A Perturbation Approach to Robot Calibration

H. O. K. Kirchner*
B. Gurumoorthy
F. B. Prinz

CMU-RI-TR-85-9

(3)

Department of Mechanical Engineering
The Robotics Institute
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

April 1985

H. O. K. Kirchner: Institut fur Festkorperphysik, Universitat Wien, A-1090 Boltzmanngasse 5, Vienna, Austria.

Table of Contents

ABSTRACT

The forward and reverse solutions of a manipulator with six joints are expanded up to second order in the 24 joint parameters.

1 INTRODUCTION

It is well known that the absolute accuracy of most available robotic manipulators is relatively poor. If the robot is commanded to move to a certain position in world space, it will actually move to a slightly different position. Typically, the differences between the commanded and the actual position may be of the order of .5% of the dimensions of the robot. In absolute terms, this inaccuracy could be between .1 and 1.2 inches (Kumar and Waldron 1981, Mooring 1983). The magnitude of this error is significantly above acceptable tolerance limits for most manufacturing applications.

If the robot is programmed by conventional teaching-by-doing methods, problems of absolute accuracy are of no importance because they are corrected through the visual feedback of the human operator.

Current trends indicate, however, that in the near future robots will be programmed off-line with the help of model based systems. In such an integrated environment, it is of vital importance that the computer model of the robot maps, as closely as possible, its physical pendant. Hence the need arises for developing mathematical tools to obtain the signature of individual errors and project it into the model as closely as possible.

It is important to note that the overall errors of the robot are of a geometric and a non-geometric kind. The latter ones may be due to joint compliance, gear transmission errors and backlash (Whitney, et al. 1984; Mooring 1983; Whitney and Lozinski 1984). The former ones may result from imprecise manufacturing of the robot links and joints. Deviations in the relative link positions, i.e., encoder errors or offsets, will be part of the geometric error.

Depending on the actual application of the manipulator, errors of one kind or another may become predominant. In high speed assembly operations, in particular, dynamic errors might be more important than errors in the manipulator geometry. In applications like welding or slow motion/high precision tasks, the situation is likely to be reversed.

This paper attempts to establish a mathematical reference frame for geometric errors only. No attempt is made to quantify the relative contribution of geometric and non-geometric errors, since this ratio is application dependent and will also depend on the design of the manipulator itself. Since in high speed operations high positional accuracy is desirable, we address the question of how to cope with geometric deviations from the ideal manipulator configuration.

Due to manufacturing errors, each of the robot links will be slightly different from the ideal one. For an N degree of freedom manipulator there will be N joints and N links. Each of the links can be characterized by two dimensions: the normal distance a between the joints it connects, and the twist angle *a* between the joint axes it connects. If the design calls for certain values of a and *a,* the real robot will have link values a + Aa and *a + Aa.* Each joint sits between two links that intersect the joint axis. The distance d between the subsection of the links, as well as the angle *6* between the links measured on a plane normal to the joint axis, characterizes the joint. If the design calls for certain values of d and *6,* the real robot will have joint values d + Ad and *9 + Ad* (Paul 1981).

Each of the- joint-link pairs achieves a rotation by an angle *a,* followed by a translation of d and a in different directions, followed once more by a rotation by an angle *a.* The real robot will effect these by amounts *a + Aa,* d + Ad, a + Aa and *6 + Ad.* The computation of subsequent rotation, translation and rotation is straightforward. Of the various representations of the resulting screw displacement that are possible (Rooney 1978), the most popular among roboticists is the use of homogeneous 4 x 4 matrices. The concatenation of N such joint-link pairs gives the final transformation that describes the translation and rotation of the end effector relative to the base of the robot.

This calculation can be done for both the ideal and the real robot. For a six degrees of freedom robot 6 x 4 = 24 joint-link parameters exist, but only six of these parameters need to be controllable, the other eighteen remaining fixed and uncontrolled. In the following, we will denote the 24 joint-link parameters by the variables $p_1,...,p_{24}$, with the first six $p_1,...,p_6$ chosen to represent the six joint-link variables that are being controlled. For the Stanford manipulator, they would be (p1 ,.«,pJ ~" *(6}**^2'd3'^4#^5'^6^ whereas for the elbow manipulator described by Paul (19811 they would be (p^./pJ = *(&,J'JQJAJ^JJ.*

It is not difficult to compute, for both the ideal and the real robot, the so-called forward solution that describes the translation and rotation of the end effector relative to the robot base. The same formal expressions result for both robots, but with the values pt + Ap^p24 + Ap24 replacing trie values P^-rP^r if Ap^.Ap^ . are the errors of the 24 joint-link parameters. Although the result for the real robot is considerably more complicated, it still is of manageable proportions and can be given in an analytic expression.

Since the errors will be small, it is tempting to expand the real forward solution around the ideal one in powers of the errors $\Delta p_1,...,\Delta p_{24}$. Calculations by Chi-haur Wu (1984) confirm the practical result that for the real robot, translation and rotation of the end effector are not quite what they are meant to be (i.e., the translation and rotation achieved by the ideal robot). They also show that the errors of a particular link propagate in a complicated way through the other links that connect it to the end effector.

The real problem arises if one wishes to find which values of the joint variables $p_1,...,p_6$ assure a certain position of the end effector. Since the forward transformation is highly nonlinear, its inversion is not a simple task. In practice, it can only be done for relatively simple configurations where most of the 18 joint-link parameters $p_7,...,p_{24}$ are identically zero. Since, by definition, this is not the case for the real robot, the exact inverse solution of the real robot is almost impossible to find! So far, even approximations to the real reverse solution have been lacking-- it was simply assumed that the real inverse solution is identical to the ideal one.

If a robot is given the task of reaching a particular position, the joints are adjusted to values $p_1,...,p_6$ computed from the ideal reverse solution. If the robot were error free, it would indeed reach the prescribed position. If it is not error free, the prescribed position is not reached. This is because the *ideal* reverse solution is being used to find the values $p_1,...,p_6$, which are then propagated through the *real* forward solution. Since ideal reverse and real forward solution do not close, errors in the positioning of the end effector are the result.

The task of geometric robot calibration can be defined as finding the errors $\Delta p_1,...,\Delta p_{24}$ and constructing the appropriate real reverse and forward solutions. The errors could, for example, be measured by triangulation of the robot.

An intermediate step to geometric robot self-calibration avoids this triangulation of the joints and links of the robot, and replaces it by triangulation of the positioning of the end effector. From these errors the errors $\Delta p_1,...,\Delta p_{24}$ are deduced and then used in the real forward solution. The task is to find corrections $\delta p_1,...,\delta p_6$ of the joint-link variables so that the *real* robot (with $\Delta p_1,...,\Delta p_{24} \neq 0$) reaches the same positioning as the ideal robot (with vanishing errors $\Delta p_1 = ... = \Delta p_{24} = 0$). Once the mapping how the joint corrections depend on the joint coordinates

$$\begin{matrix} \delta p_1(p_1,\dots,p_6) \\ \\ \delta p_6(p_1,\dots,p_6) \end{matrix} \tag{1.1}$$

is established, it can be used for further error-free manipulations. This approach is taken in this paper.

Complete geometric self-calibration would be a strategy by which the robot searches for certain calibration points by skillful variation of $\delta p_1,\dots,\delta p_6$. The resulting mapping like (1.1) could then be stored without actual evaluation and use of the errors $\Delta p_1,\dots,\Delta p_{24}$.

2 FORWARD SOLUTION

To find the forward solution for the real robot, one needs to start with the A matrices of the individual links.

The A matrix for the ideal robot is of the form (Paul 1981)

$$A_i^i = \begin{pmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & r_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.1}$$

and the A matrix of the real robot is

$$A_i^r = \begin{pmatrix} \cos(\theta_i+\Delta\theta_i) & -\sin(\theta_i+\Delta\theta_i)\cos(\alpha_i+\Delta\alpha_i) & \sin(\theta+\Delta\theta_i)\sin(\alpha_i+\Delta\alpha_i) & (a_i+\Delta a_i)\cos(\theta_i+\Delta\theta_i) \\ \sin(\theta_i+\Delta\theta_i) & \cos(\theta_i+\Delta\theta_i)\cos(\alpha_i+\Delta\alpha_i) & -\cos(\theta+\Delta\theta_i)\sin(\alpha_i+\Delta\alpha_i) & (a_i+\Delta a_i)\sin(\theta_i+\Delta\theta_i) \\ 0 & \sin(\alpha_i+\Delta\alpha_i) & \cos(\alpha_i+\Delta\alpha_i) & r_i+\Delta r_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.2}$$

These matrices are multiplied to obtain the ideal and real forward transformations

$$T^i = A_1^i A_2^i \dots A_6^i \tag{2.3}$$

$$T^r = A_1^r A_2^r \dots A_6^r \tag{2.4}$$

Although the complete forward transformation T^r can be written down in closed form, such an expression would be of little practical use. Although the inverse transformation $(T^i)^{-1}$ of the ideal forward transformation can be written down in analytic form, it is unlikely that such an expression could be found for the real inverse $(T^r)^{-1}$.

Since the differences $\Delta\theta_i,\dots,\Delta\theta_6$, $\Delta\alpha_1,\dots,\Delta\alpha_6$, $\Delta r_1,\dots,\Delta r_6$ and $\Delta a_1,\dots,\Delta a_6$ are assumed to be small, it is more advantageous to expand T^r around T^i in powers of these errors. This gives, up to second order in errors ΔA_i of the A matrices.

$$
\begin{aligned}
T^r - T^i &= T^r(\Delta A_i \neq 0) - T^r(\Delta A_i = 0) \\
&= \sum_{i=1}^{6} A_1,\dots,\Delta A_i,\dots,A_6 \\
&+ \sum_{i=1}^{6}\sum_{i\neq j=1}^{6} A_1,\dots,\Delta A_i,\dots,\Delta A_j,\dots,A_6 \qquad (2.5) \\
&+ \sum_{i=1}^{6} A_1,\dots,\Delta^2 A_i,\dots,A_6
\end{aligned}
$$

Since

$$
\frac{\partial A_i}{\partial\theta_j} = \frac{\partial A_i}{\partial\alpha_j} = \frac{\partial A_i}{\partial r_j} = \frac{\partial A_i}{\partial a_j} = 0 \quad \text{for } i \neq j \qquad (2.6)
$$

the linear errors are given by

$$
\Delta A_i = \frac{\partial A_i}{\partial\theta_i}\Bigg|_{\substack{\Delta\alpha_i=\Delta r_i\\=\Delta a_i=0}} \Delta\theta_i + \frac{\partial A_i}{\partial\alpha_i}\Bigg|_{\substack{\Delta\theta_i=\Delta r_i\\=\Delta a_i=0}} \Delta\alpha_i + \frac{\partial A_i}{\partial r_i}\Bigg|_{\substack{\Delta\theta_i=\Delta\alpha_i\\=\Delta a_i=0}} \Delta r_i + \frac{\partial A_i}{\partial a_i}\Bigg|_{\substack{\Delta\theta_i=\Delta\alpha_i\\=\Delta r_i=0}} \Delta a_i \qquad (2.7)
$$

$$
\Delta^2 A_i = \frac{\partial^2 A_i}{\partial\theta_i^2}\Bigg|_{\substack{\Delta\alpha_i=\Delta r_i\\=\Delta a_i=0}} (\Delta\theta_i)^2 + \frac{\partial^2 A_i}{\partial\alpha_i^2}\Bigg|_{\substack{\Delta\theta_i=\Delta r_i\\=\Delta a_i=0}} (\Delta\alpha_i)^2 + \frac{\partial^2 A_i}{\partial r_i^2}\Bigg|_{\substack{\Delta\theta_i=\Delta\alpha_i\\=\Delta a_i=0}} (\Delta r_i) + \frac{\partial^2 A_i}{\partial a_i^2}\Bigg|_{\substack{\Delta\theta_i=\Delta\alpha_i\\=\Delta r_i=0}} (\Delta a_i)^2 \qquad (2.8)
$$

$$+ \frac{\partial^2 A_i}{\partial\theta_i \partial\alpha_i}\Bigg|_{\Delta r_i=\Delta\alpha_i=0} \Delta\theta_i \Delta\alpha_i + \dots \qquad i = 1,\dots,6$$

with no sums on the index i. All the error matrices $\Delta A_1,\dots,\Delta A_6$ have by now been represented as homogeneous matrices multiplied by the 24 scalars Δd_i, $\Delta\theta_i$, Δr_i, $\Delta\alpha_i$ where $i = 1,\dots,6$.

Fortunately, the expansion (2.8) looks more complicated than it really is. Megahed and Renaud (1982) and Wu (1984) noticed that the first derivatives of the transformation matrices A_i are linear functions of the matrices themselves and can be written as

$$\frac{\partial A_i}{\partial\theta_i} = A_i Q_i^{\theta} \tag{2.9}$$

$$\frac{\partial A_i}{\partial d_i} = A_i Q_i^{d} \tag{2.10}$$

$$\frac{\partial A_i}{\partial r_i} = A_i Q_i^{r} \tag{2.11}$$

$$\frac{\partial A_i}{\partial \alpha_i} = A_i Q_i^{\alpha} \tag{2.12}$$

where

$$Q_i^{\theta} = \begin{pmatrix} 0 & -\cos\alpha_i & \sin\alpha_i & 0 \\ \cos\alpha_i & 0 & 0 & a_i\cos\alpha_i \\ -\sin\alpha_i & 0 & 0 & -a_i\sin\alpha_i \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{2.13}$$

$$Q^{\alpha} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{2.14}$$

$$Q_i^r = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sin\alpha_i \\ 0 & 0 & 0 & \cos\alpha_i \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad (2.15)$$

$$Q^a = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad (2.16)$$

Thus the differentiations are merely matrix multiplications

$$\Delta A_i = A_i[Q_i^\theta \Delta\theta_i + Q^a \Delta a_i + Q_i^r \Delta r_i + Q^a \Delta a_i] \qquad (2.17)$$

and

$$\Delta^2 A_i = A_i[Q_i^\theta Q_i^\theta (\Delta\theta_i)^2 + Q^a Q^a (\Delta a_i)^2 + Q_i^r Q_i^r (\Delta r_i)^2 + Q^a Q^a (\Delta a_i)^2$$
$$+ Q_i^\theta Q^a (\Delta\theta_i \Delta a_i) + \ldots] \qquad (2.18)$$

It should also be noted that Q^a and Q^a are the same for all the links. Finally, the difference $T^r - T^1$ can be written as a multiple of T^1:

$$T^r - T^1$$
$$= \sum_{i=1}^{6} (A \ldots A Q \ldots A) \Delta\theta_i + \ldots \qquad (2.19)$$
$$= \sum_{i=1}^{6} [(A_1 \ldots A_i Q_i A_i^{-1} \ldots A_1^{-1}) A_1 \ldots A_6 \Delta\theta_i + \ldots$$
$$= \sum_{i=1}^{6} [\Omega_i^\theta \Delta\theta_i + Q^a \Delta a_i + Q^r \Delta r_i + Q^a \Delta a_i] T^i$$
$$+ \sum_{i=1}^{6} \sum_{j=1}^{6} (\Omega_i^\theta \Omega_j^\theta)(\Delta\theta_i \Delta\theta_j)$$
$$+ (\Omega^a \Omega^a)(\Delta a_i \Delta a_j)$$
$$+ (\Omega_i^r \Omega_j^r)(\Delta r_i \Delta r_j)$$
$$+ (\Omega^a \Omega^a)(\Delta a_i \Delta a_j)$$
$$+ (\Omega_i^\theta \Omega_j^a)(\Delta\theta_i \Delta a_j) + \ldots] T^j$$

where

$$\Omega_i^\theta = A_1 \ldots A_i Q_i^\theta A_i^{-1} \ldots A_1^{-1}$$

$$\Omega_i^{\alpha} = A_1 \ldots A_i \Omega_i^{\alpha} A_i^{-1} \ldots A_1^{-1}$$
$$\Omega_i^{r} = A_1 \ldots A_i \Omega_i^{r} A_i^{-1} \ldots A_1^{-1} \qquad (2.20)$$
$$\Omega_i^{a} = A_1 \ldots A_i \Omega_i^{a} A_i^{-1} \ldots A_1^{-1}$$

The problem of computing derivatives like those above also arises in robot dynamics (Brady, et al. 1982). Recursive methods as proposed by Hollerbach (1980) and Book (1983) reduce the problem to linear complexity, but nonlinear complexity schemes might actually be faster (Luh, Walker and Paul 1980; Walker and Orin 1982). As an example we give the matrices $\partial A_1/\partial \alpha_1$ and $\partial^2 A_1/\partial \alpha_1 \partial \alpha_1$ for the first link of the Stanford manipulator. Its variable is θ_1, and the ideal link parameters are α = -90°, a = d = 0. One obtains, up to second order in $\Delta\alpha_1$

$$A_1^r = \begin{pmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 0 \\ \sin\theta_1 & 0 & \cos\theta_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (2.21)$$

$$+ \begin{pmatrix} 0 & -\sin\theta_1 & 0 & 0 \\ 0 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \Delta\alpha_1 + \ldots$$

$$+ \begin{pmatrix} 0 & 0 & \sin\theta_1 & 0 \\ 0 & 0 & -\cos\theta_1 & 0 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \Delta\alpha_1 \Delta\alpha_1 + \ldots$$

where we left out 3 other linear terms and 15 other quadratic terms.

At this point there is no difference between link variables and link parameters, and one could write

$$T(p_i + \Delta p_i) - T(p_i) = T^r(p_i) - T^i(p_i) = \qquad (2.22)$$

$$= \sum_{i=1}^{24} \frac{\partial T}{\partial p_i} \Delta p_i + \sum_{i=1}^{24} \sum_{j=1}^{24} \frac{\partial^2 T}{\partial p_i \partial p_j} \Delta p_i \Delta p_j$$

$$= \sum_{i=1}^{24} K_i \, \Delta p_i + \sum_{i=1}^{24} \sum_{j=1}^{24} \Delta p_i \, L_{ij} \, \Delta p_j$$

Here T, K_i and L_{ij} are 4 x 4 homogeneous matrices, with the indices i and j not being the matrix index but running from 1,...,24. There are 24 matrices K_i. Since

$$L_{ij} = L_{ji} \tag{2.23}$$

there are 24 • 25/2 = 300 matrices L_{ij}. Computation of the 24 matrices K_i by hand is still feasible (and was done by Wu (1984) for the general case), but computation of the 300 second order matrices is better left to some algebraic manipulation routine. The K_i and L_{ij} matrices are not as complicated as might appear at first sight because not all the errors $\Delta p_1,\ldots,\Delta p_{24}$ are of the same importance.

Considering one particular A_i matrix, like (2.2), it can be seen that the 3 x 3 rotation matrix in the upper left-hand corner involves only the rotational errors $\Delta\theta_i$ and $\Delta\alpha_i$, while the translational part, which are the first three elements of the last column, involve both the translational errors Δa_i and Δr_i as well as the rotational error $\Delta\theta_i$, but not $\Delta\alpha_i$. The way these errors in the individual $(A_i + \Delta A_i)$ matrices propagate into the final $(T + \Delta T)$ matrix becomes more transparent if the multiplicative structure (2.3), (2.4) is changed to a polynomial one that separates rotation and translation. In such a representation the total rotation $\bar{R}$ and total translation $\bar{t}$ derive as follows from the individual R_i and t_i of the A_i. The total transformation consists of a rotation $\bar{R}$ followed by a translation $\bar{t}$ so that

$$\bar{R}\underset{\sim}{x} + \bar{t} = R_1\ [\ldots[R_5(R_6\underset{\sim}{x} + t_6) + t_5]\} + t_1 \tag{2.24}$$

Comparison of terms shows that

$$\bar{R} = R_1R_2R_3R_4R_5R_6 \tag{2.25}$$

$$\begin{aligned}\bar{t} = {} & R_1R_2R_3R_4R_5t_6 \\ & + R_1R_2R_3R_4t_5 \\ & + R_1R_2R_3t_4 \\ & + R_1R_2t_3 \\ & + R_1t_2 \\ & + t_1\end{aligned} \tag{2.26}$$

Not all the errors occur in all the matrices and vectors. Upon inspecting (2.2), one sees that only the rotational errors $\Delta\theta_i$ and $\Delta\alpha_i$, i = 1,...,6 affect the total rotation $\bar{R}$, while all errors Ad_i, Aa_i, Ar_i, Aa_i, i = 1,...,6 affect the total translation $\bar{t}$—with exception of Aa_6. This remark is correct to any power in the expansion in the errors. Up to first order the perturbed AR and At are not made up of 4 x 6 = 24 and 4 x 21 = 84 terms. In the perturbed AR there are only 12 terms linear in Ad_i and La_i, i = 1,...,6 and no terms in Ar_i and Aa_i. In the perturbed total translation AF there are 6 terms linear in Aa_i, i = 1,...,6 and 6 terms linear in Ar_i, i = 1,...,6. In the sum for AX there are also (6-i) terms in Aa_i, i = 1,...,6 and (7-i) terms in Ad_i, i = 1,...,6. Altogether there are only 60 and not 84 terms. Errors in the joint coordinate $A\#_1$ and parameters Aa_1 propagate most into the total translation t, errors Ad_i and Aa_i of higher indexed links less so, until the error Aa_6 of the last link does not affect the total translation at all.

3 ERROR EVALUATION

The forward transform matrices used hitherto are redundant. Physically the position of the end effector is characterized only by 6 scalar quantities. Three of them are the components of the translation vector, or T_{14}, T_{24} and T_{34} of the 4 x 4 T-matrix. The other three could be chosen as some three components of the rotational part of the T-matrix, say T.., T.. and T... These are the direction cosines between two of the reference axes and the three axes of the end effector. The way the rotation parameters are measured for calibration purposes is largely a matter of convenience. Other components of the rotation matrix or combinations of components could be used. Mathematically the most elegant, but in practice not least cumbersome way would be to use the axis of rotation which is the eigenvector of the rotation matrix and the angle of rotation C which can be deduced from the trace Tr $\underline{R}$ of the rotation matrix as

$$C = \text{arc cos}[(\text{Tr } \underline{R} - 1)/2] \qquad \{3.1\}$$

The rotation could then be described by the three components of the rotation vector which points in the direction of the eigenvector of $\underline{R}$ and is of length tg({72) where

$$tg^2\{£/2) * (2 - \text{Tr } \underline{R})/\text{Tr } \underline{R} \qquad (3.2)$$

in the **following** we will not specify how the resulting transformation is represented, but we will merely speak of Cartesian coordinates $x_1,...,x_6$, which are not necessarily of **vectoria!** character. They are simply six scalar functions that describe the position of the end effector.

The first task is to find the 24 errors $\Delta p_1,\ldots,\Delta p_{24}$ from 6 measured Cartesian errors $\Delta x_1,\ldots,\Delta x_6$. If one expands the difference between the position

$$x_i(p_1,\ldots,p_6;\ \Delta p_1,\ldots,\Delta p_6;\ p_7,\ldots,p_{24};\ \Delta p_7,\ldots,\Delta p_{24})$$

the *real* robot reaches when its joint coordinates have the values $p_1,\ldots,p_6$, and the position

$$x_i(p_1,\ldots,p_6;\ 0,\ldots,0;\ p_7,\ldots,p_{24};\ 0,\ldots,0)$$

the ideal robot would have reached with the same values $p_1,\ldots,p_6$ of the joint variables in terms of the joint errors

$$\Delta p_1,\ldots,\Delta p_{24}$$

one obtains

$$\begin{aligned} & x_i(p_1,\ldots,p_6;\ \Delta p_1,\ldots,\Delta p_6;\ p_7,\ldots,p_{24};\ \Delta p_7,\ldots,\Delta p_{24}) \\ & \quad - x_i(p_1,\ldots,p_6;\ 0,\ldots,0;\ p_7,\ldots,p_{24};\ 0,\ldots,0) \\ & \quad = \Delta x_i(p_j,\Delta p_j) \qquad\qquad i = 1,\ldots,6 \\ & \quad = \sum_{j=1}^{24} \left.\frac{\partial x_i}{\partial p_j}\right|_{\substack{\Delta p_s = 0 \\ j \neq s \in (1,\ldots,24)}} \Delta p_j + \sum_{j=1}^{24} \sum_{j=1}^{24} \left.\frac{\partial^2 x_i}{\partial p_j \partial p_k}\right|_{\substack{\Delta p_s = 0 \\ j \neq s \in (1,\ldots,24) \\ k \neq s \in (1,\ldots,24)}} \Delta p_j \Delta p_k + \ldots \end{aligned} \tag{3.3}$$

The first (real) values of $x_1,\ldots,x_6$ are measured, the second (ideal) values of $x_1,\ldots,x_6$ and the derivatives are computed from the ideal forward solution. The problem of calibration is to solve the system of six equations (3.3) for the 24 unknowns $\Delta p_1,\ldots,\Delta p_{24}$. Clearly, the problem is undetermined because the same Cartesian errors can be caused by various different sets of joint errors. A strictly consistent evaluation of errors would only be possible if their number were restricted to six. These six could then be found from the six equations (3.3). If less than eighteen errors Δp_i do not vanish identically, they can be consistently evaluated if there is no explicit variation of Δp_i with p_j, i.e., if the errors do not change across the working space of the robot. In that case, four sets of calibration points $x_i^{(0)}$, $x_i^{(1)}$, $x_i^{(2)}$, $x_i^{(3)}$ can be used to form a set of 24 Cartesian variables. This gives 24 equations of the form

$$\Delta x_i = \sum_{j=1}^{24} K_j^i \,\Delta p_j + \sum_{j=1}^{24}\sum_{k=1}^{24} \Delta p_j L_{jk}^i \,\Delta p_k + \ldots \qquad i = 1,\ldots,24 \tag{3.4}$$

where

$$K_j^{i+n} = \left.\frac{\partial x_i^{(n)}}{\partial p_j}\right|_{\substack{\Delta p_s = 0 \\ j \neq s \in (1,\ldots,24)}} \tag{3.5}$$

and

$$L_{jk}^{i+n} = L_{kj}^{i+n} = \left.\frac{\partial^2 x_i^{(n)}}{\partial p_j \partial p_k}\right|_{\substack{\Delta p_s = 0 \\ j \neq s \in (1,\ldots,24) \\ k \neq s \in (1,\ldots,24)}} \tag{3.6}$$

are operators that are linear and bilinear acting on Δp_j. They are known from the four real forward solutions used to find $x_i^{(0)}$, $x_i^{(1)}$, $x_i^{(2)}$, $x_i^{(3)}$. If the expansion (3.3) is taken to linear terms only, (3.4) is a linear system of 24 equations in 24 unknowns, which can be solved by inversion. If the expansion (3.3) is taken to quadratic terms, (3.4) is a system of 24 quadratic equations in 24 unknowns. Geometrically this is equivalent to finding the intersection of 24 quadratic surfaces in a space of dimension 24. The system could be solved by using a linear guess followed by a Newton-Raphson iteration. Since the expansion (3.3) could, however, be combined to any order of Δp_i, it seems more consistent to solve it by inversion of the series (3.4). One obtains

$$\Delta p_s = [K^{-1}]_{si} \left\{ \Delta x_i - \Delta x_s [K^{-1}]_{sj} L_{jk}^i \, [K^{-1}]_{kt} \Delta x_t + \ldots \right\} \tag{3.7}$$

with all the indices running from 1...24, summation over double indices being understood.

If the errors Δp_i vary across the working space of the robot, evaluation of (3.7) is not entirely consistent, because the data sets on the left-hand side of (3.4) have been obtained for different sets of joint variables, and thus for different errors Δp_i.

Each of the points $x_i^{(0)}$, $x_i^{(1)}$, $x_i^{(2)}$, $x_i^{(3)}$ gives an undetermined system of equations vith infinitely many solutions. The solution (3.7) satisfies each of these four sets of ix equations, but there is no guarantee that it is actually the physically correct olution for each of the four points. Continuity arguments indicate that the common olution should be close to the physically correct ones. If there is no explicit ariation across the working space, the common solution is indeed the correct one or each of the points. If there is an explicit dependence, the choice of four alibration points becomes problematic: the nearer the four calibration configurations re chosen to each other, the more ill-conditioned the problem becomes nathematically; the further away the points are chosen, the more the results Δp_s will epresent some average over the six-dimensional volume covered by the four points. he whole work space of the robot could, however, be divided into six-dimensional implexes with seven corners each, for each of which the Δp_s are computed and ttributed to the center of the seven points involved. The resulting verdetermination of the problem would require the use of pseudo-inverses. Some olynomial fit could then be used to represent the behavior of the errors Δp_s nroughout the working volume.

REVERSE TRANSFORMATION

Since at present there is no standard algorithm available to construct reverse olutions in general, they are usually found by geometric intuition or trial and error. uch an approach is, however, likely to fail if the forward solution becomes complex n the presence of errors. The resulting trigonometric equations become too omplicated to be solved. They can be transformed to algebraic equations via a alf-tangent substitution (Duffy 1980), but these are of high order and difficult to andle.

The perturbation approach developed so far avoids the need for the integral reverse ansformation, what is needed are relationships between perturbations in joint and aboratory space. Unlike the forward and the reverse transforms, which are highly onlinear, there is a linear relationship between differential changes in the Cartesian nd differential changes in the joint coordinates. Since the forward solution of the erturbed robot is known, it can be differentiated to obtain that linear transformation, ne Jacobian. Near the singular points of the robot, the joint motions become nearly dependent and the Jacobian becomes singular. For most positions, however, ne Jacobian can be inverted, although that is a time-consuming process. An dditional complication arises because the forward transformation describes how 6 variables $x_1,\ldots,x_6$ depend on 6 variables $p_1,\ldots,p_6$ and on 18 parameters $p_7,\ldots,p_{24}$. In the inverse transform the six variables $p_1,\ldots,p_6$ depend on the six Cartesian coordinates $x^1,\ldots,x^6$, but the eighteen quantities $p^7,\ldots,p^{24}$ parameterize the transformation. In the case of the real robot 24 additional error terms $\Delta p^1,\ldots,\Delta p^{24}$ appear as parameters.

In order to find the difference of the two forward Jacobians let us first consider a Taylor expansion of the forward transformation.

$$x_i(p_1,\ldots,p_6;\ \Delta p_1,\ldots,\Delta p_6;\ p_7,\ldots,p_{24};\ \Delta p_7,\ldots,\Delta p_{24})$$
$$-\ x_i(p_1,\ldots,p_6;\ 0,\ldots,0;\ p_7,\ldots,p_{24};\ 0,\ldots,0)$$

$$= \sum_{j=1}^{6} \left.\frac{\partial x_i}{\partial p_j}\right|_{\substack{\Delta p_s = 0 \\ \text{for} \\ j \neq s \in (1,\ldots,24)}} \Delta p_j + \sum_{j=7}^{24} \left.\frac{\partial x_i}{\partial p_j}\right|_{\substack{\Delta p_s = 0 \\ \text{for} \\ j \neq s \in (1,\ldots,24)}} \Delta p_j + \ldots \qquad i = 1,\ldots,6 \qquad (4.1)$$

$$+ \sum_{j=1}^{24} \sum_{k=1}^{24} \left.\frac{\partial^2 x_i}{\partial p_j \partial p_k}\right|_{\substack{\Delta p_s = 0 \\ \text{for} \\ j \neq s \in (1,\ldots,24) \\ k \neq s \in (1,\ldots,24)}} \Delta p_j \Delta p_k + \ldots$$

This expansion is with respect to all Δp_j, $j = 1,\ldots,24$ values. It is taken relative to the position of no errors at all, which is the position the *ideal* robot would have reached.

By definition $0 = \Delta p_1 = \ldots = \Delta p_{24}$ for the ideal robot and the ideal Jacobian is

$$J_{ideal} = \left.\frac{\partial x_i}{\partial p_j}\right|_{\substack{\Delta p_k = 0 \\ \text{for} \\ j \neq k \in (1,\ldots,24)}} \qquad \begin{matrix} i = 1,\ldots,6 \\ j = 1,\ldots,6 \end{matrix} \qquad (4.2)$$

The ideal Jacobian of (4.2) and the other coefficients of equation (4.1) cannot even in principle, be measured on the real robot which is to be calibrated. They can only be computed from the ideal forward solution.

Another expansion can be performed with respect to controllable changes $\delta p_1,\ldots,\delta p_6$, with the uncontrollable errors $\Delta p_1,\ldots,\Delta p_{24}$ being held constant at their actual value:

$$x_i(p_1,\ldots,p_6;\ \delta p_1 + \Delta p_1,\ldots,\delta p_6 + \Delta p_6;\ p_7,\ldots,p_{24};\ \Delta p_7,\ldots,\Delta p_{24})$$
$$- x_i(p_1,\ldots,p_6;\ \Delta p_1,\ldots,\Delta p_6;\ p_7,\ldots,p_{24};\ \Delta p_7,\ldots,\Delta p_{24})$$
$$= \sum_{j=1}^{6} \left.\frac{\partial x_i}{\partial p_j}\right|_{\substack{\delta p_s = 0 \text{ for } j \neq s \epsilon (1,\ldots,6) \\ \Delta p_s \neq 0 \text{ for } s \epsilon (1,\ldots,24)}} \delta p_j \qquad i = 1,\ldots,6 \tag{4.3}$$
$$+ \sum_{j=1}^{6}\sum_{k=1}^{6} \left.\frac{\partial^2 x_i}{\partial p_j \partial p_k}\right|_{\substack{\delta p_s = 0 \text{ for } j \neq s \epsilon (1,\ldots,6),\ k \neq s \epsilon (1,\ldots,6) \\ \Delta p_s \neq 0 \text{ for } j \epsilon (1,\ldots,24),\ k \epsilon (1,\ldots,24)}} \delta p_j \delta p_k + \ldots$$

This expansion is taken relative to the position with errors introduced because of non-vanishing $\Delta p_1,\ldots,\Delta p_{24}$, which is the position the *real* robot would reach without adjustments of $\delta p_1,\ldots,\delta p_6$. By definition of the real robot, the errors Δp_k for k = 1,...,24 do not vanish and the real Jacobian is

$$J_{real} = \left.\frac{\partial x_i}{\partial p_j}\right|_{\substack{\delta p_k = 0 \text{ for } j \neq k \epsilon (1,\ldots,6) \\ \Delta p_k \neq 0 \text{ for } k \epsilon (1,\ldots,24)}} \qquad i = 1,\ldots,6 \quad j = 1,\ldots,6 \tag{4.4}$$

where it is understood that $\Delta p_1,\ldots,\Delta p_{24}$ are being held constant at the error values of the actual robot. In principle, the real Jacobian (4.4) can be measured for the real robot. This could be done by comparing the changes in Cartesian positions with changes in $\delta p_1,\ldots,\delta p_6$. These can be controlled arbitrarily, unlike the errors $\Delta p_1,\ldots,\Delta p_{24}$ over which there is no control

The task of calibration compensation is now to compensate by adjustments of $\delta p_1,\ldots,\delta p_6$ which, according to (4.4) propagate into Cartesian changes through the *real* Jacobian, for the Cartesian changes introduced by the errors $\Delta p_1,\ldots,\Delta p_{24}$ which, according to equation (4.2) propagate through the *ideal* Jacobian. The equation to be satisfied is

$$\sum_{j=1}^{6} \frac{\partial x_i}{\partial p_j}\Bigg|_{\substack{\delta p_s = 0 \text{ for } j \neq s \in (1,\ldots,6) \\ \Delta p_s \neq 0 \text{ for } s \in (1,\ldots,24)}} \delta p_j$$

$$+ \sum_{j=1}^{6}\sum_{k=1}^{6} \frac{\partial^2 x_i}{\partial p_j \partial p_k}\Bigg|_{\substack{\delta p_s = 0 \text{ for } s \neq j \in (1,\ldots,6),\ s \neq k \in (1,\ldots,6) \\ \Delta p_s \neq 0 \text{ for } s \in (1,\ldots,24)}} \delta p_j \delta p_k + \ldots$$

$$+ \sum_{j=1}^{24} \frac{\partial x_i}{\partial p_j}\Bigg|_{\Delta p_s = 0 \text{ for } j \neq s \in (1,\ldots,24)} \Delta p_j + \sum_{j=1}^{24}\sum_{k=1}^{24} \frac{\partial^2 x_i}{\partial p_j \partial p_k}\Bigg|_{\substack{\Delta p_s = 0 \text{ for } j \neq s \in (1,\ldots,24) \\ k \neq s \in (1,\ldots,24)}} \Delta p_j \Delta p_k$$

$$= 0 \qquad i = 1,\ldots,6 \tag{4.5}$$

This is of the form

$$[J_r]_{ij}\ \delta p_j + \delta p_j H^i_{jk}\ \delta p_k + \ldots + f_i = 0 \tag{4.6}$$

with all indices running from 1,....,6, summation over double indices being understood. The linear coefficient is the real Jacobian of (4.4), the quadratic coefficient is

$$H^i_{jk} = \frac{\partial^2 x_i}{\partial p_j \partial p_k}\Bigg|_{\substack{\delta p_s = 0 \text{ for } s \neq j \in (1,\ldots,6),\ s \neq k \in (1,\ldots,6) \\ \Delta p_s \neq 0 \text{ for } s \in (1,\ldots,24)}} \tag{4.7}$$

The inhomogeneous term is

$$f_i = \sum_{j=1}^{24} \frac{\partial x_i}{\partial p_j}\Bigg|_{\substack{\Delta p = 0 \\ \text{for} \\ j \,\#\, Sf(i....24)}} \Delta p_j + \sum_{j=1}^{24}\sum_{k=1}^{24} \frac{\partial^2 x_i}{\partial p_j \partial p_k}\Bigg|_{\substack{\Delta p = 0 \\ \text{for} \\ j \,\#\, S^*(1,...,24) \\ k \,\#\, s«(1....24)}} \Delta p_j \Delta p_k + \dots \tag{4.8}$$

and can be taken to any power in the errors Ap_j. The equation (4.6) is of the same structure as equation (3.4) and can be solved by inversion with the result

5>s = " H~\i <i + V J r "\ H!k H'Vt + -> {4_9)

where now all the sums run from 1,-..,6, summation over double indices being understood.

If $Ap_7 = \dots = \Delta p_{24} = 0$ one obtains simply that

$$Sp_i * - Ap_1 \qquad i = 1....6 \tag{4.10}$$

because for that case $J_r = J_i$ and the two Jacobians close. In this case, the errors Ap^i of the joints are compensated by adjustments - Ap! of the joints themselves. If the errors in the joint parameters $Ap^7,...,Ap^{24}$ do not vanish, they cause errors in the Cartesian coordinates that are carried forward through the ideal forward solution and ideal forward Jacobian, because the robot was assumed error free. The Cartesian errors are compensated for by adjustments that are carried forward through the real transform and Jacobian. This is reflected in equation (4.9) insofar as the correction terms $[J^{r-1}]$ and H^{jk} refer to the real robot, while the source terms f! according to (4.8) refer to the ideal robot. The expansion 14.9) is essentially an expansion of the non-closing product $[J^{r}][J.^{!}] \# 1$.

5 FINAL REMARKS

The problem under investigation is differential in nature **because** small differences between the real and the ideal configuration are being considered. In a similar way, robot dynamics Is concerned with differential changes of both world and Joint coordinates with respect to time. The dynamic problem becomes tractable because, unlike position transforms, which are highly nonlinear, the rate of changes of the joint coordinates and the changes of the Cartesian coordinates are linearly related. The Taylor expansions of the present paper essentially linearize the problem in a similar way, and make its inversion possible.

In recent years the actual computational time required for the solution of the dynamic problem of ideal, error-free robots has been reduced considerably by the use of iterative schemes (Brady, et al. 1982; Megahed and Renaud 1982). The implementation of special purpose processors further decreases computational times by up to two orders of magnitude (Duelen, Kirchhoff and Held 1984). One could hope that similar economics could advance the technique developed in this paper to actual on-line implementation. On the kinematic level this would mean calibrated motion of the real robot. On the dynamic level the goal is the use of the real rather than the ideal Jacobian.

In literature, it has been recognized (or rather remembered) that screw theory or motor calculus are suitable instruments for handling this type of problem (Ball 1900; v. Mises 1924; Rooney 1978; Featherstone 1983, 1984; Sugimoto and Duffy 1982; Sugimoto 1984; Hunt 1984). The six components of the most general displacement possible are represented without redundancy in that six dimensional vectorial approach, and the redundancy of the 4 x 4 homogeneous matrices is avoided. For notational reasons we prefer the purely multiplicative structure of the conventional notation accepted by roboticists.

6 REFERENCES

1. Ball, R. S. 1900. The Theory of Screws. Cambridge: Cambridge University Press.

2. Book, W. J. 1984. Recursive Lagrangian Dynamics of Flexible Manipulator Arms. Int. J. Robotics Research 3(3):87-101.

3. Brady, M., Hollerbach, J. M., Johnson, T. L., Lozano-Perez, T., and Mason, M. T. 1982. Robot Motion: Planning and Control. Cambridge: MIT Press.

4. Duffy, J. 1980. Analysis of Mechanisms and Robot Manipulators. London: Arnold; New York: Wiley.

5. Duelen, G., Kirchhoff, U. and Held, J. 1984 (May 21-23). The Inverse Kinematic Problem in Realtime Application. Computer Based Factory Automation. 11th Conf. on Production Research and Technology, Carnegie-Mellon University. Dearborn: Society of Manufacturing Engineers, pp. 393-400.

6. Featherstone, R. 1983. The Calculation of Robot Dynamics using Articulated Body Inertias. Int. J. Robotics Research 2(1): 13-30.

7. Featherstone, R. 1984 (April 10-11). Spatial Notation: A Tool for Robot Dynamics. Institute of Measurement and Control Symposium on "Robotics-Dynamics, Control and Advanced Programming/' Cambridge.

8. Hollerbach, J. M. 1980. A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity. IEEE Transactions on Systems, Man and Cybernetics SMC-10(11):739-736.

9. Hunt, K. H. 1984. Mechanisms—Research Directions in Kinematics and Geometry. J. Mechanisms, Transmissions and Automation in Design 106(3):262-263.

10. Kumar, A., and Waldron, K. J. 1981. Numerical plotting of positioning accuracy of manipulators. Mechanism Mach. Theory 16(4):361-368.

11. Lenarcic, J. 1983. A new method for calculating the Jacobian for a robot manipulator. Robotica 2(4):205-209.

12. Lenarcic, J. 1984. Kinematic Equations of Robot Manipulator. Digital Systems for Industrial Automation 2{2):133-151.

13. Luh, J. Y. S., Walker, M. W., and Paul, R. P. C. 1980. On-line Computational Scheme for Mechanical Manipulators. J. Dynamic Systems, Measurement and Control 1Q2(i):69-76.

14. Megahed, S., and Renaud, M. 1982 (June 9-11). Minimization of the computation time necessary for the dynamic control of robot manipulators. 12th Int. Symp. on Industrial Robots, Paris, France. Bedford: IFS Publications, pp. 469-478.

15. Mooring, B. W. 1983 (August 7-11). The Effect of Joint Axis Misalignment on Robot Positioning Accuracy. **Proc.** 1983 ASME Int. Conf. on Computers in Engineering, Chicago. New York: ASME, pp. IS 151-155.

16. Oh, Se-Young, Grin, D., and Bach, M. 1984. An Inverse Kinematic Solution for Kinematically Redundant Robot Manipulators. Journal of Robotic Systems t(3):235-249.

17. Grin, D., and **Schrader,** W. W. 1983. Efficient Jacobian determination for robot manipulators. Proc. 6th World Congress on Theory of Machines and Mechanisms, pp. 994-997.

18. Paul, R. P. 1981, Robot Manipulators: Mathematics, Programming, and Control. Cambridge, MA: MIT Press.

19. **Rooney,** J- 1978. A comparison of representations of general spatial screw displacement. Environment and Planning B5(11:45-88.

20. **Sugimoto,** K„ and Duffy, J. 1982. Application of Linear Algebra to Screw Systems. Mechanism and Machine Theory 17(11:73-83.

21. Sugimoto, K. 1984. Determination of Joint Velocities of Robots by Using Screws. J. Mechanisms, Transmissions and Automation in Design 106(2):222-227.

22. v. Mises, R. 1924. Motorrechnung, ein neues Hilfsmittel der Mechanik. ZAMM 4(2):155-181,(3):193-213.

23. Walker, M. W., and Orin, D. E. 1982. Efficient Dynamic Computer Simulation of Robotic Mechanisms. J. Dynamic Systems, Measurement and Control 104(3):205-211.

24. Whitney, D. E., DeFazio, T. L., Gustavson, R. E., Rourke, J. M., Seltzer, D. S., Edsall, A. C., Lozinski, C. A., and Kenwood, G. J. 1984 (May 21-23). Short- and Long-Term Robot Feedback: Multi-Axis Sensing, Control and Updating. Computer Based Factory Automation. 11th Conf. on Production Research and technology, Carnegie-Mellon University. Dearborn: Society of Manufacturing Engineers, pp. 147-152.

25. Whitney, D. E., and Lozinski, C. A. 1984. Industrial Robot Calibration Methods and Results. Proc. 1984 ASME Conf. on Computers in Engineering, Las Vegas.

26. Wu, Chi-haur 1982. A Kinematic CAD Tool for the Design and Control of a Robot Manipulator. Int. J. Robotics Research 3(1):58-67.