NISTIR 6722

# Updating a Turning Center Error Model by Singular Value Decomposition

David E.Gilsinn[a]
Herbert T. Bandy[b]
Alice V. Ling[c]

[a]U. S. DEPARTMENT OF COMMERCE
Technology Administration
Mathematical and Computational Sciences Division
National Institute of Standards
and Technology
Gaithersburg, MD 20899-8230

[b]U. S. DEPARTMENT OF COMMERCE
Technology Administration
Manufacturing Metrology Division
National Institute of Standards
and Technology
Gaithersburg, MD 20899-8230

[c]AFRL/DEX
3550 Aberdeen Ave. SE
Kirtland AFB, NM 87117-5776

1901-2001
NIST CENTENNIAL

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

NISTIR 6722

# Updating a Turning Center Error Model by Singular Value Decomposition

**David E.Gilsinn**[a]
**Herbert T. Bandy**[b]
**Alice V. Ling**[c]

[a]U. S. DEPARTMENT OF COMMERCE
Technology Administration
Mathematical and Computational Sciences Division
National Institute of Standards
and Technology
Gaithersburg, MD 20899-8230

[b]U. S. DEPARTMENT OF COMMERCE
Technology Administration
Manufacturing Metrology Division
National Institute of Standards
and Technology
Gaithersburg, MD 20899-8230

[c]AFRL/DEX
3550 Aberdeen Ave. SE
Kirtland AFB, NM 87117-5776

March 2001

U.S. DEPARTMENT OF COMMERCE
Donald L. Evans, Secretary

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Dr. Karen H. Brown, Acting Director

# Updating a Turning Center Error Model By Singular Value Decomposition

**David E. Gilsinn**
Mathematical and Computational Sciences Division

**Herbert T. Bandy**
Manufacturing Metrology Division

U.S. Department of Commerce
Technology Administration
National Institute of Standards and Technology
Gaithersburg, MD 20899

**Alice V. Ling**
AFRL/DEX
3550 Aberdeen Ave. SE
Kirtland AFB, NM 87117-5776

NISTIR 6722

Certain commercial software products are identified in this paper in order to adequately specify the computational procedures. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the software products identified are necessarily the best available for the purpose.

## Abstract

The precision of manufacturing using machine tools depends on the accuracy of the relative position of the cutting tool with respect to the workpiece. Kinematic modeling of machine tools is used to describe this relative position. This motion can be modeled by homogeneous coordinate transformation matrices composed of both rotational elements as well as positional offset elements of the associated coordinates. The rotation and translation components of the homogeneous transformations are considered to be functions of nominal tool position and machine temperatures. In general these functions are low order polynomials in terms of position and temperatures. The coefficients are usually calculated with least squares curve fitting techniques. The data for these fits are obtained by measuring actual coordinate positions and temperatures on the machine tool based upon desired programmed nominal coordinates. The process of measuring and modeling the errors of machine axis positions as functions of nominal positions and temperatures is referred to as machine tool characterization. The geometric-thermal models developed through machine tool characterization may not fully predict the errors encountered by a machine tool during machining. The data from machine characterization usually provides the structure to derive the basic form of the equations used to model the various error components used in the homogeneous matrices. This process of model updating involves determining the residual systematic errors of the machine tool and applying an algorithm to update the geometric-thermal model coefficients. The updating algorithm described in this report begins with adding perturbation terms to the characterization coefficients of the geometric-thermal model. These coefficients are estimated by an "inverse" process, using residual systematic errors, determined from part measurements on a coordinate measuring machine. The main tool used in identifying the perturbation terms is called a *generalized* or *pseudo inverse* matrix. This matrix is applied to the residual error vector to obtain a "best" approximate solution to the least squares problem.

**Key Words:** generalized inverse; geometric-thermal model; least squares; machine tool model; post-process analysis; pseudo inverse

## Table of Contents

## 0.0 Introduction

The precision of manufacturing using machine tools depends on the accuracy of the relative position of the cutting tool with respect to the workpiece. Kinematic modeling of machine tools is used to describe this relative position. In this paper a machine tool is assumed to be decomposable into basic rigid components. The motions of these components can be related to each other by means of relative motions of coordinate systems associated with each component. The machine tool is assumed to have a reference axis system (arbitrarily assigned by the user). From this axis system two chains of linked coordinates emanate. One connects the reference axis system with the tool point and the other with the workpiece. The motion of a coordinate system relative to another can be modeled by a homogeneous coordinate transformation matrix. This matrix is composed of both rotational elements as well as positional offset elements of the associated coordinates. The significance of homogeneous matrices is that a coordinate system defining the motion of a machine component can be linked to the reference coordinate system by a sequence of matrix multiplications. The errors in the location of the tool with respect to the workpiece are related to the relative errors in the locations of the coordinate frames in the axis chains. The rotation and translation components of the homogeneous matrices defining the relative motions of the linked coordinate systems are considered to be functions of nominal tool position and machine temperatures. In general these functions are low order polynomials in terms of position and temperatures. The coefficients are usually calculated with least squares curve fitting techniques. The data for these fits are obtained by measuring actual coordinate positions and temperatures on the machine tool based upon desired (CNC programmed) nominal coordinates. Modeling of machine tools in this manner has been used by many researchers, see for example references [1] to [11].

The process of measuring and modeling the errors of machine axis positions as functions of nominal positions and temperatures is referred to as machine tool characterization. One of the principal applications of kinematic modeling and machine tool characterization is to develop a geometric-thermal model that relates the machine tool error to the geometric imperfection and temperature variation in the machine tool. This model can also be used for real-time correction of machine tool errors as functions of nominal position and machine tool temperature profile. References [3] and [12] through [15] describe several machine tool characterization experiments.

The geometric-thermal models developed through machine tool characterization may not fully predict the errors encountered by a machine tool during machining. This is because the characterization is usually based on measurements that can be taken only when the machine tool is not operating. Many factors, such as coolant spray, load-induced deformation, and long term machine wear, combine to make such models ineffective. The data from machine characterization usually provides the structure to derive the basic form of the equations used to model the various error components used in the homogeneous matrices. The models developed through machine characterization can be considered as first order models that require coefficient adjustment and periodic coefficient updating in order to capture slowly changing process conditions in a production environment. This process of model updating involves determining the residual systematic errors of the machine tool and applying an algorithm to update the geometric-thermal model coefficients.

The updating algorithm described in this report begins with adding perturbation terms to the characterization coefficients of the geometric-thermal model. These coefficients are estimated by an "inverse" process, using residual systematic errors, determined from part measurements on a coordinate measuring machine (CMM). They are then combined with the existing model coefficients to compute the updated model coefficients. The "inverse" process used is a form of least-squares estimation that takes into account the possibility that the equations that arise in the usual least squares process may not be uniquely solvable and standard least squares procedures may fail.

A discussion of one implementation of Post-Process analysis can be found in [20] through [23] although few details of the computational inversion process are given. That implementation relies on inversion of data acquired on individual features. The current algorithm significantly differs from that implementation in that it uses data obtained from measurements of an entire part to determine residual errors rather than restricting the inversion process to a feature at a time. Using data from an entire part provides an averaged set of coefficients applicable to all features on the part. Inverting, based on fitting separate feature measurements, adjusts the same coefficients in the geometric-thermal model differently and does not provide a single consistent adjustment of all parameters that apply to the entire part. Once all parameters are uniformly adjusted, errors on individual features can easily be determined.

The main tool used in identifying the perturbation terms is called a *generalized* or *pseudo inverse* matrix. This matrix is applied to the residual error vector to obtain a "best" approximate solution to the least squares problem. The term "best" will be described in Appendix E. The essential tool in forming the generalized inverse is a matrix theorem which proves that any given matrix can be decomposed into an associated diagonal matrix, some of whose elements might be zero. The diagonal elements are called *singular values* and indicate how many perturbation coefficients are significant. A full discussion of the numerical aspects related to the singular value decomposition and pseudo inverse generation are given in Appendix E and in reference [24].

This report describes the algorithmic details of the post-process analysis procedure and presents an application to one part produced on a turning center. The authors are aware that the application of the method to one part and, in fact, a reasonably simple part does not constitute an adequate test of the procedure. However, budgetary constraints at the time of the study limited the current work. The authors, however, felt that it was worthwhile publishing the current results as a stimulus for further research. Future testing of the algorithm is planned as budgets allow. These new tests will highlight various parts designed to utilize the extent of the workspace and temperature ranges experienced by the turning center. The object of the tests is to determine whether there is a fixed set of coefficients for the geometric-thermal model that apply to the entire workspace or whether the workspace needs to be divided into sub-areas with separate sets of coefficients applicable to each sub-area. They will also study the robustness of the compensated parameters to perturbations of the data points upon which the model updates were based.

The report is divided into six sections. Section 2 presents the error equations for the machine tool (in this report a turning center was used). Section 3 describes the development of the perturbation equations. Section 3 describes the formation of the system matrix. Section 4 describes the application of the method to estimate the perturbed machine tool model coefficients by using a machined part on the turning center. Section 5 concludes with a discussion of the results. Section 6 includes the references.

## 1.0 Kinematic Model for a Turning Center

The turning center that is modeled in this section is shown in Figure 1. Although 40 thermocouples have been identified numerically from 0 to 39 (see Appendix A.1, Table A1), only 36 were actually used to measure temperatures since 3 through 6 were not attached to the machine tool during testing. This figure also shows the locations of various thermocouples used to model the individual component errors that are included in the complete kinematic model.

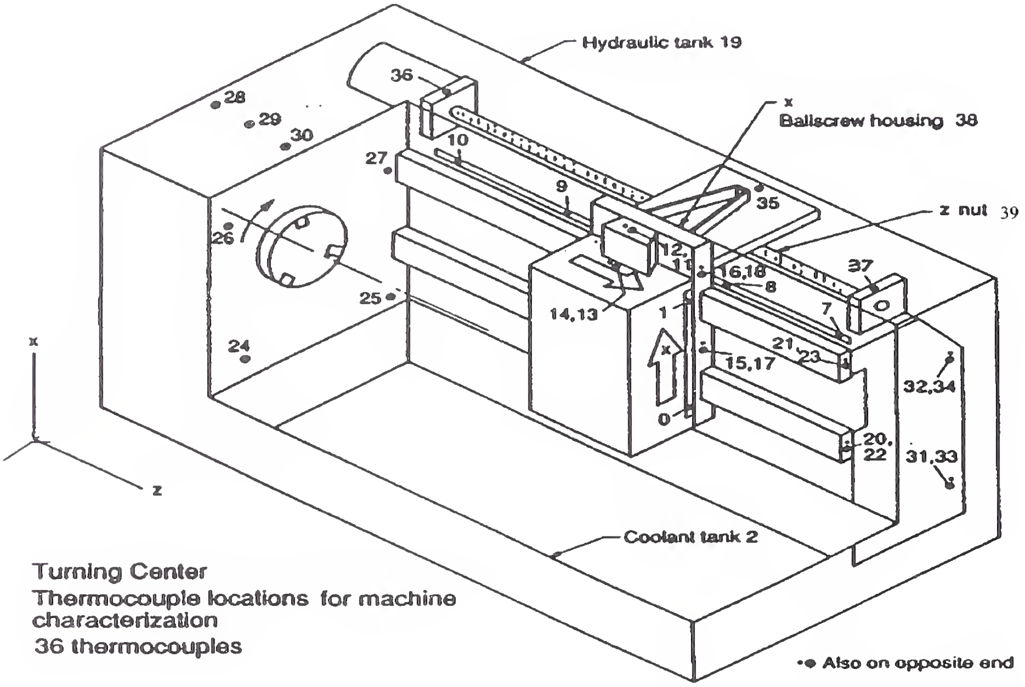

**Figure 1: Turning Center with Thermocouple Locations**

The formal development of the geometric-thermal error equations from the kinematic model is given in Appendix A.1 and only the results are given below. The definitions of the terms in the model are also given in Appendix A.1. The methods used to develop these equations are well known and can be found in [3-5]. The coordinate system for the turning center is shown in Figure 1.

The modeled displacement errors of the tool from the workpiece are given by:

$$\begin{aligned}
E_x &= \varepsilon_y(s,T)*Z_5 + [\varepsilon_y(z,T) + \varepsilon_y(x,T) + \varepsilon_y(t,T) - \varepsilon_y(s,T)]*Z_4 \\
&+ [\varepsilon_y(z,T) - \varepsilon_y(s,T)]*Z_2 - \varepsilon_y(s,T)*(Z_1 + z) + \delta_x(z,T) + \delta_x(x,T) \\
&- \delta_x(w,T) + \delta_x(t,T) - \delta_x(s,T) + \delta_x(c,T)
\end{aligned}$$

$$\begin{aligned}
E_y &= -\varepsilon_x(s,T)*Z_5 + [-\varepsilon_x(z,T) - \varepsilon_x(x,T) - \varepsilon_x(t,T) + \varepsilon_x(s,T)]*Z_4 \\
&+ [-\varepsilon_x(z,T) + \varepsilon_x(s,T)]*Z_2 + \varepsilon_x(s,T)*(Z_1 + z) + [\varepsilon_z(z,T) + \varepsilon_z(x,T) + \varepsilon_z(t,T)]*X_4 \quad (1) \\
&+ \varepsilon_z(z,T)*(X_2 + x) + \delta_y(z,T) + \delta_y(x,T) - \delta_y(w,T) + \delta_y(t,T) - \delta_y(s,T) + \delta_y(c,T)
\end{aligned}$$

$$\begin{aligned}
E_z &= -\varepsilon_y(s,T)*X_5 + [-\varepsilon_y(z,T) - \varepsilon_y(x,T) - \varepsilon_y(t,T) + \varepsilon_y(s,T)]*X_4 \\
&+ [-\varepsilon_y(z,T) + \varepsilon_y(s,T)]*(X_2 + x) + \varepsilon_y(s,T)*X_1 + \delta_z(z,T) + \delta_z(x,T) \\
&- \delta_z(w,T) + \delta_z(t,T) - \delta_z(s,T) + \delta_z(c,T)
\end{aligned}$$

These equations give the generic error formulas for a turning center with its absolute reference axis system and machine coordinate system assumed aligned at "cold start" in the "home" position. The constants, $X_1$, $Z_1$, $X_2$, $Z_2$, $X_4$, $Z_4$, $X_5$, $Z_5$, are relative offsets of the machine component coordinate systems from the absolute reference axis system. Note that $X_3$, $Z_3$ are found to be non-significant and thus do not appear in equation (1). The significant offset constants are given in Table A.2 of Appendix A.1. Many of the components are deemed non-significant on a turning center (since the turning center is a 2-axis machine) so that only nine errors are included in the final model for the turning center to yield the error equations

$$\begin{aligned}
E_x &= [\varepsilon_y(z,T) + \varepsilon_y(x,T)]*Z_4 + \varepsilon_y(z,T)*Z_2 + [\delta_x'(z,T) + \alpha_p(T)*\Delta z] + \delta_x(x,T) \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (2) \\
E_z &= -[\varepsilon_y(z,T) + \varepsilon_y(x,T)]*X_4 - \varepsilon_y(z,T)*(X_2 + x) + \delta_z(z,T) \\
&+ [\delta_z'(x,T) + \alpha_o(T)*\Delta x] - \delta_z(s,T)
\end{aligned}$$

Appendix A.2 lists the program used to symbolically generate equations (1) and (2) using the symbol manipulator program MACSYMA. Note that the final error compensation model for the turning center requires only two coordinate system offsets. These offsets are given by

$(Z_4, X_4)$ - Tool offset from the turret.
$(Z_2, X_2)$ - Cross slide offset from the carriage axis.

The notation $T$ represents the vector of all thermocouple temperatures given by $T = (T_0, \cdots, T_{39})$.

The following error components on the turning center are the characterization equations and the coefficients used in the current study are given in Appendix B.1. The rotational displacement of all points on the turning center have a yaw component that is a function of temperature

$$Mach\_Yaw = A_{10,1}((T_{10} + T_{21})/2) + A_{10,2} \qquad (3)$$

X yaw is

$$\varepsilon_y(x,T) = A_{11}x^2 + A_{12}x + (A_{13} + Mach\_Yaw) \qquad (4)$$

Z yaw is

$$\varepsilon_y(z,T) = A_{21}z^2 + A_{22}z + (A_{23} + Mach\_Yaw) \qquad (5)$$

The X straightness error with Z motion is

$$\delta_x{}'(z,T) = A_{31}z^2 + A_{22}Z + (A_{33}T_{39} + A_{34}) \qquad (6)$$

The X-axis displacement is

$$\delta_x(x,T) = A_{41}x^2 + (A_{42} + A_{43}T_{35})x + (A_{44}T_{35} + A_{45}) \quad (7)$$

The Z displacement is

$$\delta_z(z,T) = A_{51}z^2 + A_{52}z + (A_{53}T_{12} + A_{54}) \qquad (8)$$

The Z straightness of the X axis is

$$\delta_z{}'(x,T) = A_{61}x^2 + A_{62}x + (A_{63}T_{30} + A_{64}) \qquad (9)$$

Spindle growth is

$$\delta_z(s,T) = A_{71}T_{27}^2 + A_{72}T_{27} + A_{73}T_{19,0} + A_{74} \qquad (10)$$

The temperature of thermocouple 19 is taken at the initial temperature before the machine starts to warm up.

The squareness of the X-axis with respect to the spindle axis is

$$\alpha_o(T) = A_{81}T_0 + A_{82} \quad (11)$$

The parallelism error is

$$\alpha_p(T) = A_{91}T_{19,0}^2 + A_{92}T_{19,0} + A_{93}T_{31} + A_{94} \qquad (12)$$

## 2.0 Forming the Perturbation Equations

In order to adjust the coefficients of (2) we form a set of perturbed equations. These equations are built up from perturbing the coefficients of the component error equations. These are

$$
\begin{aligned}
\varepsilon_y(x,T)+\Delta(\varepsilon_y(x,T)) &= A_{11}(1+\Delta A_{11})x^2 + A_{12}(1+\Delta A_{12})x + A_{13}(1+\Delta A_{13}) \\
&\quad + A_{10,1}(1+\Delta A_{10,1})((T_{10}+T_{21})/2) + A_{10,2}(1+\Delta A_{10,2}) \\
\varepsilon_y(z,T)+\Delta(\varepsilon_y(z,T)) &= A_{21}(1+\Delta A_{21})z^2 + A_{22}(1+\Delta A_{22})z + A_{23}(1+\Delta A_{23}) \\
&\quad + A_{10,1}(1+\Delta A_{10,1})((T_{10}+T_{21})/2) + A_{10,2}(1+\Delta A_{10,2}) \\
\delta_x'(z,T)+\Delta(\delta_x'(z,T)) &= A_{31}(1+\Delta A_{31})z^2 + A_{32}(1+\Delta A_{32})z + A_{33}(1+\Delta A_{33})T_{39} \\
&\quad + A_{34}(1+\Delta A_{34}) \\
\delta_x(x,T)+\Delta(\delta_x(x,T)) &= A_{41}(1+\Delta A_{41})x^2 + A_{42}(1+\Delta A_{42})x + A_{43}(1+\Delta A_{43})T_{35}x \\
&\quad + A_{44}(1+\Delta A_{44})T_{35} + A_{45}(1+\Delta A_{45}) \\
\delta_z(z,T)+\Delta(\delta_z(z,T)) &= A_{51}(1+\Delta A_{51})z^2 + A_{52}(1+\Delta A_{52})z + A_{53}(1+\Delta A_{53})T_{12} \\
&\quad + A_{54}(1+\Delta A_{54}) \\
\delta_z'(x,T)+\Delta(\delta_z'(x,T)) &= A_{61}(1+\Delta A_{61})x^2 + A_{62}(1+\Delta A_{62})x + A_{63}(1+\Delta A_{63})T_{30} \\
&\quad + A_{64}(1+\Delta A_{64}) \\
\delta_z(s,T)+\Delta(\delta_z(s,T)) &= A_{71}(1+\Delta A_{71})T_{27}^2 + A_{72}(1+\Delta A_{72})T_{27} + A_{73}(1+\Delta A_{73})T_{19,0} \\
&\quad + A_{74}(1+\Delta A_{74}) \\
\alpha_o(T)+\Delta(\alpha_o(T)) &= A_{81}(1+\Delta A_{81})T_0 + A_{82}(1+\Delta A_{82}) \\
\alpha_p(T)+\Delta(\alpha_p(T)) &= A_{91}(1+\Delta A_{91})T_{19,0}^2 + A_{92}(1+\Delta A_{92})T_{19,0} + A_{93}(1+\Delta A_{93})T_{31} \qquad (15) \\
&\quad + A_{94}(1+\Delta A_{94})
\end{aligned}
$$

Note that the perturbations are structured in such a way that the characterization coefficients are carried along into the difference terms. This was done in order to carry the influence of the characterization coefficients along during the fitting process. Now, to obtain the perturbation terms subtract equations (4) through (12) from (15) to get

$$
\begin{aligned}
\Delta(\varepsilon_y(x,T)) &= (A_{11}x^2)\Delta A_{11} + (A_{12}x)\Delta A_{12} + (A_{13})\Delta A_{13} \\
&\quad + (A_{10,1})((T_{10} + T_{21})/2)\Delta A_{10,1} + (A_{10,2})\Delta A_{10,2} \\
\Delta(\varepsilon_y(z,T)) &= (A_{21}z^2)\Delta A_{21} + (A_{22}z)\Delta A_{22} + (A_{23})\Delta A_{23} \\
&\quad + (A_{10,1})((T_{10} + T_{21})/2)\Delta A_{10,1} + (A_{10,2})\Delta A_{10,2} \\
\Delta(\delta_x{}'(z,T)) &= (A_{31}z^2)\Delta A_{31} + (A_{32}z)\Delta A_{32} + (A_{33}T_{39})\Delta A_{33} \\
&\quad + (A_{34})\Delta A_{34} \\
\Delta(\delta_x(x,T)) &= (A_{41}x^2)\Delta A_{41} + (A_{42}x)\Delta A_{42} + (A_{43}T_{35}x)\Delta A_{43} \\
&\quad + (A_{44}T_{35})\Delta A_{44} + (A_{45})\Delta A_{45} \\
\Delta(\delta_z(z,T)) &= (A_{51}z^2)\Delta A_{51} + (A_{52}z)\Delta A_{52} + (A_{53}T_{12})\Delta A_{53} \\
&\quad + (A_{54})\Delta A_{54} \\
\Delta(\delta_z{}'(x,T)) &= (A_{61}x^2)\Delta A_{61} + (A_{62}x)\Delta A_{62} + (A_{63}T_{30})\Delta A_{63} \\
&\quad + (A_{64})\Delta A_{64} \\
\Delta(\delta_z(s,T)) &= (A_{71}T_{27}^2)\Delta A_{71} + (A_{72}T_{27})\Delta A_{72} + (A_{73}T_{19,0})\Delta A_{73} \\
&\quad + (A_{74})\Delta A_{74} \\
\Delta(\alpha_o(T)) &= (A_{81}T_0)\Delta A_{81} + (A_{82})\Delta A_{82} \\
\Delta(\alpha_p(T)) &= (A_{91}T_{19,0}^2)\Delta A_{91} + (A_{92}T_{19,0})\Delta A_{92} + (A_{93}T_{31})\Delta A_{93} \\
&\quad + (A_{94})\Delta A_{94}
\end{aligned}
\tag{16}
$$

The perturbation form of the kinematic model becomes

$$
\begin{aligned}
&\Delta(E_x) = [\Delta(\varepsilon_y(z)) + \Delta(\varepsilon_y(x))] * Z_4 + \Delta(\varepsilon_y(z)) * Z_2 \\
&+ [\Delta(\delta_x{}'(z)) + \Delta(\alpha_p) * \Delta z] + \Delta(\delta_x(x)) \\
\\
&\Delta(E_z) = -[\Delta(\varepsilon_y(z)) + \Delta(\varepsilon_y(x))] * X_4 - \Delta(\varepsilon_y(z)) * (X_2 + x) \\
&+ \Delta(\delta_z(z)) + [\Delta(\delta_z{}'(x)) + \Delta(\alpha_o) * \Delta x] - \Delta(\delta_z(s))
\end{aligned}
\tag{17}
$$

In order to relate the perturbed residual position errors in (17) to the perturbed error component coefficients substitute (16) into (17). Now, the problem becomes one of determining the perturbation coefficients from the observed (measured) residual error vector $(\Delta(E_x), \Delta(E_y))$. To do this we create a system of equations of the following form

$$
\begin{aligned}
\Delta(E_x) = {} & C_{11}(x,z)d_{11} + C_{12}(x,z)d_{12} + C_{13}(x,z)d_{13} \\
& + C_{21}(x,z)d_{21} + C_{22}(x,z)d_{22} + C_{23}(x,z)d_{23} \\
& + C_{31}(x,z)d_{31} + C_{32}(x,z)d_{32} + C_{33}(x,z)d_{33} + C_{34}(x,z)d_{34} \\
& + C_{41}(x,z)d_{41} + C_{42}(x,z)d_{42} + C_{43}(x,z)d_{43} + C_{44}(x,z)d_{44} + C_{45}(x,z)d_{45} \\
& + C_{91}(x,z)d_{91} + C_{92}(x,z)d_{92} + C_{93}(x,z)d_{93} + C_{94}(x,z)d_{94} \\
& + C_{10,1}(x,z)d_{10,1} + C_{10,2}(x,z)d_{10,2} \\
\\
\Delta(E_z) = {} & D_{11}(x,z)d_{11} + D_{12}(x,z)d_{12} + D_{13}(x,z)d_{13} \\
& + D_{21}(x,z)d_{21} + D_{22}(x,z)d_{22} + D_{23}(x,z)d_{23} \\
& + D_{51}(x,z)d_{51} + D_{52}(x,z)d_{52} + D_{53}(x,z)d_{53} + D_{54}(x,z)d_{54} \\
& + D_{61}(x,z)d_{61} + D_{62}(x,z)d_{62} + D_{63}(x,z)d_{63} + D_{64}(x,z)d_{64} \\
& + D_{71}(x,z)d_{71} + D_{72}(x,z)d_{72} + D_{73}(x,z)d_{73} + D_{74}(x,z)d_{74} \\
& + D_{81}(x,z)d_{81} + D_{82}(x,z)d_{82} \\
& + D_{10,1}(x,z)d_{10,1} + D_{10,2}(x,z)d_{10,2}
\end{aligned}
\tag{18}
$$

where the $C_{ij}(x,z), D_{ij}(x,z), d_{ij}$ terms will be defined below.

When (16) is substituted into (17) the unknown coefficients are $\Delta A_{ij}$. From the numerical point of view it will be necessary later to scale these unknowns in equation (16) in order to control the magnitude of the elements in the resulting matrix that arises in the fitting process. The scaled unknowns are the $d_{ij}$ coefficients in (18). In order to construct them we will introduce a scaling array $s(i),\ i = 1,\cdots,35$ and redefine the perturbation terms so that

$$
\begin{aligned}
\Delta A_{11} &= s(1) * d_{11} \\
\Delta A_{12} &= s(2) * d_{12} \\
\Delta A_{13} &= s(3) * d_{13} \\
\Delta A_{21} &= s(4) * d_{21} \\
\Delta A_{22} &= s(5) * d_{22} \\
\Delta A_{23} &= s(6) * d_{23} \\
\Delta A_{31} &= s(7) * d_{31} \\
\Delta A_{32} &= s(8) * d_{32} \\
\Delta A_{33} &= s(9) * d_{33} \\
\Delta A_{34} &= s(10) * d_{34} \\
\Delta A_{41} &= s(11) * d_{41} \\
\Delta A_{42} &= s(12) * d_{42} \\
\Delta A_{43} &= s(13) * d_{43} \\
\Delta A_{44} &= s(14) * d_{44} \\
\Delta A_{45} &= s(15) * d_{45} \\
\Delta A_{51} &= s(16) * d_{51} \\
\Delta A_{52} &= s(17) * d_{52} \\
\Delta A_{53} &= s(18) * d_{53} \\
\Delta A_{54} &= s(19) * d_{54} \\
\Delta A_{61} &= s(20) * d_{61} \\
\Delta A_{62} &= s(21) * d_{62} \\
\Delta A_{63} &= s(22) * d_{63} \\
\Delta A_{64} &= s(23) * d_{64} \\
\Delta A_{71} &= s(24) * d_{71} \\
\Delta A_{72} &= s(25) * d_{72} \\
\Delta A_{73} &= s(26) * d_{73} \\
\Delta A_{74} &= s(27) * d_{74} \\
\Delta A_{81} &= s(28) * d_{81} \\
\Delta A_{82} &= s(29) * d_{82} \\
\Delta A_{91} &= s(30) * d_{91} \\
\Delta A_{92} &= s(31) * d_{92} \\
\Delta A_{93} &= s(32) * d_{93} \\
\Delta A_{94} &= s(33) * d_{94} \\
\Delta A_{10,1} &= s(34) * d_{10,1} \\
\Delta A_{10,2} &= s(35) * d_{10,2}
\end{aligned}
\tag{19}
$$

where $d_{ij}$ are new perturbation coefficients. The scaling is necessary in order to guarantee that terms in the fitting process matrix are all approximately the same order of magnitude and are not too large. The scaling array is determined in such a way that the maximum value in each column of the matrix $J$ is one. This is done for each column by first finding the maximum of the absolute values of the elements in each column. If the maximum is zero then the scale for that column is set to one. Otherwise the scale is set to the reciprocal of the maximum. For a discussion of scaling and its application to problems of inversion see [25].

Returning to the substitution of (16) into (17) we further introduce (18) into (17) and rearrange terms. Now, define the following coefficients

$$
\begin{aligned}
C_{11}(x,z) &= \left(A_{11} * s(1) * Z_4\right)x^2 \\
C_{12}(x,z) &= \left(A_{12} * s(2) * Z_4\right)x \\
C_{13}(x,z) &= A_{13} * s(3) * Z_4 \\
C_{21}(x,z) &= \left(A_{21} * s(4) * \left(Z_4 + Z_2\right)\right)z^2 \\
C_{22}(x,z) &= \left(A_{22} * s(5) * \left(Z_4 + Z_2\right)\right)z \\
C_{23}(x,z) &= A_{23} * s(6) * \left(Z_4 + Z_2\right) \\
C_{31}(x,z) &= A_{31} * s(7) * z^2 \\
C_{32}(x,z) &= A_{32} * s(8) * z \\
C_{33}(x,z) &= A_{33} * T_{39} * s(9) \\
C_{34}(x,z) &= A_{34} * s(10) \\
C_{41}(x,z) &= A_{41} * s(11) * x^2 \\
C_{42}(x,z) &= A_{42} * s(12) * x \\
C_{43}(x,z) &= A_{43} * T_{35} * s(13) * x \\
C_{44}(x,z) &= A_{44} * T_{35} * s(14) \\
C_{45}(x,z) &= A_{45} * s(15) \\
C_{91}(x,z) &= A_{91} * T_{19.0}^2 * s(30) * \Delta z \\
C_{92}(x,z) &= A_{92} * T_{19.0} * s(31) * \Delta z \\
C_{93}(x,z) &= A_{93} * T_{31} * s(32) * \Delta z \\
C_{94}(x,z) &= A_{94} * s(33) * \Delta z \\
C_{10.1}(x,z) &= A_{10.1} * (2Z_4 + Z_2)((T_{10} + T_{21})/2) * s(34) \\
C_{10.2}(x,z) &= A_{10.2} * (2Z_4 + Z_2) * s(35)
\end{aligned}
\qquad (20)
$$

and

$$
\begin{aligned}
&D_{11}(x,z) = -\left(A_{11} * s(1) * X_4\right)x^2 \\
&D_{12}(x,z) = -\left(A_{12} * s(2) * X_4\right)x \\
&D_{13}(x,z) = -A_{13} * s(3) * X_4 \\
&D_{21}(x,z) = -\left(A_{21} * s(4) * \left(X_4 + X_2 + x\right)\right)z^2 \\
&D_{22}(x,z) = -\left(A_{22} * s(5) * \left(X_4 + X_2 + x\right)\right)z \\
&D_{23}(x,z) = -A_{23} * s(6) * \left(X_4 + X_2 + x\right) \\
&D_{51}(x,z) = A_{51} * s(16) * z^2 \\
&D_{52}(x,z) = A_{52} * s(17) * z \\
&D_{53}(x,z) = A_{53} * T_{12} * s(18) \\
&D_{54}(x,z) = A_{54} * s(19) \\
&D_{61}(x,z) = A_{61} * s(20) * x^2 \\
&D_{62}(x,z) = A_{62} * s(21) * x \\
&D_{63}(x,z) = A_{63} * T_{30} * s(22) \\
&D_{64}(x,z) = A_{64} * s(23) \\
&D_{71}(x,z) = -A_{71} * T_{27}^2 * s(24) \\
&D_{72}(x,z) = -A_{72} * T_{27} * s(25) \\
&D_{73}(x,z) = -A_{73} * T_{19,0} * s(26) \\
&D_{74}(x,z) = -A_{74} * s(27) \\
&D_{81}(x,z) = A_{81} * T_0 * s(28) * \Delta x \\
&D_{82}(x,z) = A_{82} * s(29) * \Delta x \\
&D_{10,1}(x,z) = -A_{10,1} * (X_2 + 2X_4 + x) * ((T_{10} + T_{21})/2) * s(34) \\
&D_{10,2}(x,z) = -A_{10,2} * (X_2 + 2X_4 + x) * s(35)
\end{aligned}
\tag{21}
$$

By substituting equations (20) and (21) into (17) we get (18), which is the system we sought to form. System (18) itself cannot be directly solved for the unknowns, but if we introduce the residual errors measured at a set of spatial points on the part then it will be feasible to solve for the unknowns in a least squares manner. In the next section we form the system of equations needed to solve for the unknowns. There are 35 unknowns, each associated with a column of the system matrix. However, not all of the coefficients appear in both (20) and (21) simultaneously. These missing coefficients lead to blocks of zeroes in the matrix defined in the next section.

## 3.0 Forming and Solving the System Matrix

We form the system of equations to solve for the unknown $d_{ij}$'s in equation (18) as follows. Let there be n points on the part at which residual errors are to be computed from CMM measurements. Denote the points by $(x_i, z_i)$ for $i = 1, \ldots, n$. Then set

$$\begin{aligned}
J_{i,1} &= C_{11}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,2} &= C_{12}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,3} &= C_{13}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,4} &= C_{21}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,5} &= C_{22}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,6} &= C_{23}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,7} &= C_{31}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,8} &= C_{32}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,9} &= C_{33}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,10} &= C_{34}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,11} &= C_{41}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,12} &= C_{42}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,13} &= C_{43}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,14} &= C_{44}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,15} &= C_{45}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,k} &= 0, k = 16, \cdots, 29 \\
J_{i,30} &= C_{91}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,31} &= C_{92}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,32} &= C_{93}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,33} &= C_{94}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,34} &= C_{10,1}(x_{n+1-i}, z_{n+1-i}) \\
J_{i,35} &= C_{10,2}(x_{n+1-i}, z_{n+1-i})
\end{aligned} \tag{22}$$

In $C_{91}$ to $C_{94}$ the value of $\Delta z$ is computed as $\Delta z = z_{n+1-i} - z_{n-i}$ for $i = 1, \cdots, n-1$. For $i = n,\ \ \Delta z = z_2 - z_1$.

For $i = n+1, \ldots, 2n$ the matrix elements are written as

$$\begin{aligned}
J_{i,1} &= D_{11}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,2} &= D_{12}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,3} &= D_{13}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,4} &= D_{21}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,5} &= D_{22}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,6} &= D_{23}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,k} &= 0, k = 7,\cdots,15 \\
J_{i,16} &= D_{51}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,17} &= D_{52}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,18} &= D_{53}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,19} &= D_{54}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,20} &= D_{61}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,21} &= D_{62}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,22} &= D_{63}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,23} &= D_{64}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,24} &= D_{71}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,25} &= D_{72}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,26} &= D_{73}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,27} &= D_{74}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,28} &= D_{81}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,29} &= D_{82}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,k} &= 0, \quad k = 30,\cdots,33 \\
J_{i,34} &= D_{10,1}(x_{2n+1-i}, z_{2n+1-i}) \\
J_{i,35} &= D_{10,2}(x_{2n+1-i}, z_{2n+1-i})
\end{aligned} \tag{23}$$

In $D_{81}$ and $D_{82}$ the value of $\Delta x$ is computed as $\Delta x = x_{2n+1-i} - x_{2n-i}$ for $i = n+1,\cdots,2n-1$. For $i = 2n$, $\Delta x = x_2 - x_1$.

Finally, for ease of notation, set

$$\begin{aligned}
V(i) &= \Delta E_x^{n+1-i}, i = 1,\cdots,n \\
V(i) &= \Delta E_z^{2n+1-i}, i = n+1,\cdots,2n
\end{aligned} \tag{24}$$

and

$$\begin{aligned}
U(1) &= d_{11} \\
U(2) &= d_{12} \\
U(3) &= d_{13} \\
U(4) &= d_{21} \\
U(5) &= d_{22} \\
U(6) &= d_{23} \\
U(7) &= d_{31} \\
U(8) &= d_{32} \\
U(9) &= d_{33} \\
U(10) &= d_{34} \\
U(11) &= d_{41} \\
U(12) &= d_{42} \\
U(13) &= d_{43} \\
U(14) &= d_{44} \\
U(15) &= d_{45} \\
U(16) &= d_{51} \\
U(17) &= d_{52} \\
U(18) &= d_{53} \\
U(19) &= d_{54} \\
U(20) &= d_{61} \\
U(21) &= d_{62} \\
U(22) &= d_{63} \\
U(23) &= d_{64} \\
U(24) &= d_{71} \\
U(25) &= d_{72} \\
U(26) &= d_{73} \\
U(27) &= d_{74} \\
U(28) &= d_{81} \\
U(29) &= d_{82} \\
U(30) &= d_{91} \\
U(31) &= d_{92} \\
U(32) &= d_{93} \\
U(33) &= d_{94} \\
U(34) &= d_{10,1} \\
U(35) &= d_{10,2}
\end{aligned} \tag{25}$$

We now have the notation set up to form the system

$$JU = V \qquad (26)$$

where $J$ is a matrix with 2n rows, since there are $n$ probe points each with an x and a z error, by 35 columns, one for each scaled perturbation coefficient. Note that if any residual error in equation (24) is zero then the associated row of $J$ is set to zero.

If $J$ were a square matrix and non singular then the solution would be simply

$$U = J^{-1}V\,, \qquad (27)$$

but in general J is an overdetermined system with more rows than columns. Furthermore, since many of the columns might be dependent on each other a standard least squares approach will likely fail. For this reason we apply a matrix called the generalized or pseudo inverse and compute

$$U = J^{+}V. \qquad (28)$$

The details involved with computing the pseudo inverse are given in Appendix E and in [24].

Once U has been computed the $\Delta A_{ij}$ coefficients can be determined from (19) and the updated coefficients, given by

$$A_{ij}(1 + \Delta A_{ij})\,. \qquad (29)$$

## 4.0 Application

In this section an application of the inversion process described in the previous sections will be given. The part shown in Figure 2 was machined. It consisted of two concentric cylinders. The smaller cylinder has a nominal diameter of 50.8 mm (2 in), and the larger one has a nominal diameter of 152.4 mm (6 in). To compute the process errors, the part was probed with a touch trigger probe at 35 points while still secured on the machine tool. For a discussion of on-machine probing see Bandy and Gilsinn [17].

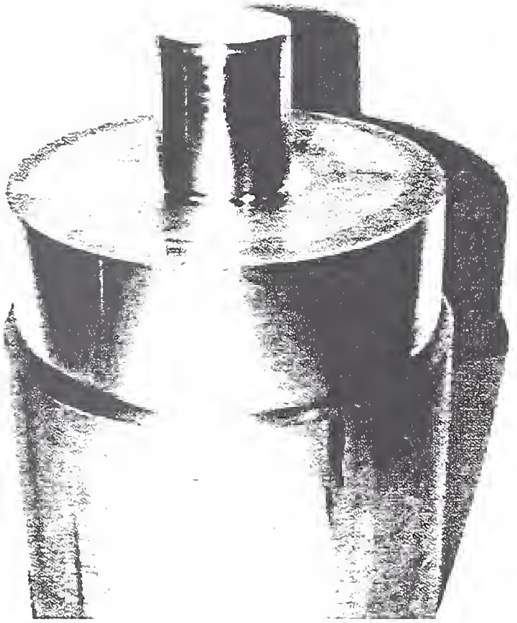

**Figure 2: Two Cylinder Turned Part**

In order to use on-machine touch trigger probe readings in determining machine tool errors, various coordinate systems have to be linked. For example, the turning center error models are based on calibrated error component equations that describe error motions at points in the reference coordinate system of the turning center. When parts are turned, however, points on the part are identified relative to a part coordinate system. Therefore the part coordinate zero needs to be located relative to the turning center reference coordinate system. In Figure 2 the part coordinate zero is set as the center point of the top face of the small cylinder. The positive Z-axis is taken vertically and the positive X-axis is taken to the right. In order to link the part zero point to the turning center reference coordinate system two sets of data are read. First the probe offsets in the X and Z directions are obtained from a database file. These offsets are distances from a point on the turning center tool turret called the tool setting point. As the turret moves the location of the tool setting point with respect to the reference coordinate system can be queried so that when the touch trigger probe is brought to the part zero point the location of the part zero relative to the reference coordinate system can be computed. As a result any part coordinate can be identified with a reference coordinate. Therefore, when points on a part are identified by their part coordinates, error estimates at these points can then be computed by identifying the part coordinates with their machine reference coordinates. For a further discussion of computing machine offsets see Appendix C.

A coordinate measuring machine (CMM) is used to measure the part in Figure 2 at the same part coordinates that were probed on the turning center. This is done by establishing a working reference plane on the CMM along the top surface of the small cylinder and setting the reference plane zero at the same point as the part zero is set on the turning center. Then, the 35 points probed on the turning center are probed at the same coordinates on the CMM. Error measurements are made normal to the surface features probed. Realizing that there is statistical variation in the measurements, the CMM is programmed to measure the 35 points along the surface at 8 angles around the part in Figure 2. The error measurements at these equally spaced angles are then averaged.

The estimation of the new coefficients for the component error equations in the turning center error model proceeds by: 1) reading the old coefficients from a database, 2) reading the thermocouple temperatures on the turning center when the part was manufactured, 3) reading the initial turning center temperatures before machining, 4) reading the nominal points probed on the part and the averaged CMM error measurements. The turning center errors are computed by the methods of Appendix D and the system matrix $J$ is formed as in Section 4. Next, the predicted turning center errors are computed using the old coefficients and the difference between the CMM measured errors and the predicted turning center errors, as in equation (D5). This produces the residual errors at the probed points. The pseudoinverse of the system matrix, $J^+$, is applied to the residual errors to produce the scaled correction terms $d_{ij}$. The coefficient correction terms are computed by rescaling the terms in (19). Finally the new coefficients are computed by adding one to the rescaled correction terms and then multiplying by the old coefficients as in equation (29).

The computed results are shown below in figures 3 and 4. Of the 35 points chosen on the surface for analysis, eight were selected on the top surface of the small cylinder and nine each were selected on the small cylinder side, the large cylinder top and the large cylinder side. Figure 3 shows the results of plotting the averaged CMM error measurements in the X direction along with the predicted machine errors using the updated coefficients. The solid lines are the predicted errors and the dashed lines are the averaged CMM error measurements at the points along the small cylinder (points 9 to 17) and along the large cylinder (points 27 to 35). Figure 4 shows a similar plot, but for the errors in the Z direction. The errors along the top of the small cylinder are plotted for points 1 through 8 and those along the top of the large cylinder are for points 18 through 26. The maximum, mean and the standard deviation of the differences between the measured and predicted errors are given in Table 1. The new coefficients are given in Appendix B2.

| | Error | Difference (mm) | |
|---|---|---|---|
| | Maximum | Mean | Standard Deviation |
| X Direction | 0.0041 | 0.0014 | 0.0010 |
| Z Direction | 0.0025 | 0.0008 | 0.0007 |

**Table 1: This shows the Maximum, Mean and Standard Deviations of the differences in the errors plotted in figures 3 and 4.**

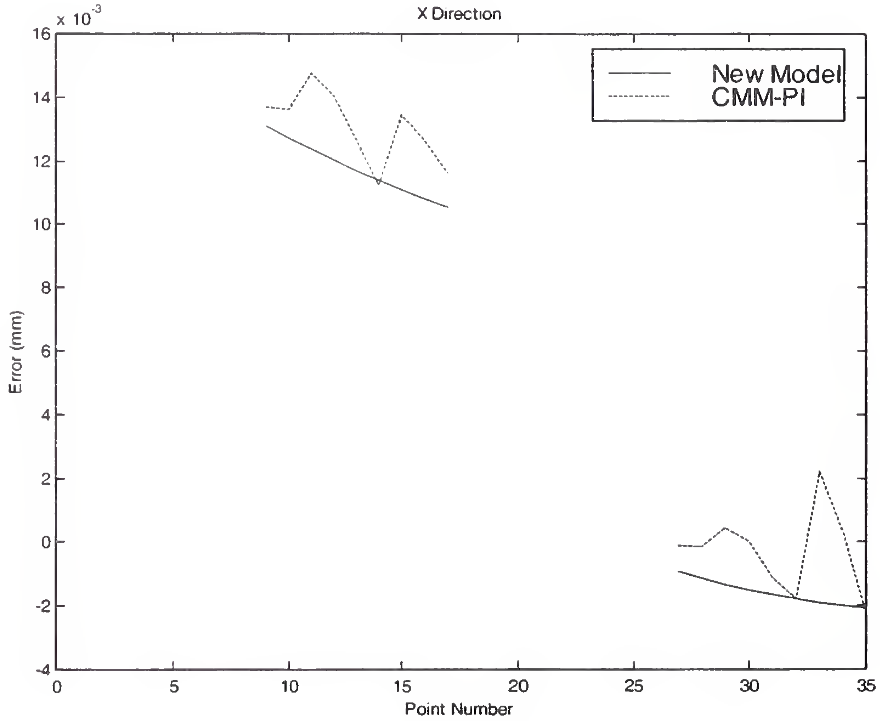

**Figure 3: Comparing the Measured Turning Center Errors (dashes) with the Predicted Errors (solid) based on Updated Turning Center Model in the X Direction.**

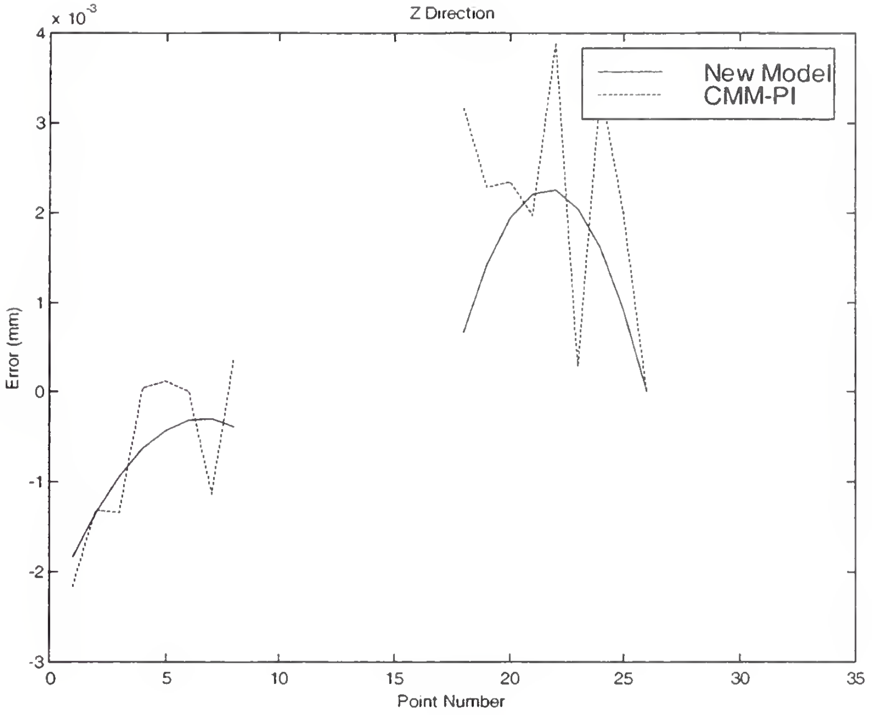

**Figure 4: Comparing the Measured Turning Center Errors (dashes) with the Predicted Errors (solid) based on Updated Turning Center Model in the Z Direction.**

## 5.0 Discussion

Machine tool error models are often developed and calibrated by using such techniques as laser interferometry and capacitance gauges to measure component errors such as scale and straightness errors. Since a machine tool's various parts such as bearings cannot in general be warmed up to functional conditions experienced during machining, another method must be developed to modify the calibrated machine error models to represent errors experienced under cutting conditions. This report proposes a method that can be used to modify the coefficients of a calibrated model so that it can be used during machining operations to compensate for machine tool errors.

The method employed here is a modified least squares technique that takes into account the fact that the normal equations generated during the least squares process may not be solvable in the usual sense. For that reason a generalized inverse is applied to the normal equations to generate the best solution in the sense of Appendix E. This is a well known tool for solving least squares problems and a thorough discussion can be found in [24].

The results shown in Table 1 and Figures 3 and 4 clearly show that the method produces an adjusted error model that accurately reproduces the measured data used to update the error model for the turning center used during the study. Appendices B1 and B2 show that the method significantly changes the coefficients to the basic form of the model. A reader should recognize that the model fits are adequate for the range of the data. They do not guarantee that the updated model could be used to compensate errors on turned parts dissimilar to the one used in the study. Further research must address the question of how to extend the model adjustment procedure described in this report to hold for families of similar parts or for other classes of parts. Further studies will also emphasize manufacturing parts under varying temperature conditions. Parts will also be designed to test different features placed throughout the workspace

Since the intent of the machine tool error model is to be used for real time error compensation no attempt is made to fit model forms beyond cubic spatial terms. This is because past experience has shown that cubic polynomials can be evaluated and their results passed to a machine tool to correct tool position within the real time constraints of the cutting process. Furthermore, higher order polynomials can introduce unwanted oscillations between fitting points.

Again the authors emphasize that this is not a report that claims that using singular value decomposition is the only tool for modifying or developing a machine tool error compensation model. However, given the initial error component model forms on which to base the turning center error compensation model, the authors felt that singular value decomposition was a sufficiently robust tool to use for the current study.

.

## 6.0 References

1. Chatterjee, S., An Assessment of Quasi-Static and Operational Errors in NC Machine Tools, Journal of Manufacturing Systems, vol. 16, 1, pp. 59-68, 1997.
2. Chen, X.B., Geddam, S., Yuan, Z.J., Accuracy Improvement of Three-Axis CNC Machining Centers by Quasi-Static Error Compensation, journal of Manufacturing Systems, Vol 16, 5, pp. 323-336, 1997.
3. Donmez, M.A., A General Methodology for Machine Tool Accuracy Enhancement: Theory, Application and Implementation, Ph.D. Thesis, Purdue University, 1985.
4. Donmez, M.A., Blomquist, D.S., Hocken, R.J., Liu, C.R., Barash, M.M., A general methodology for machine tool accuracy enhancement by error compensation, Precision Engineering, 8, pp.187-195, 1986.
5. Donmez, M.A., Yee, K.W., Damazo, B., Some Guidelines for Implementing Error Compensation on Machine Tools, NISTIR 5236, National Institute of Standards and Technology, Gaithersburg, MD, 1993.
6. Dorndorf, U., Kiridena, V.S.B., Ferreira, P.M., Optimal Budgeting of Quasistatic Machine Tool Errors, Journal of Eng. for Industry, Vol. 116, pp. 43-53, 1994.
7. Ferreira, P.M., Liu, C.R., A Method for Estimating and Compensating Quasistatic Errors of Machine Tools, Journal of Engineering for Industry, Vol. 115, pp. 149-159, 1993.
8. Kreng, V.B., Liu, C.R., Chu, C.N., A Kinematic Model for Machine Tool Accuracy Characterisation, Int. J. Adv. Manuf. Technol, 9, pp. 79-86, 1994.
9. Sethupathi, R.B., Reduced Polynomial Representation for Machine Error Data, MS Thesis, The University of North Carolina at Charlotte, 1996.
10. Soons, J.A., Theuws, F.C., Schellekens, P.H., Modeling the errors of multi-axis machines: a general methodology, Precision Engineering, Vol, 14, pp. 5-19, 1992
11. Yang, S., Yuan, J., Ni, J., Accuracy Enhancement of a Horizontal Machining Center by Real-Time Error Compensation, Journal of Manufacturing Systems, Vol. 15, 2, pp. 113-124, 1996.
12. Donmez, A., Characterization of CNC Turning Center, 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: 77-100; 1990
13. Rudder, F. F., Machine Tool Characterization, NISTIR 89-4045, Progress Report of the Quality in Automation Project for FY88, C. D. Lovett (Ed.)., National Institute of Standards and Technology, Gaithersburg, MD: 48-73; 1989.
14. Rudder, F. F., Characterization of the Vertical Machining Center, 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: 33-76; 1990
15. Rudder, F. F., Characterization of the Vertical Machining 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: 5-23; 1991.

16. 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: 41-50; 1991.
17. 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, Gaithersburg, MD; 1996.
18. 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, 20899.
19. Yee, K. W., Bandy, H. T., Boudreaux, J., Wilkin, N. D., Automated Compensation of Part Errors Determined by In-Process Gauging, NISTIR 4854, National Institute of Standards and Technology, Gaithersburg, MD; 1992.
20. Vorburger, T.V., Yee, K.W., Scace, B.R., Rudder, F.F., A Strategy for the Quality Control of Automated Machine Tools, NISTIR 4773, National Institute of Standards and Technology, Gaithersburg, MD, 1992.
21. Vorburger, T.V., Yee, K.W., Scace, B.R., Rudder, F.F.,Post-Process Control of Machine Tools, Manufacturing Review, Vol. 7, 3, pp. 252-266, 1994.
22. Mou, J., Donmez, M.A., Cetinkunt, S., An Adaptive Error Correction Method Using Feature-Based Analysis Techniques for Machine Performance Improvement, Part 1: Theory Derivation, Journal of Engineering for Industry, Vol. 117, pp. 584-590, 1995.
23. Mou, J., Donmez, M.A., Cetinkunt, S., An Adaptive Error Correction Method Using Feature-Based Analysis Techniques for Machine Performance Improvement, Part 2: Experimental Verification, Journal of Engineering for Industry, Vol. 117, pp. 591-600, 1995
24. Lawson, C.L., Hanson, R.J., Solving Least Squares Problems, Prentice-Hall, Inc. Englewood Cliffs, 1974.
25. Stewart, G.W., Introduction to Matrix Computations, Academic Press, New York, 1973.
26. Donmez, M. A., Characterization of CNC Turning Center, NISTIR 4322, Progress Report of the Quality in Automation Project for FY89, M. A. Donmez (Ed.). National Institute of Standards and Technology, Gaithersburg, MD: 77-100; 1990.
27. 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.

# APPENDIX A

## A.1 Developing the Kinematic Model for a Turning Center

In this appendix the homogeneous transformations for the coordinate systems modeling the turning center components along with the final error model will be given. For a detailed discussion of the theoretical development of the transformations see [3-5]. The turning center that is modeled in this section is shown in Section 2, Figure 1. Table A1 lists thermocouples along with their position descriptors.

The first step in the model development is the establishment of a reference coordinate system. On the turning center this will be taken as the or absolute coordinate system, the origin of which is located at the farthest travel to the left along the z-axis and the farthest travel vertically alone the x-axis. For the turning center, modeled in this study, the absolute coordinate system origin aligns itself with a point that is located at the upper right rear of the tool slot as one faces the turret, when the turret is brought to Home position. This point is also called the tool setting point.

Let $(Z_1, X_1)$ be the offset of the origin of the z-slide coordinate system from the reference coordinate system. Upper case letters are constants representing coordinate system offsets from each other, whereas lower case axis letters represent variable values. Thus the origin of the z-slide axis system is set $Z_1$ units along the z-axis from the origin of the reference coordinate system and $X_1$ units along the x-axis from the origin of the reference coordinate system. The motion of the z-slide coordinate system (also called the carriage coordinate system) relative to the machine coordinate system is described by the homogeneous transformation

$$ {}^{0}T_{z} = \begin{pmatrix} 1 & -\varepsilon_z(z,T) & \varepsilon_y(z,T) & X_1 + \delta_x(z,T) \\ \varepsilon_z(z,T) & 1 & -\varepsilon_x(z,T) & \delta_y(z,T) \\ -\varepsilon_y(z,T) & \varepsilon_x(z,T) & 1 & Z_1 + z + \delta_z(z,T) \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (A.1) $$

$\varepsilon_z(z,T)$ is the roll error of the carriage given z motion as a function of the temperature state, T, of the machine.

$\varepsilon_x(z,T)$ is the pitch error of the carriage given z motion and temperature state T.

$\varepsilon_y(z,T)$ is the yaw error of the carriage given z motion and temperature state T.

$\delta_x(z,T)$ is the x-displacement error given z motion and temperature state T.

$\delta_y(z,T)$ is the y-displacement error given z motion and temperature state T.

$\delta_z(z,T)$ is the z scale error given z motion and temperature state T.

$$\delta_x(z,T) = \delta_x'(z,T) + \alpha_p(T) * \Delta z \qquad \text{(A.2)}$$

where

$\delta_x'(z,T)$ is the x-straightness error of the carriage given z motion and temperature state T.

$\alpha_p(T)$ is the parallelism error between the z-motion and axis average line of the spindle given temperature state T.

$\Delta z$ is the incremental z motion.

$T$ temperature state of the machine and is taken to be the vector $T = (T_0, \cdots, T_{39})$, of thermocouple temperatures.

Let $(Z_2, X_2)$ be the offset of the origin of the x-slide axis system (also called the cross slide) from the origin of the z-slide axis system then the transformation representing the motion of the x-slide motion relative to the z-slide is given by

$$^{z}T_x = \begin{pmatrix} 1 & -\varepsilon_z(x,T) & \varepsilon_y(x,T) & X_2 + x + \delta_x(x,T) \\ \varepsilon_z(x,T) & 1 & -\varepsilon_x(x,T) & \delta_y(x,T) \\ -\varepsilon_y(x,T) & \varepsilon_x(x,T) & 1 & Z_2 + \delta_z(x,T) \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{(A.3)}$$

| Thermocouple Locations | |
|---|---|
| Number | Location |
| 0 | Bottom of X-Scale (Glass) |
| 1 | Top of X-Scale |
| 2 | Coolant Tank |
| 3 | Not Used |
| 4 | Not Used |
| 5 | Not Used |
| 6 | Not Used |
| 7 | Right of Z-Scale |
| 8 | Right Center of Z-Scale |
| 9 | Left Center of Z-Scale |
| 10 | Left of Z-Scale |
| 11 | Top of X-Way |
| 12 | Bottom of X-Way |
| 13 | Top of X-Head |
| 14 | Bottom of X-Head |
| 15 | Bottom Left of Z-Slide |
| 16 | Top Left of Z-Slide |
| 17 | Bottom Right of Z-Slide |
| 18 | Top Right of Z-Slide |
| 19 | Hydraulic Tank |
| 20 | Left End of Lower Z-Way |
| 21 | Left End of Upper Z-Way |
| 22 | Right End of Lower Z-Way |
| 23 | Right End of Upper Z-Way |
| 24 | Lower Front of Spindle Head |
| 25 | Lower Rear of Spindle Head |
| 26 | Upper Front of Spindle Head |
| 27 | Upper Rear of Spindle Head |
| 28 | Left of Top of Spindle Head |
| 29 | Middle of Top of Spindle Head |
| 30 | Right of Top of Spindle Head |
| 31 | Bottom Left of Bed |
| 32 | Top Left of Bed |
| 33 | Bottom Right of Bed |
| 34 | Top Right of Bed |
| 35 | Near X-Drive Motor Shaft Bearing |
| 36 | Left Z-Ballscrew Bearing |
| 37 | Right Z-Ballscrew |
| 38 | X-Ballscrew Housing |
| 39 | Z-Ballscrew Nut |

**Table A1: Thermocouple Locations on the Turning Center.**

where

$\varepsilon_x(x,T)$ is the roll error of the cross slide with x motion at temperature state T.

$\varepsilon_z(x,T)$ is the pitch error of the cross slide with x motion at temperature state T.

$\varepsilon_y(x,T)$ is the yaw error of the cross slide with x motion at temperature state T.

$\delta_y(x,T)$ is the y displacement of the cross slide with x motion at temperature state T.

$\delta_x(x,T)$ is the x scale error of the cross slide with x motion at temperature state T.

$\delta_z(x,T)$ is the z displacement error of the cross slide with x motion at temperature state T.

$$\delta_z(x,T) = \delta_z'(x,T) + \alpha_o(T) * \Delta x \qquad \text{(A.4)}$$

where

$\delta_z'(x,T)$ is the z straightness of the cross slide with x motion at temperature state T.

$\alpha_o(T)$ is the orthogonality error between the x motion and the axis average line of the spindle at temperature state T.

$\Delta x$ is the incremental x motion.

Let the turret and x-slide coordinates overlay so that

$$^{x}T_t = \begin{pmatrix} 1 & -\varepsilon_z(t,T) & \varepsilon_y(t,T) & \delta_x(t,T) \\ \varepsilon_z(t,T) & 1 & -\varepsilon_x(t,T) & \delta_y(t,T) \\ -\varepsilon_y(t,T) & \varepsilon_x(t,T) & 1 & \delta_z(t,T) \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{(A.5)}$$

where, due to misalignments of the turret on its rotational axis,

$\varepsilon_x(t,T), \varepsilon_y(t,T), \varepsilon_z(t,T)$ are the rotational errors in station-to-station motion of the tool turret at temperature state T.

$\delta_x(t,T), \delta_y(t,T), \delta_z(t,T)$ are the translation errors in station-to-station motion of the tool turret at temperature state T.

A turret on the lathe has the capability of holding, for example, various cutters, tools and on-machine measurement devices. Station-to-station motion here refers to rotation of the turret from one cutter or tool position on the turret to another.

Let $(Z_4, X_4)$ be the offset of the cutter coordinate system from the turret coordinate system. Since the cutting tool coordinate frame is assigned to the tool tip (a point), the rotation errors have no effect. The errors involved are due in part to the fact that an actual tool is in error from its nominal dimension and in part to thermal effects during the cutting operations. The cutting tool error matrix relative to the turret is given by

$$ {}^{t}T_{c} = \begin{pmatrix} 1 & 0 & 0 & X_4 + \delta_x(c,T) \\ 0 & 1 & 0 & \delta_y(c,T) \\ 0 & 0 & 1 & Z_4 + \delta_z(c,T) \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{(A.6)} $$

where

$\delta_x(c,T), \delta_y(c,T), \delta_z(c,T),$ are the incremental tool dimension changes in the x, y and z directions at temperature state T.

Let $(Z_5, X_5)$ be the offset of the spindle coordinate system from the machine coordinate system. In the generic turning center model, the spindle has unrestricted rotation about the z axis so that $\varepsilon_z(s,T) = 0$. The spindle errors are due to either axis of rotation errors of the spindle or errors resulting from misalignments of the chuck. The spindle error matrix relative to the machine coordinate system is given by

$$ {}^{0}T_{s} = \begin{pmatrix} 1 & 0 & \varepsilon_y(s,T) & X_5 + \delta_x(s,T) \\ 0 & 1 & -\varepsilon_x(s,T) & \delta_y(s,T) \\ -\varepsilon_y(s,T) & \varepsilon_x(s,T) & 1 & Z_5 + \delta_z(s,T) \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{(A.7)} $$

where

$\varepsilon_x(s,T)$ is the tilt of the spindle in the non-sensitive direction, i.e. yaw motion about the x axis, at temperature state T.

$\varepsilon_y(s,T)$ is the tilt of the spindle in the sensitive direction or a pitch motion about the y axis at temperature state T.

$\delta_x(s,T)$ is the translations of the spindle in the sensitive or x direction at temperature state T.

$\delta_y(s,T)$ is the translations of the spindle in the non-sensitive or y direction at temperature state T.

$\delta_z(s,T)$ is the z-axial motion of the spindle at temperature state T.

Let $(z(w), x(w))$ be the desired point on the work piece to which the tool is to be brought. Finally, for the workpiece assume no rotational errors. The workpiece error transformation is given by:

$$ {}^{s}T_{w} = \begin{pmatrix} 1 & 0 & 0 & x(w)+\delta_x(w,T) \\ 0 & 1 & 0 & \delta_y(w,T) \\ 0 & 0 & 1 & z(w)+\delta_z(w,T) \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{(A.8)} $$

where

$\delta_x(w,T), \delta_y(w,T), \delta_z(w,T)$ are the changes of the coordinates of the point on the workpiece to which the tool is to be applied in the x, y and z directions at temperature state T.

To determine $(z(w), x(w))$ at the work point one equates the ideal transformation (i.e. without errors) products of the two chains

$$ {}^{0}T_{z}\,{}^{z}T_{x}\,{}^{x}T_{t}\,{}^{t}T_{c} = {}^{0}T_{s}\,{}^{s}T_{w} \quad \text{(A.9)} $$

and get

$$ \begin{aligned} x(w) &= X_1 + X_2 + X_4 + x - X_5 \\ z(w) &= Z_1 + Z_2 + Z_4 + z - Z_5 \end{aligned} \quad \text{(A.10)} $$

The offsets are given in Table A.2 for the turret lathe. The coordinate systems are not placed arbitrarily but are placed on the machine tool components in such a way that they are related to the measurement characteristics of the metrology devices used to calibrate the error components.

| | X | Z |
|---|---|---|
| 1 | -148 | -43 |
| 2 | 30 | -26 |
| 3 | 0 | 0 |
| 4 | -35 | 30 |
| 5 | -386 | 0 |

**Table A.2: Coordinate System Offsets in mm.**

The resulting generic error matrix can be computed as

$$E = ({}^{s}T_{w}^{-1})({}^{0}T_{s}^{-1}){}^{0}T_{z}\,{}^{z}T_{x}\,{}^{x}T_{t}\,{}^{t}T_{c} \qquad \text{(A.11)}$$

A symbolic manipulation package, such as MACSYMA, can be used to generate the inverse matrices and compute the final product E (see Appendix A.2.). The final displacement errors of the tool from the workpiece are given by:

$$\begin{aligned}
E_x &= \varepsilon_y(s,T) * Z_5 + (\varepsilon_y(z,T) + \varepsilon_y(x,T) + \varepsilon_y(t,T) - \varepsilon_y(s,T)) * Z_4 \\
&+ (\varepsilon_y(z,T) - \varepsilon_y(s,T)) * Z_2 - \varepsilon_y(s,T) * (Z_1 + z) + \delta_x(z,T) + \delta_x(x,T) \\
&- \delta_x(w,T) + \delta_x(t,T) - \delta_x(s,T) + \delta_x(c,T)
\end{aligned}$$

$$\begin{aligned}
E_y &= -\varepsilon_x(s,T) * Z_5 + (-\varepsilon_x(z,T) - \varepsilon_x(x,T) - \varepsilon_x(t,T) + \varepsilon_x(s,T)) * Z_4 \\
&+ (-\varepsilon_x(z,T) + \varepsilon_x(s,T)) * Z_2 + \varepsilon_x(s,T) * (Z_1 + z) + (\varepsilon_z(z,T) + \varepsilon_z(x,T) + \varepsilon_z(t,T)) * X_4 \\
&+ \varepsilon_z(z,T) * (X_2 + x) + \delta_y(z,T) + \delta_y(x,T) - \delta_y(w,T) + \delta_y(t,T) - \delta_y(s,T) + \delta_y(c,T)
\end{aligned} \qquad \text{(A.12)}$$

$$\begin{aligned}
E_z &= -\varepsilon_y(s,T) * X_5 + (-\varepsilon_y(z,T) - \varepsilon_y(x,T) - \varepsilon_y(t,T) + \varepsilon_y(s,T)) * X_4 \\
&+ (-\varepsilon_y(z,T) + \varepsilon_y(s,T)) * (X_2 + x) + \varepsilon_y(s,T) * X_1 + \delta_z(z,T) + \delta_z(x,T) \\
&- \delta_z(w,T) + \delta_z(t,T) - \delta_z(s,T) + \delta_z(c,T)
\end{aligned}$$

These equations give the generic error formulas for a turning center with its absolute reference coordinate system and the tool setting point assumed aligned at “cold start” in the “home” position.

There are only two equations in the final model. Errors in the y direction are not included since they are in the non-sensitive direction. The nonsensitive direction in turning centers is the direction perpendicular to the plane in which the two machine slides move. The only error components included were those found to be significant and not in nonsensitive directions. Only seven errors are included in the final model for the turning center to yield the error equations

$$\begin{aligned}
E_x &= (\varepsilon_y(z,T) + \varepsilon_y(x,T)) * Z_4 + \varepsilon_y(z,T) * Z_2 + (\delta_x{}'(z,T) + \alpha_p(T) * \Delta z) + \delta_x(x,T) \\
\\
E_z &= -(\varepsilon_y(z,T) + \varepsilon_y(x,T)) * X_4 - \varepsilon_y(z,T) * (X_2 + x) \;\; + \delta_z(z,T) \\
&+ (\delta_z{}'(x,T) + \alpha_o(T) * \Delta x) - \delta_z(s,T)
\end{aligned} \qquad \text{(A.13)}$$

The MACSYMA script used to generate these equations is given in Appendix A.2.

## A.2 MACSYMA Script for Kinematic Model Generation

This appendix contains the listing of a command script for the symbol manipulator language MACSYMA. The script can be loaded into MACSYMA by using the BATCH command. Note that the small variable arithmetic is taken care of by observing that the result of the product of two terms with small order variables is just the linearization of the final result with respect to all of the small variables. This can be accomplished by taking the Taylor series of the product up to the first order terms only. The final result of this script is given by equation (A.13).

```
/* Construct the list of error components */
list1:[ezz,eyz,dxz,exz,dyz,dzz,ezx,eyx,dxx,exx,dyx,dzx]$
list2:[ezt,eyt,dxt,ext,dyt,dzt,dxc,dyc,dzc]$
list3:[eys,dxs,exs,dys,dzs,dxw,dyw,dzw,ap,ao]$
list:append(list1,list2,list3)$
/* Declare Final Error Array */
array(Err,3)$
/* Initialize all transformation arrays to 0 */
array(ioTz, 4,4)$
array(izTx, 4,4)$
array(ixTt, 4,4)$
array(itTc, 4,4)$
array(ioTs, 4,4)$
array(isTw, 4,4)$
array(eoTz, 4,4)$
array(ezTx, 4,4)$
array(exTt, 4,4)$
array(etTc, 4,4)$
array(eoTs, 4,4)$
array(esTw, 4,4)$
for i:1 thru 4 do
        for j:1 thru 4 do
                (ioTz[i,j] : 0.0,
                izTx[i,j] : 0.0,
                ixTt[i,j] : 0.0,
                itTc[i,j] : 0.0,
                ioTs[i,j] : 0.0,
                isTw[i,j] : 0.0,
                eoTz[i,j] : 0.0,
                ezTx[i,j] : 0.0,
                exTt[i,j] : 0.0,
                etTc[i,j] : 0.0,
                eoTs[i,j] : 0.0,
                esTw[i,j] : 0.0)$
/* Initialize the ideal transformation components */
ioTz[1,1] : 1$
```

```
ioTz[2,2] : 1$
ioTz[3,3] : 1$
ioTz[4,4] : 1$
ioTz[1,4] : X1$
ioTz[3,4] : Z1+z$
izTx[1,1] : 1$
izTx[2,2] : 1$
izTx[3,3] : 1$
izTx[4,4] : 1$
izTx[1,4] : X2+x$
izTx[3,4] : Z2$
ixTt[1,1] : 1$
ixTt[2,2] : 1$
ixTt[3,3] : 1$
ixTt[4,4] : 1$
itTc[1,1] : 1$
itTc[2,2] : 1$
itTc[3,3] : 1$
itTc[4,4] : 1$
itTc[1,4] : X4$
itTc[3,4] : Z4$
ioTs[1,1] : 1$
ioTs[2,2] : 1$
ioTs[3,3] : 1$
ioTs[4,4] : 1$
ioTs[1,4] : X5$
ioTs[3,4] : Z5$
isTw[1,1] : 1$
isTw[2,2] : 1$
isTw[3,3] : 1$
isTw[4,4] : 1$
isTw[1,4] : xw$
isTw[3,4] : zw$
/* Initialize Error Components */
eoTz[1,1] : 1$
eoTz[1,2] : -ezz$
eoTz[1,3] : eyz$
eoTz[1,4] : dxz + ap*dz$
eoTz[2,1] : ezz$
eoTz[2,2] : 1$
eoTz[2,3] : -exz$
eoTz[2,4] : dyz$
eoTz[3,1] : -eyz$
eoTz[3,2] : exz$
eoTz[3,3] : 1$
eoTz[3,4] : dzz$
```

```
eoTz[4,4] : 1$
ezTx[1,1] : 1$
ezTx[1,2] : -ezx$
ezTx[1,3] : eyx$
ezTx[1,4] : dxx$
ezTx[2,1] : ezx$
ezTx[2,2] : 1$
ezTx[2,3] : -exx$
ezTx[2,4] : dyx$
ezTx[3,1] : -eyx$
ezTx[3,2] : exx$
ezTx[3,3] : 1$
ezTx[3,4] : dzx + ao*dx$
ezTx[4,4] : 1$
exTt[1,1] : 1$
exTt[1,2] : -ezt$
exTt[1,3] : eyt$
exTt[1,4] : dxt$
exTt[2,1] : ezt$
exTt[2,2] : 1$
exTt[2,3] : -ext$
exTt[2,4] : dyt$
exTt[3,1] : -eyt$
exTt[3,2] : ext$
exTt[3,3] : 1$
exTt[3,4] : dzt$
exTt[4,4] : 1$
etTc[1,1] : 1$
etTc[1,4] : dxc$
etTc[2,2] : 1$
etTc[2,4] : dyc$
etTc[3,3] : 1$
etTc[3,4] : dzc$
etTc[4,4] : 1$
eoTs[1,1] : 1$
eoTs[1,3] : eys$
eoTs[1,4] : dxs$
eoTs[2,2] : 1$
eoTs[2,3] : -exs$
eoTs[2,4] : dys$
eoTs[3,1] : -eys$
eoTs[3,2] : exs$
eoTs[3,3] : 1$
eoTs[3,4] : dzs$
eoTs[4,4] : 1$
esTw[1,1] : 1$
```

```
esTw[1,4] : dxw$
esTw[2,2] : 1$
esTw[2,4] : dyw$
esTw[3,3] : 1$
esTw[3,4] : dzw$
esTw[4,4] : 1$
/* Generate Matrices */
mioTz : genmatrix(ioTz,4)$
mizTx : genmatrix(izTx,4)$
mixTt : genmatrix(ixTt,4)$
mitTc : genmatrix(itTc,4)$
mioTs : genmatrix(ioTs,4)$
misTw : genmatrix(isTw,4)$
meoTz : genmatrix(eoTz,4)$
mezTx : genmatrix(ezTx,4)$
mexTt : genmatrix(exTt,4)$
metTc : genmatrix(etTc,4)$
meoTs : genmatrix(eoTs,4)$
mesTw : genmatrix(esTw,4)$
oTz : mioTz.meoTz$
zTx : mizTx.mezTx$
xTt : mixTt.mexTt$
tTc : mitTc.metTc$
oTs : mioTs.meoTs$
sTw : misTw.mesTw$
/* Compute Full Error Matrix */
E : (sTw^^-1).(oTs^^-1).oTz.zTx.xTt.tTc$
/* Zero variables not measured */
ezz:0$
exz:0$
dyz:0$
ezx:0$
exx:0$
dyx:0$
ezt:0$
eyt:0$
dxt:0$
ext:0$
dyt:0$
dzt:0$
dxc:0$
dyc:0$
dzc:0$
eys:0$
dxs:0$
exs:0$
```

```
dys:0$
dxw:0$
dyw:0$
dzw:0$
xw:X1+X2+X4+x-X5$
zw:Z1+Z2+Z4+z-Z5$
/* Evaluate E again */
for i:1 thru 4 do
        for j:1 thru 4 do
                E[i,j] : ev(E[i,j])$
/* Get the linearized displacement errors */
for i:1 thru 3 do
        Err[i] : taylor(E[i,4],list,0,1);
```

# APPENDIX B

This appendix lists the calibration coefficients for the original geometric-thermal model as well as the adjusted coefficients. Since the angular error equations are calibrated in arc seconds the coefficients are scaled to radians. Furthermore, straightness errors are calibrated in micrometers, and are converted to millimeters. All of the other component error equations are calibrated in millimeters. Temperatures are in degrees Celsius. This discussion of units is only given in order to interpret the coefficients given below. The methods described in this report are not dependent on these specific units.

## B.1 Error Model Calibration Coefficients

This appendix lists those initial coefficients $A_{ij}$ that apply to most of the machining conditions encountered on the turning center portrayed in Figure 1. The coefficients given here represent the coefficient values of the component error functions given in Section 3 for spatial motion in the negative x and positive z direction. These coefficients were developed during an earlier characterization of the turning center (see [26-27]). Spindle speed is assumed to be 3000 revolutions per minute.

### X Yaw

| $A_{11}$ | $A_{12}$ | $A_{13}$ |
|---|---|---|
| -1.339000E-04 | 5.289925E-02 | -3.490320E-01 |

### Z Yaw

| $A_{21}$ | $A_{22}$ | $A_{23}$ |
|---|---|---|
| 1.811200E-05 | 8.326300E-04 | -6.450000E-01 |

### X Straightness with Z Motion

| $A_{31}$ | $A_{32}$ | $A_{33}$ | $A_{34}$ |
|---|---|---|---|
| -4.192000E-05 | 3.091716E-02 | -4.283536E-01 | 1.007243E+01 |

### X Displacement

| $A_{41}$ | $A_{42}$ | $A_{43}$ | $A_{44}$ | $A_{45}$ |
|---|---|---|---|---|
| 3.362300E-07 | 2.466100E-04 | -2.828000E-06 | 1.700000E-04 | -1.808700E-03 |

### Z Displacement

| $A_{51}$ | $A_{52}$ | $A_{53}$ | $A_{54}$ |
|---|---|---|---|
| -1.521630E-08 | 3.270300E-05 | 1.231370E-03 | -3.586880E-02 |

### Z Straightness of X Motion

| $A_{61}$ | $A_{62}$ | $A_{63}$ | $A_{64}$ |
|---|---|---|---|
| 5.097300E-05 | -1.594320E-02 | 6.321090E-03 | 2.711352E-01 |

### Spindle Drift

| $A_{71}$ | $A_{72}$ | $A_{73}$ | $A_{74}$ |
|---|---|---|---|
| -1.020000E-04 | 1.206751E-02 | -3.894400E-03 | -1.292980E-01 |

### X Axis Squareness with respect to the Spindle Axis

| $A_{81}$ | $A_{82}$ |
|---|---|
| -1.354370E+00 | 1.031200E+01 |

### Parallelism of Z Axis with respect to the Spindle Axis

| $A_{91}$ | $A_{92}$ | $A_{93}$ | $A_{94}$ |
|---|---|---|---|
| -7.416000E-01 | 3.410200E+01 | 4.710000E-02 | -3.934200E+02 |

### Machine Yaw

| $A_{10,1}$ | $A_{10,2}$ |
|---|---|
| -5.113500E+00 | 1.146000E+02 |

## B.2 Error Model Adjusted Coefficients

This appendix lists the updated coefficients $A_{ij}$ .

### X Yaw

| $A_{11}$ | $A_{12}$ | $A_{13}$ |
|---|---|---|
| -1.5941621E-02 | -6.5335550 | -8.1679140E+02 |

### Z Yaw

| $A_{21}$ | $A_{22}$ | $A_{23}$ |
|---|---|---|
| 1.8534552E-03 | 5.8945116E-01 | 1.8865618E+02 |

### X Straightness with Z Motion

| $A_{31}$ | $A_{32}$ | $A_{33}$ | $A_{34}$ |
|---|---|---|---|
| 3.0374870E-04 | -1.4928799E-01 | -3.2036746 | -5.6535790E+01 |

## X Displacement

| $A_{41}$ | $A_{42}$ | $A_{43}$ | $A_{44}$ | $A_{45}$ |
|---|---|---|---|---|
| 1.3291142E-06 | 4.0311660E-05 | -1.0762552E-05 | -2.3918381E-03 | -6.8416490E-023 |

## Z Displacement

| $A_{51}$ | $A_{52}$ | $A_{53}$ | $A_{54}$ |
|---|---|---|---|
| -5.3957796E-06 | -1.0142461E-03 | -2.1621554E-03 | -1.1814533E-01 |

## Z Straightness of X Motion

| $A_{61}$ | $A_{62}$ | $A_{63}$ | $A_{64}$ |
|---|---|---|---|
| -3.3490506E-03 | -9.2555965E-01 | -2.7085232 | -8.1174190E+01 |

## Spindle Drift

| $A_{71}$ | $A_{72}$ | $A_{73}$ | $A_{74}$ |
|---|---|---|---|
| -5.1565641E-06 | 1.487646E-02 | -3.5329871E-04 | -4.7854670E-02 |

## X Axis Squareness with respect to the Spindle Axis

| $A_{81}$ | $A_{82}$ |
|---|---|
| -1.0356114E-01 | 4.0332133E+01 |

## Parallelism of Z Axis with respect to the Spindle Axis

| $A_{91}$ | $A_{92}$ | $A_{93}$ | $A_{94}$ |
|---|---|---|---|
| -7.6842435E-01 | 3.3485040E+01 | -4.9867230E-01 | -4.0761008E+02 |

## Machine Yaw

| $A_{10,1}$ | $A_{10,2}$ |
|---|---|
| 1.4453392 | 2.7857098E+02 |

## APPENDIX C: Adjusting for Tool Offsets

All points on a part are usually specified in terms of part coordinate system points

$$(z^p, x^p). \qquad \text{(C.1)}$$

Therefore the part origin must be determined relative to the machine coordinate system. This offset will be designated as:

$$(Z_{off}, X_{off}). \qquad \text{(C.2)}$$

and is decomposed into two separate offsets. The first offset, $(Z_{tool\_set}, X_{tool\_set})$, is the offset from the machine coordinate system to a point on the turret, called the tool locating point, to which tool inserts in the turret are referenced. The second offset, $(Z_{tool}, X_{tool})$, is the $Z$ and $X$ values of the tool point length from the tool locating point. Thus,

$$\begin{aligned} Z_{off} &= Z_{tool\_set} + Z_{tool}, \\ X_{off} &= X_{tool\_set} + X_{tool}. \end{aligned} \qquad \text{(C.3)}$$

The conversion of a part coordinate to an absolute coordinate is then given by

$$\begin{aligned} z &= z^p + Z_{off}, \\ x &= x^p + X_{off}. \end{aligned} \qquad \text{(C.4)}$$

## APPENDIX D: Determining the Machine Tool Errors from Part Measurements

Although equation (18) associates the x and z residual machine errors at a general point with the perturbation terms $d_{ij}$, this equation cannot directly be used to estimate the perturbations. The x and z residual errors at a set of sample points on several parts must be determined. These residual errors are usually obtained by inspecting parts on a CMM.

Assume that the machining is done in, say, the positive z and negative x direction. This is the standard cutting direction used during this study. The significance of this assumption lies in the selection of the characterization equation coefficients given in Appendix B.1. Let there be *n* inspection points on the surface of a part. One needs to take *n* > 17, since there are 35 unknown perturbation terms and two coordinates per inspected point. Designate these nominal inspection points as

$$(z_1^{nom}, x_1^{nom}), \cdots, (z_n^{nom}, x_n^{nom}) \quad \text{(D1)}$$

Suppose on a CMM these points are inspected and the CMM returns the following coordinates, where the superscript *p* means inspected,

$$(z_1^p, x_1^p), \cdots, (z_n^p, x_n^p) \quad \text{(D2)}$$

Suppose that the process errors can be identified at *n* inspection points on the surface of the part using process-intermittent probing methods as described in [17]. If these errors are defined as dimensional discrepancies normal to the surface, then the errors must be decomposed into x and z components at each of the inspection points. Thus we suppose that we have measured process error component pairs

$$(E_{z,PI}^1, E_{x,PI}^1), \cdots, (E_{z,PI}^n, E_{x,PI}^n) \ \text{(D3)}$$

Next, we convert the inspection points, assumed given in part coordinates to absolute coordinates by computing using equation (C4).

For the absolute coordinates (C4) the predicted z and x errors are computed from equation (2), using the coefficients from Appendix B1. Once this has been done we have *n* predicted machine error pairs

$$(E_{z,pred}^1, E_{x,pred}^1), \cdots, (E_{z,pred}^n, E_{x,pred}^n). \quad \text{(D4)}$$

The residual errors are then computed at the inspection points as

$$\begin{aligned} \Delta E_x^i &= E_{x,meas}^i - E_{x,PI}^i - E_{x,pred}^i \\ \Delta E_z^i &= E_{z,meas}^i - E_{z,PI}^i - E_{z,pred}^i \end{aligned} . \qquad \text{(D5)}$$

The subscript *meas* indicates that these errors are measured errors on the CMM and the subscript *PI* indicates that these errors are the process-intermittent errors measured on the part while it is on the machine tool.

## APPENDIX E: Numerical Aspects of Pseudo Inversion

Equation (26) is in general an overdetermined system in that there are more rows than columns or more known parameter coefficients than unknown parameters. Ordinary inversion of $J$ is not possible. A reasonable approach to obtaining a "solution" to (26) is to employ a least squares procedure. If the columns of $J$ are linearly independent then this can usually be done by fairly standard least squares software. However, if some of the columns are dependent then other procedures must be applied. It is for that reason we introduce the idea of a pseudo inverse that will be described below.

In order to "solve" (26) for the unknown vector $U$ we need to examine some issues involved with the least-squares solution of equation (26). A numerical difficulty arises if there are nonzero vector solutions $u$ of

$$Ju = 0 \qquad \text{(E1)}$$

There are an infinite number of solutions of (E1) since any multiple of a solution is also a solution. Thus if $U_1$ is a solution of (26) and $u$ is a solution of (41) then, for any constant $c$, the vector $U_1 + cu$ is a solution of (26) since $J(U_1 + cu) = JU_1 + cJu = JU_1$. The problem then is to select from the infinite number of solutions one particular "good" solution.

To analyze the difficulty we need to introduce some terminology from matrix theory. The matrix $J$ is a transformation of vectors of length 35 to vectors of length 2n. The vectors of length 35 will be called the *domain* vectors and the vectors of length 2n will be called the *range* vectors. The range of $J$ is a linear vector space in that the sum of two elements in the range is also in the range as well as a scalar multiple of an element in the range since $Ju_1 + Ju_2 = J(u_1 + u_2)$ and $cJu = J(cu)$. The matrix $J$ can also be written in the following form

$$J = \left(J_1, J_2, \cdots, J_{35}\right) \qquad \text{(E2)}$$

where $J_i$, for $i = 1, \cdots, 35$, are the columns of the matrix. Each of these columns can also be thought of as a 2n-length vector. If $u$ is a vector of length 35 with scalar elements $u_i$, for $i = 1, \cdots, 35$, then we can write, using (E2),

$$Ju = (J_1, J_2, \cdots, J_{35})\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{35} \end{pmatrix} = \sum_{i=1}^{35} u_i J_i \qquad \text{(E3)}$$

This shows that the range of the matrix $J$ is the set of linear combinations of the columns of $J$, thought of as vectors. A subset of the column vectors $J_i$, for $i = 1,\cdots,35$, is said to be *linearly independent* if

$$\sum_{j=1}^{k} u_j J_{i_j} = 0 \quad \text{(E4)}$$

implies that $u_j = 0$, for $j = 1,\cdots,k$, where $k \leq 35$. This says that the k vectors (or columns) form a set of coordinate vectors for the range space and that k is the *dimension* of the range space. The number $k$ is called the *column rank* of the matrix $J$. If $k$ is 35 then the matrix is said to be of *full rank*. The null space of the matrix $J$ is the set of all vectors $u$ that satisfy $Ju = 0$. This set is also a vector space. Its dimension, also called its *nullity*, is the largest set of linearly independent vectors that satisfy $Ju = 0$. A fundamental result in matrix theory is that the sum of the rank and the nullity of a matrix is the dimension of the domain space.

In order to determine a least-squares solution of equation (26) some measure of the difference between the right and left-hand sides of (26) must be defined. The usual measure or metric used extends the idea of Euclidean distance to, in this case, vectors of length $2n$, where n is the number of probed points on the workpiece. We will denote the length or norm of a $2n$-vector by

$$\|v\| = \sqrt{\sum_{i=1}^{2n} v_i} \,. \quad \text{(E5)}$$

The least-squares problem associate with equation (26) is to find the $2n$-vector that minimizes the norm of the difference between the right and left-hand sides of (26). This is symbolically written as

$$\min_u \|Ju - v\| \quad \text{(E6)}$$

where $v$ is a right-hand side vector for (26). From linear regression analysis the solution of (E6) depends on solving the system

$$J^T Ju = J^T v \quad \text{(E7)}$$

where the superscript $T$ represents the transpose of the matrix. There are two cases to consider: (1) $rank(J) = 35$ and (2) $rank(J) = k < 35$. If the rank of $J$ is 35, then the rank of $J^T J$ is 35 and it is a square matrix. Therefore $J^T J$ has an inverse and (E7) is uniquely solvable. This is the traditional least-squares solution of (E6). However, if $rank(J) = k < 35$, then the rank of $J^T J$ is $k$ (< 35), which means that the nullity of $J^T J$ is nonzero and (E7) has an infinite number of solutions or an infinite number of vectors that satisfy (E6). The object then is to select an appropriate solution for (38) and therefore ultimately for (26).

A significant method for decomposing matrices, such as $J$, is called the *Singular Value Decomposition* and is strongly related to the eigenvalue-eigenvector decomposition of $J^T J$. The singular value decomposition of the $2n$ by 35 matrix $J$ of rank $k$ is given by

$$J = ODW^T \qquad \text{(E8)}$$

where O, a $2n$ by $2n$ matrix and W, a 35 by 35 matrix, are *orthogonal*, which means they satisfy

$$\begin{aligned} OO^T &= O^T O = I \\ WW^T &= W^T W = I \end{aligned} \qquad \text{(E9)}$$

and $D$ is a $2n$ by 35 diagonal matrix with nonnegative diagonal elements arranged to be nonincreasing from upper left to lower right. Some of them in the lower right can be 0. The relation to the matrix $J^T J$ is given by

$$J^T J = WD^T DW^T \qquad \text{(E10)}$$

which is not hard to show by multiplying the transpose of (E8) times (E8) and using (E9). This shows that the singular values of $J$ are the positive square roots of the eigenvalues of $J^T J$.

The singular value decomposition of $J$ can be used to solve the least-squares minimization problem (E6) by using the following result (See [24]).

Suppose that $J$ is decomposed, using the singular value decomposition to the form (E10) where the matrix $D$ can be written in the block form

$$D = \begin{pmatrix} D_{11} & 0 \\ 0 & 0 \end{pmatrix} \qquad \text{(E11)}$$

where the submatrix $D_{11}$ is a diagonal $k \times k$ matrix with the nonnegative diagonal elements, containing the singular values, arranged to be nonincreasing from upper left to lower right. Introduce two new variables $g$ and $y$ by

$$\begin{aligned} O^T v = g &= \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} \\ W^T u = y &= \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \end{aligned} \qquad \text{(E12)}$$

where $g_1, y_1$ are vectors of length $k$, the rank of $D_{11}$, $g_2$ is a vector of length $2n$-$k$ and $y_2$ is a vector of length 35-$k$. Note that at this point, since $u$ is unknown, $y$ is unknown. Now let $\tilde{y}_1$ be the unique solution of the matrix equation

$$D_{11} y_1 = g_1 \qquad \text{(E13)}$$

Then the first conclusion is that all solutions of the minimization problem (E6) are of the form

$$\hat{u} = W \begin{pmatrix} \tilde{y}_1 \\ y_2 \end{pmatrix} \qquad \text{(E14)}$$

where $y_2$ is an arbitrary vector of length 35-$k$. Next, any vector of the form (E14) generates the same residual vector $r$ given by

$$r = v - J\hat{u} = O \begin{pmatrix} 0 \\ g_2 \end{pmatrix} \qquad \text{(E15)}$$

This says also that the vector of the difference between any two vectors of the form (E14) lies in the null space of $J$. Therefore any two vectors of the form (E14) differ by a vector in the null space of $J$. The norm of $r$ satisfies $\|r\| = \|v - Ju\| = \|g_2\|$ and, finally, the unique solution of the minimization problem (38) with the minimum length is given by

$$\tilde{u} = W \begin{pmatrix} \tilde{y}_1 \\ 0 \end{pmatrix} \qquad \text{(E16)}$$

The fact that (E16) is the unique solution of minimum length of (E6) permits the definition of a concept of *generalized inverse* of the matrix $J$ even though it is not a square matrix. In particular the following result also holds. If the matrix $J$ is written in the form of its singular value decomposition (E8) where the diagonal matrix $D$ is written in the block form (E11) then the unique minimum length solution of the minimization problem (E6) can also be written as

$$\tilde{u} = W \begin{pmatrix} D_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix} O^T v \qquad \text{(E17)}$$

by combining (E12), (E13) and (E16). Now define

$$J^+ = W \begin{pmatrix} D_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix} O^T \qquad \text{(E18)}$$

The matrix $J^+$ is called the *generalized inverse* of *J*.

Although the matrix $D_{11}$ has all of its diagonal elements nonzero so that it has the form

$$D_{11} = \begin{pmatrix} d_1 & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & d_k \end{pmatrix} \quad \text{(E19)}$$

care must be taken when the inverse is computed numerically. Theoretically the inverse of $D_{11}$ has elements of the form $\frac{1}{d_i}, i = 1, \cdots, k,$ but the values of $d_i$ may be sufficiently small so that their reciprocals are numerically meaningless. Thus, when the generalized inverse of a matrix is computed the diagonal matrix of singular values $D_{11}$ is usually redefined to be the matrix that contains only the singular values whose absolute values are larger than some tolerance. All others are set to zero, so that from a numerical point of view the rank of $D_{11}$ may be less than *k*. A tolerance that is used often in determining those singular values that are to be eliminated is given by

$$tol = 2n\|J\|\varepsilon \quad \text{(E20)}$$

where the norm of a matrix is computed in a similar manner to the norm of a vector in that the square root of the sum of the squares of all of the elements is taken as the norm and $\varepsilon$ is called the *machine epsilon* and is that positive finite machine precision number that satisfies $1 + \varepsilon = 1$. For an ordinary PC using double precision the number is approximately 2.2E-16.