaled bars that are called "whiskers". These errors are reflected in the probed errors reported in Table 1, which are errors normal to the surfaces averaged for four parts. The values are in micrometers. After applying the spline algorithm to model the errors on the front dome and small linear feature to its left on the semi-finish cut, the errors were considerably reduced on the finish cut as shown in Table 1, and the "whiskers" plot in Figure 5. Figure 6 shows the spline model of the errors on the leading two features of the part with a zero slope specified at the boundary point with the linear feature and a slightly greater than zero slope specified at the part zero point near the right hand corner of Figure 6.

6.0 Conclusions

Compensation of process related errors based on process-intermittent measurements and modeling has shown in the past that the procedure can correct errors on parts with linear features (see references [8] through [10]). This procedure has been extended to correcting errors on parts with arc features. In order to maintain path and slope continuity between tangent features, it was necessary to use splines with boundary constraints.

The splines with constraints have been demonstrated to adequately model machining errors probed on semi-finished parts. These models have successfully been used to reduce the part errors on the finish part to a small fraction of the original errors on the semi-finished part. In fact, the models allow the same tool that caused the errors to be used to correct them. Table 1 clearly shows the magnitude of reduction obtained.

The algorithm presented in this report allows the splines to be represented in the compact form of equation (29). This ensures that the resulting polynomials are of low order so that they can be used in error compensation during machining processes. That is, the error model evaluation time is not a significant factor to the error compensation process.

Figure 3: Part with Hemispherical Dome used to Test Algorithm

Mean Turning Center Errors for Four Parts				
Point Number	Nominal X Gauging Coordinate (mm)	Nominal Z Gauging Coordinate (mm)	Semi-finish Part Error (μm)	Finish Part Error (μm)
1	63.87211	-149.1631	90.297	-0.5715
2	60.46455	-143.0105	92.71	-0.889
3	57.05696	-136.858	92.964	-1.651
4	53.6494	-130.7054	94.1705	-0.762
5	50.24184	-124.5529	92.5195	-3.175
6	44.704	-106.934	12.446	-0.254
7	44.704	-92.0115	11.176	-0.508
8	44.704	-77.089	10.9855	-2.54
9	44.704	-62.1665	11.684	-3.048
10	44.704	-47.244	12.446	-2.286
11	43.434	-44.704	0	0
12	37.084	-38.354	6.731	-3.302
13	36.99706	-34.54598	21.717	7.8105
14	36.70041	-31.76392	31.623	3.2385
15	36.19487	-29.01213	44.577	-1.8415
16	35.48329	-26.30632	52.705	-4.7625
17	34.56976	-23.66183	62.8015	-2.54
18	33.45945	-21.09373	69.977	1.016
19	32.15869	-18.61668	76.2	1.778
20	30.67487	-16.24472	78.359	-0.762
21	29.01645	-13.99139	77.2795	-3.429
22	27.19286	-11.8695	76.7715	-1.397
23	25.2145	-9.891141	70.9295	-3.429
24	23.09261	-8.067548	63.5	-3.2385
25	20.83928	-6.409131	58.8645	-1.651
26	18.46732	-4.925314	51.435	-1.524
27	15.99027	-3.624555	45.2755	-3.1115
28	13.42217	-2.514244	36.83	-6.0325
29	10.77768	-1.600708	32.9565	-5.842
30	8.071866	-0.889127	27.432	1.651
31	5.320081	-0.383591	8.0645	-0.8255
32	2.538019	-0.086944	-18.733	-8.636

Table 1: Semi-finish and Finish Errors for Turned Part

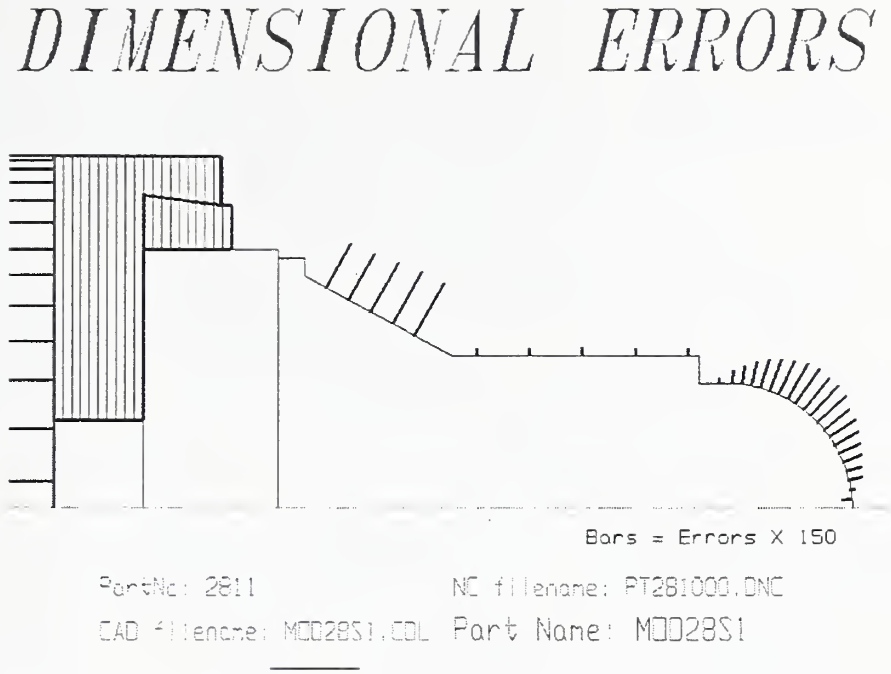

Figure 4: A "whiskers" plot of the Errors on the Semi-Finish Part

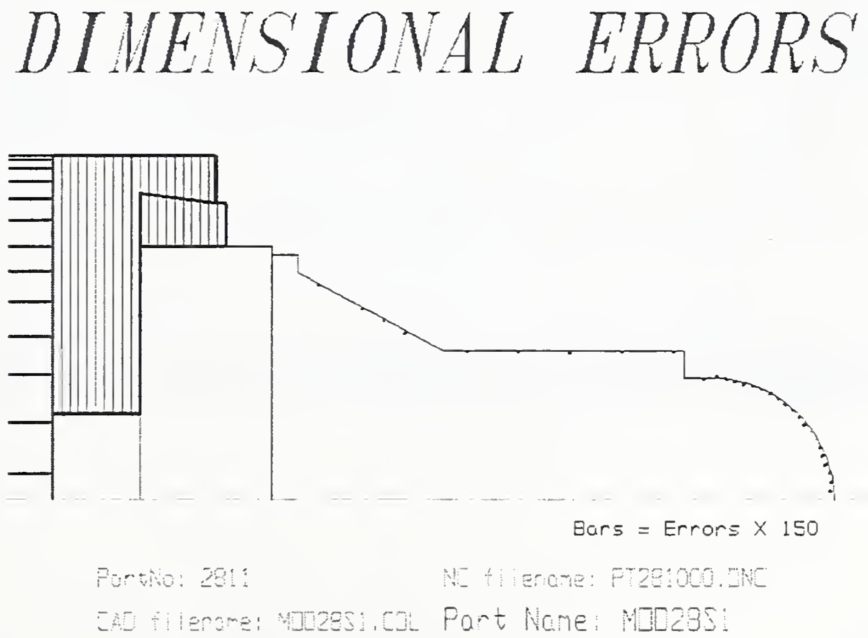

Figure 5: "Whiskers" Plot of the Reduced Errors on the Finish Part

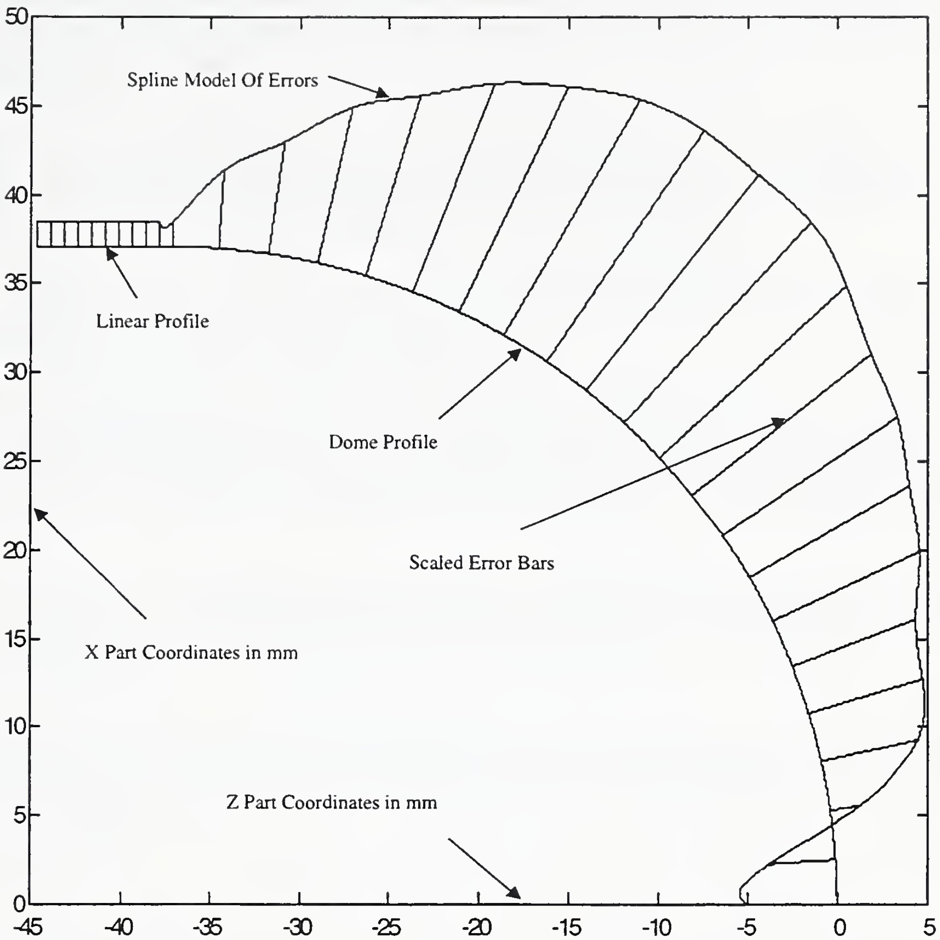

Figure 6: Scaled Semi-Finish Errors in Micrometers on the Leading Dome Profile and Linear Feature to the Left of the Dome Profile. See Points 12 through 32 in Table 1.

7.0 References

[1] Donmez, M.A., Yee, K.W., Neumann, D.H., Greenspan, L., Implementing Real-Time Control for Turning Center, NISTIR 4536, Progress Report of the Quality in Automation Project for FY90, M.A. Donmez (Ed.)., National Institute of Standards and Technology, Gaithersburg, MD: 25-39; 1991.

[2] Bandy, H.T., Process-Intermittent Error Compensation, NISTIR 4536, Progress Report of the Quality in Automation Project for FY90, M.A. Donmez (Ed.)., National Institute of Standards and Technology, Gaithersburg, MD: 25-39; 1991.

[3] Fan, K.C., Chow, Y.H., In-Process Dimensional Control of the Workpiece during Turning, Prec. Eng., Vol. 13, No 1, January 1991, 27-32.

[4] Yee, K. W., Real-Time Error Corrector and Process-Intermittent Probing, NISTIR 4322, Progress Report of the Quality in Automation Project for FY89, T. V. Vorburger, B. Scace (Ed.)., National Institute of Standards and Technology, Gaithersburg, MD: 9-31; 1990.

[5] Yee, K. W.; Gavin R. J., Implementing Fast Part Probing and Error Compensation on Machine Tools, NISTIR 4447, National Institute of Standards and Technology, Gaithersburg, MD; 1990.

[6] Gupta, S.K., Regli, W.C., Nau, D. S., Manufacturing Feature Instances: Which Ones to Recognize?, NISTIR 5655, National Institute of Standards and Technology, Gaithersburg, MD, 1995.

[7] De Boor, C. A Practical Guide to Splines. New York: Springer-Verlag; 1978.

[8] Bandy, H.T., Gilsinn, D. E., PIECS-A Software Program for Machine Tool Process-Intermittent Error Compensation, NISTIR 5797, National Institute of Standards and Technology, 1996.

[9] Bandy, H. T., Gilsinn, D. E., Data Management for Error Compensation and Process Control, SPIE Proceedings on Modeling, Simulation, and Control Technologies for Manufacturing, Vol. 2596, 1995, 114-123.

[10] Bandy, H. T.; Gilsinn, D. E. Compensation of Errors Detected by Process-Intermittent Gauging. Proceedings of the American Society for Precision Engineering 1995 Annual Meeting ; 12; 1995 October 15-20; Austin, TX.

[11] Yee, K. W.; Bandy, H. T.; Boudreaux, J.; Wilkin, N. D. Automated Compensation of Part Errors Determined by In-Process Gauging. Nat. Inst. of Stand. and Tech. (U.S.) NISTIR 4854, 1992 June.

[12] Smith, P. Splines as a Useful and Convenient Statistical Tool. The American Statistician. 33(2); 1979 May.

[13] Wold, S. Splines Functions in Data Analysis. Technometrics. 16(1); 1974 February.

[14] Montgomery, D. C.; Peck, E. A. Introduction to Linear Regression Analysis. New York: John Wiley & Sons, Inc. ; 1992.

[15] Lawson, C. L.; Hanson, R. J. Solving Least Squares Problems. Englewood Cliffs: Prentice- Hall, Inc.; 1974.