Accurate Determination of Object Position from Imprecise Data

K. C. Gupta
Department of Mechanical Engineering
(M/C 251)
University of Illinois at Chicago
Chicago, Illinois 60607-7022
e-mail: kcgupta@uic.edu

P. Chutakanonta
Department of Mechanical Engineering
(M/C 251)
University of Illinois at Chicago
Chicago, Illinois 60607-7022

The problem of accurate determination of object position from imprecise and excess measurement data arises in kinematics, biomechanics, robotics, CAD/CAM and flight/vehicle simulator design. Several methods described in the literature are reviewed. Two new methods which take advantage of the modern matrix oriented software (e.g., MATLAB, IMSL, EISPACK) are presented and compared with a "basic" method. It is found that both of the proposed decomposition methods (I: SVD/QR and II: SVD/QS) give better absolute results than a "basic" method available from the text books. On a relative basis, the second method (SVD/QS Decomposition) gives slightly better results than the first method (SVD/QR Decomposition). Examples are presented for the cases when the points chosen are nearly dependent and when the independent points have small random errors in their coordinates.

Introduction

Experimental determination of the position of an object is a problem of fundamental significance which is encountered in kinematics, robotics applications, CAD/CAM integration, FMS cell design, biomedical instruments and prosthetic device development, and flight and vehicle simulator designs. Typically, several points, called markers, are identified on the object and their coordinates measured experimentally by using radiography, light photogrammetry or motion picture photography. Errors of measurement are thus introduced into the marker coordinates. The goal is to determine, as accurately as possible, the position of the object either in the 4×4 homogeneous matrix representation or in terms of the screw parameters.

There is extensive literature on the theory of spatial transformations and displacements (Beggs, 1966; Suh and Radcliffe, 1978; Bottema and Roth, 1979; Angeles, 1982; Beggs, 1983; Cheng and Gupta, 1989; Gupta, ASME, 1997; and Gupta, Springer, 1997). The problem of determining object position from three or four exact point coordinates has also been addressed in the literature. Although these methods can be utilized when the point coordinates also contain experimental errors, the results are unreliable and inconsistent due to (i) the high degree of error sensitivity of the exact methods, (ii) the influence of point indexing scheme used, and (iii) the variations introduced by the specific choices of three or four points in the set. There is a very limited amount of work which deals with the more general problem when a large number of marker point coordinates are available.

This paper focuses upon the situation when: (i) patterned or random errors of measurement are present in the coordinates of the marker points, and (ii) excess (or redundant) measurements are made to compensate for the inevitable experimental inaccuracies. The prior works related to the extraction of object position from exact or inexact point coordinates are reviewed first. Then, two new methods, called SVD/QR Decomposition Method and SVD/QS Decomposition Method, are presented in detail. These methods exploit the capabilities of modern matrix oriented software such as MATLAB, IMSL, EISPACK. Examples have been presented to demonstrate robustness and reliability of these new methods.

Contributed by the Mechanisms Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Feb., 1998; revised Sept., 1998. Associate Technical Editor: K. Kazerovnian.

Methods Using Exact Point Coordinate Data

Under ideal conditions, it may be assumed that the experimental errors of measurement are negligible. Then, it is possible to apply classical methods to find the object position. A triad of noncollinear points is necessary to define the position of a body in three dimensions. Therefore, several methods are based upon the use of three exact point coordinates. On the other hand, four noncoplanar points can effectively utilize the special structure of the 4×4 homogeneous matrix formulation and that is why there are also methods which utilize four exact point coordinates.

Beggs (1966) proposed a method utilizing three exact points with body coordinates $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$, and global coordinates $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{P}_3$, respectively. The relation among these is through the rotation matrix R and a translational vector $\mathbf{d}$:

$$\mathbf{P}_i = [R]\mathbf{p}_i + \mathbf{d}, \quad i = 1, 2, 3. \tag{1}$$

Rearranging these equations to eliminate the vector $\mathbf{d}$, we get

$$[\mathbf{P}_2 - \mathbf{P}_1 \; \mathbf{P}_3 - \mathbf{P}_2 \; \mathbf{P}_1 - \mathbf{P}_3] = [R][\mathbf{p}_2 - \mathbf{p}_1 \; \mathbf{p}_3 - \mathbf{p}_2 \; \mathbf{p}_1 - \mathbf{p}_3], \tag{2}$$

or,

$$[B]_{3\times3} = [R]_{3\times3}[C]_{3\times3}. \tag{3}$$

However, this system can not be solved for the rotation matrix R because $[C]^{-1}$ doses not exist (its rank <3). After expansion of this system, three dependent equations are dropped, and we are left with six linear equations in nine unknowns. After solving the linear system in terms of three "free" variables, the orthogonality conditions for the rotation matrix are used to fix the values of the three "free" variables. The vector $\mathbf{d}$ is found from any one of the following equations:

$$\mathbf{d} = \mathbf{P}_i - [R]\mathbf{p}_i, \; i = 1, 2, \text{ or } 3. \tag{4}$$

This is only a brief outline of the method; specific details, which are quite tedious, and the consideration of special cases are omitted from the discussion here. Another method by Beggs (1983) exploits a special vector form of the spatial displacement formula (Bottema and Roth, 1979),

$$\mathbf{P}_i - \mathbf{p}_i = [\tan(\theta/2)]\mathbf{u} \times (\mathbf{P}_i + \mathbf{p}_i - 2\mathbf{Q}) + s\mathbf{u}, \tag{5}$$

Where $\mathbf{u}$ is the unit vector in the direction of the spatial screw axis, **Q** the screw axis location, θ the rotation angle and s the translation amount with respect to the screw axis. After taking dot product with **u**, a set of three equations is formed as

$$s = (\mathbf{P}_i - \mathbf{p}_i)^t\mathbf{u},\ i = 1, 2, 3, \tag{6}$$

and this linear system can be solved for the three components of **u** in terms of the parameter s. Then the fact that **u** is a unit vector, i.e., $\mathbf{u}^t\mathbf{u} = 1$, is used to determine the values of both s and **u**. Angle θ is found by forming certain cross product (for sin θ) and dot product (for cos θ), and finally, the location vector **Q** is found—but these details are omitted here.

Bottema and Roth (1979) give the following elegant formulae to find the parameters of the screw displacement:

$$\mathbf{u} \tan(\theta/2) = \mathbf{D}/E, \tag{7}$$

where,

$$\mathbf{D} = [(\mathbf{P}_3 - \mathbf{P}_1) - (\mathbf{p}_3 - \mathbf{p}_1)] \times [(\mathbf{P}_2 - \mathbf{P}_1) - (\mathbf{p}_2 - \mathbf{p}_1)], \tag{8}$$

$$E = [(\mathbf{P}_3 - \mathbf{P}_1) - (\mathbf{p}_3 - \mathbf{p}_1)]\cdot[(\mathbf{P}_2 - \mathbf{P}_1) + (\mathbf{p}_2 - \mathbf{p}_1)], \tag{9}$$

and

$$\mathbf{Q} = (1/2)[\mathbf{P}_1 + \mathbf{p}_1 + \mathbf{u} \times (\mathbf{P}_1 - \mathbf{p}_1)\,\mathrm{ctg}(\theta/2) - (\mathbf{u}\cdot(\mathbf{P}_1 + \mathbf{p}_1))\mathbf{u}], \tag{10}$$

$$s = \mathbf{u}\cdot(\mathbf{P}_1 - \mathbf{p}_1). \tag{11}$$

From these explicit formulae for screw parameters θ, s, **u** and **Q**, the rotation matrix $R(\theta, \mathbf{u})$ and vector $\mathbf{d}(\theta, s, \mathbf{u}, \mathbf{Q})$ can be found easily. These direct formulae are simpler to use than either method of Beggs (1966 or 1983).

Angeles (1986) used three exact points to define their centroids in the body and global systems,

$$\mathbf{c} = (1/3) \sum_{i=1}^{3} \mathbf{p}_i \tag{12}$$

$$\mathbf{C} = (1/3) \sum_{i=1}^{3} \mathbf{P}_i \tag{13}$$

and the respective moments of inertias,

$$\mathbf{I}_b = \sum_{i=1}^{3} (\delta_i^t\delta_i[\mathbf{1}] - \delta_i\delta_i^t), \tag{14}$$

$$\mathbf{I}_g = \sum_{i=1}^{3} (\Delta_i^t\Delta_i[\mathbf{1}] - \Delta_i\Delta_i^t), \tag{15}$$

where $\delta_i = \mathbf{p}_i - \mathbf{c}$, and $\mathbf{\Delta}_i = \mathbf{P}_i - \mathbf{C}$. The two moments of inertias are related through the similarity transformation,

$$I_g = RI_bR^t. \tag{16}$$

Then, after finding the eigenvalues and eigenvectors of the moment of inertia matrices, using the Householder reflections, the rotation matrix R is constructed, and finally, the vector **d** is found.

Laub and Shiflett (1982) use certain theorems of linear algebra to find the rotation matrix R and vector **d**. Rewrite

$$\mathbf{P}_i = [R]\mathbf{p}_i + \mathbf{d},\quad i = 1, 2, 3 \tag{1}$$

as

$$[\mathbf{P}_1\ \mathbf{P}_2\ \mathbf{P}_3] = [R][\mathbf{p}_1\ \mathbf{p}_2\ \mathbf{p}_3] + \mathbf{d}[1\ 1\ 1], \tag{17}$$

or, symbolically as

$$[\mathbf{P}]_{3\times3} = [R]_{3\times3}[\mathbf{p}]_{3\times3} + [\mathbf{dh}^t]_{3\times3}, \tag{18}$$

where

$$[\mathbf{P}]_{3\times3} = [\mathbf{P}_1\ \mathbf{P}_2\ \mathbf{P}_3],$$
$$[\mathbf{p}]_{3\times3} = [\mathbf{p}_1\ \mathbf{p}_2\ \mathbf{p}_3], \tag{19}$$
$$\mathbf{h}^t = [1\ 1\ 1].$$

The expression for the rotation matrix R can then be written as

$$[R] = [\mathbf{P}][\mathbf{p}]^{-1} - [\mathbf{dh}^t][\mathbf{p}]^{-1} = [Q] - [\mathbf{dv}^t], \tag{20}$$

where $Q = [\mathbf{P}][\mathbf{p}]^{-1}$ and $\mathbf{v} = [\mathbf{p}]^{-t}\mathbf{h}$. The unknowns are the matrix R on the left hand side and vector **d** on the right hand side. Theorems of linear algebra are then used to find **d** as

$$\mathbf{d} = [Q - (\det Q)Q^{-t}]\mathbf{v}/(\mathbf{v}^t\mathbf{v}), \tag{21}$$

which is substituted into the expression for the rotation matrix to find it as $R = Q - \mathbf{dv}^t$.

Fenton and Shi (1989) compare these methods and conclude that the method of Bottema and Roth (1979) is the most efficient, while those of Laub and Shiflett (1982) and Angeles (1986) may have slight advantages in terms of accuracy when the point coordinate data contain small errors.

If the exact coordinates of four noncoplanar points are available, then these can be written in 4×4 homogeneous matrix representation as (Suh and Radcliffe, 1978)

$$\begin{bmatrix} \mathbf{P}_1 & P_2 & \mathbf{P}_3 & \mathbf{P}_4 \\ 1 & 1 & 1 & 1 \end{bmatrix} = [A] \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{22}$$

or, symbolically as

$$[B]_{4\times4} = [A]_{4\times4}[C]_{4\times4}, \tag{23}$$

where $[A]$ is the unknown 4×4 homogeneous transformation matrix. Because the four points are noncoplanar, C^{-1} exists, and the 4×4 transformation matrix can be found as $[A] = [B][C]^{-1}$.

Methods Using Imprecise Point Coordinate Data

If the point coordinate data have experimental errors, then the methods presented in the previous section can be used to find the position of the object from three or four inexact points. However, this approach has the following problems:

(I) the errors tend to get amplified,
(II) different results are found by
  (a) using different methods,
  (b) changing the indexing scheme of points in some methods, e.g., (Bottema and Roth, 1979),
  (c) altering the set of points being used.

Therefore, it is necessary to use methods which can utilize the excess data from the measurements. With built in redundancy in the data, it is reasonable to expect that some errors of measurement will cancel out, or their impact will be diminished.

Spoor and Veldpaus (1980) present a method which is based upon solving a constrained least-square problem using the coordinate data for n points. Define averages **p** and **P** as

$$\mathbf{p} = (1/n) \sum_{i=1}^{n} \mathbf{p}_i, \tag{24}$$

$$\mathbf{P} = (1/n) \sum_{i=1}^{n} \mathbf{P}_i, \tag{25}$$

the matrix M as

$$M = (1/n) \sum_{i=1}^{n} [\mathbf{p}_i\mathbf{P}_i^t] - [\mathbf{p}\ \mathbf{P}^t], \tag{26}$$

and constant f_o as

$$f_o = (1/n) \sum_{i=1}^{n} (\mathbf{p}_i^t\mathbf{p}_i + \mathbf{P}_i^t\mathbf{P}_i) - (\mathbf{p}^t\mathbf{p} + \mathbf{P}^t\mathbf{P}). \quad (27)$$

The error to be minimized is

$$f = (1/n) \sum_{i=1}^{n} (\mathbf{P}_i - R\mathbf{p}_i - \mathbf{d})^2, \quad (28)$$

which can be written as

$$f = f_o + (R\mathbf{p} + \mathbf{d} - \mathbf{p})^t(R\mathbf{p} + \mathbf{d} - \mathbf{p}) - 2\ \text{Trace}\ (M^tR). \quad (29)$$

The orthogonality constraint, $R^tR = I$, is incorporated through Lagrange multipliers. The solution details involve the solution of a related eigenvalue problem, and after the rotation matrix R and vector $\mathbf{d}$ are found, the screw parameters θ, s, $\mathbf{u}$, $\mathbf{Q}$ are found in the usual way.

Veldpaus, Wolting and Dortmans (1988) modified this tedious method by eliminating the time consuming steps of eigenvalue problem solution. Initially, the orthogonality constraint $R^tR = I$ is ignored, the symbol R is replaced by H, and the unconstrained least-square problem is solved for matrix H (now an approximation of rotation matrix R) and $\mathbf{d}$. Since the matrix H found in this way is not orthogonal, it is refined to produce an approximate orthogonal matrix R.

Shiflett and Laub (1995) have presented a method which also utilizes n point coordinates. Using a notation similar to Laub and Shiflett (1982), see equations (17–19), the data are organized as

$$[\mathbf{P}]_{3\times n} = [R]_{3\times 3}[\mathbf{p}]_{3\times n} + [\mathbf{dh}^t]_{3\times n}, \quad (30)$$

where

$$[\mathbf{P}]_{3\times n} = [\mathbf{P}_1\ \mathbf{P}_2 \ldots \mathbf{P}_n]_{3\times n},$$
$$[\mathbf{p}]_{3\times n} = [\mathbf{p}_1\ \mathbf{p}_2 \ldots \mathbf{p}_n]_{3\times n},$$
$$\mathbf{h}^t = [1\ 1 \ldots 1]_{1\times n}. \quad (31)$$

A constrained least-square problem is solved by minimizing the error ϵ,

$$\epsilon = \text{Trace}\ \{[[R][\mathbf{p}] - [\mathbf{P}]]^t[[R][\mathbf{p}] + \mathbf{dh}^t - [\mathbf{P}]]\}, \quad (32)$$

subject to the orthogonality constraint $R^tR = I$. After some derivations, the solution for the rotation matrix R is found as

$$R = B[B^tB]^{-1/2}, \quad (33)$$

where

$$[B]_{3\times 3} = [\mathbf{P}][I - (1/n)\mathbf{hh}^t][\mathbf{p}]^t, \quad (34)$$

and the vector $\mathbf{d}$ as

$$\mathbf{d} = (1/n)[[R][\mathbf{p}] + \mathbf{dh}^t - [\mathbf{P}]]_{3\times n}\mathbf{h}_{n\times 1}. \quad (35)$$

If the singular value decomposition for matrix B is used, i.e.,

$$B_{3\times 3} = U_{3\times 3}\sigma_{3\times 3}V^t_{3\times 3}, \quad (36)$$

where U and V are 3×3 orthogonal matrices, and σ is the 3×3 diagonal matrix containing singular values, then the rotation matrix R can be found simply as

$$R_{3\times 3} = U_{3\times 3}V^t_{3\times 3}. \quad (37)$$

Some issues related to real-time coordinate measurements and computations are discussed by Heeren and Veldpaus (1992). Data smoothing of the measured coordinate data before using the formulae is suggested by Challis (1995).

SVD/QR-Decomposition Method

The literature review has shown that while several excellent methods exist to determine object position from three or four exact point coordinate data, only a few methods, which are quite tedious to apply, exist for the case when redundant but imprecise point coordinate data are available. In view of the importance of the problem of determining object position accurately, this paper will present two methods for the case when excess data is available. Both method take advantage of matrix computation capabilities which are now readily available in MATLAB, IMSL or EISPACK, etc.

Let the body coordinates of the n marker points be $\mathbf{p}_i$, $i = 1, 2, \ldots, n$ and their global coordinates be $\mathbf{P}_i$, $i = 1, 2, \ldots, n$. Arrange these data in the form:

$$\begin{bmatrix} \mathbf{P}_1 & \mathbf{P}_2 & \ldots & \mathbf{P}_n \\ 1 & 1 & \ldots & 1 \end{bmatrix} = [A] \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \ldots & \mathbf{p}_n \\ 1 & 1 & \ldots & 1 \end{bmatrix}, \quad (38)$$

where the unknown 4×4 displacement matrix A is

$$[A] = \begin{bmatrix} \mathbf{R} & \mathbf{d} \\ 0 & 1 \end{bmatrix}. \quad (39)$$

Writing this relationship symbolically, we get a generalization of Eq. (23) as

$$[B]_{4\times n} = [A]_{4\times 4}[C]_{4\times n}, \quad (40)$$

where the $4 \times n$ matrices B and C contain the measured data for the n marker points. The 4×4 matrix A, which is to be determined, has a special form in that its 3×3 principal minor is the (orthogonal) rotation matrix R and its fourth row is (0, 0, 0, 1).

To apply the techniques of modern linear algebra, especially to take advantage of available software such as MATLAB, IMSL or EISPACK, let us rewrite the above matrix equation as

$$[C^t]_{n\times 4}[A^t]_{4\times 4} = [B^t]_{n\times 4}. \quad (41)$$

This is an overdetermined system of equations, i.e., the number of equations ($=4n$) exceeds the number of unknowns ($=16$). Because the last columns of C^t and B^t are always identical, the last column of A^t must be $(0, 0, 0, 1)^t$, and we can conclude that there are only 12 unknown elements in matrix A. However, in order to utilize powerful tools of modern linear algebra, we will let this result fall out naturally from the linear algebra solution. If there are any deviations form these expected values, then we can get an estimate of round-off errors of numerical computation.

A reliable and stable solution of the overdetermined system of equations is by the singular value decomposition (SVD) method. The singular value decomposition of the $n \times 4$ matrix C^t is (Strang, 1988; Press, *et al.* 1992).

$$C^t = \mathcal{Q}_1\mathcal{D}\mathcal{Q}_2^t, \quad (42)$$

where $\mathcal{Q}_1$ is an $n \times 4$ column-orthogonal matrix, $\mathcal{D}$ is a 4×4 diagonal matrix whose elements are the nonnegative singular values σ_i, and $\mathcal{Q}_2$ is a 4×4 orthogonal matrix. Thus, note that

$$\mathcal{Q}_1^t\mathcal{Q}_1 = I_{4\times 4}, \quad \text{but} \quad \mathcal{Q}_1\mathcal{Q}_1^t \neq I_{n\times n}, \quad (43)$$

and

$$\mathcal{Q}_2^t\mathcal{Q}_2 = \mathcal{Q}_2\mathcal{Q}_2^t = I_{4\times 4}. \quad (44)$$

The inverse of the diagonal matrix $\mathcal{D}$, i.e., $\mathcal{D}^{-1}$, will have elements $(1/\sigma_i)$, $\sigma_i \neq 0$. If some of the singular values σ_k are zero, or close to machine precision, then $\mathcal{D}^{-1}$ does not really exist but the corresponding entries $(1/\sigma_k)$ are replaced by zero, not infinity, and such a modified inverse matrix is called $\mathcal{D}^+$. If all of the singular values are nonzero, i.e., positive, then $\mathcal{D}^+ \equiv \mathcal{D}^{-1}$; otherwise, some internal dependencies exist among the data and the use of $\mathcal{D}^+$ instead of nonexistent $\mathcal{D}^{-1}$ eliminates these. Substituting the SVD of C^t into the overdetermined system (41), and simplifying by using the appropriate orthogonality conditions (43–44), we get the solution for the 4 × 4 displacement matrix A as (Gupta, 1997)

$$A = B\mathcal{Q}_1\mathcal{D}^+\mathcal{Q}_2^t. \tag{45}$$

As mentioned earlier, if the last row of A is not exactly (0, 0, 0, 1), then that is an indication of the magnitude of the round-off errors in computations. On the other hand, if the 3 × 3 principal minor (T) of matrix A is not exactly orthogonal, i.e., $T^tT \neq I_{3\times3}$, then that is an indication of the experimental errors in the point coordinate data.

To estimate these experimental errors, we factorize the 3 × 3 matrix T further by using the qr-decomposition such that

$$[T]_{3\times3} = [R']_{3\times3}[U']_{3\times3}, \tag{46}$$

where R' is an orthogonal matrix ($R'^tR' = I_{3\times3}$), and U' is an upper triangular matrix which contains an estimate of the experimental errors ϵ_{ij}. In fact, the elements of matrix U' can be written as

$$u'_{ij} = \pm\delta_{ij} + \epsilon_{ij}, \quad i = 1, 2, 3; \quad j = 1, 2, 3, \tag{47}$$

where δ_{ij} is the Kronecker's delta ($\delta_{ij} = 1$ for $i = j$; $\delta_{ij} = 0$ for $i \neq j$) and $\epsilon_{ij} = 0$ for lower triangular elements ($i > j$). The normal expectation would be for the matrix U' to be nearly an identity matrix, but its diagonal elements are usually ($\pm1 + \epsilon_{ii}$). This is because many sophisticated methods for qr-decomposition yield an orthogonal matrix R' with the determinant of (+1) or (−1). Since we are interested only in rotation matrices (i.e., orthogonal matrices with determinant +1), define matrix U with elements

$$u_{ij} = u'_{ij}\,|_{\epsilon_{ij}=0} \tag{48}$$

and then the rotation matrix R as

$$R = R'U. \tag{49}$$

In absence of experimental errors, the matrix U' would be an identity matrix $I_{3\times3}$. The qr-decomposition process thus extracts experimental errors ($\epsilon_{ij}, i \leq j$) from 3 × 3 matrix T and produces a good estimate R of the true rotation matrix. The screw parameters θ, s, $\mathbf{u}$, $\mathbf{Q}$ are then determined from the rotation matrix R and vector $\mathbf{d}$.

SVD/QS-Decomposition Method

The steps defined by equations (38–45) are still followed. However, an alternate decomposition for the 3 × 3 principal minor T of matrix A in Eq. (45) is used. Let us first discuss an undesirable characteristics of the qr-decomposition in equation (46).

The factorization $T = R'U'$ is unbalanced in that the lower triangular elements ($i > j$) of matrix U' are all zero. Thus the errors ϵ_{ij} are contained only in the diagonal elements ($i = j$) and the upper triangular elements ($i < j$) of matrix U'. Furthermore, the qr-factorization process implies certain operational priority for the columns of U', which usually is: column 1, column 2, column 3; if a standard software is used, then the user does to have control over this column priority issue. Because of these two reasons, the matrix U' isolates errors in somewhat unbalanced and artificial way. The error matrix E can be defined as

$$E = U' - U = (\epsilon_{ij})_{3\times3} \tag{50}$$

An alternative would be to require the error matrix E to have a more balanced structure, and more specifically, a symmetric structure ($\epsilon_{ij} = \epsilon_{ji}$). This is accomplished through a qs-decomposition, or polar decomposition, of $T_{3\times3}$ (Strang, 1988). First, SVD-decomposition of $T_{3\times3}$ is found as

$$T = \mathcal{Q}_1\mathcal{D}\mathcal{Q}_2^t. \tag{51}$$

All matrices in Eq. (51) are of size 3 × 3; both $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are orthogonal matrices and $\mathcal{D}$ is a diagonal matrix with nonnegative elements. Since matrix T represents an estimate of the rotation matrix, det $|T| \approx +1$; also, det $|\mathcal{D}| > 0$. Thus,

$$\det|\mathcal{Q}_1| = \det|\mathcal{Q}_2| = \pm1. \tag{52}$$

Using the orthogonality of $\mathcal{Q}_2$: $\mathcal{Q}_2^t\mathcal{Q}_2 = I$, Eq. (51) is rearranged as

$$T = (\mathcal{Q}_1\mathcal{Q}_2^t)(\mathcal{Q}_2\mathcal{D}\mathcal{Q}_2^t) = RV. \tag{53}$$

In this decomposition, $R = \mathcal{Q}_1\mathcal{Q}_2^t$ is the desired 3 × 3 rotation matrix (orthogonal matrix with +1 determinant, in view of equation (52)) and $V = \mathcal{Q}_2\mathcal{D}\mathcal{Q}_2^t$ is a symmetric matrix whose elements are

$$v_{ij} = \delta_{ij} + \epsilon_{ij}, \; \epsilon_{ij} = \epsilon_{ji}. \tag{54}$$

The diagonal elements are ($1 + \epsilon_{ii}$) and the problem which occurs in equation (47) of the previous method does not occur here. The symmetric error matrix is

$$E = V - I = (\epsilon_{ij})_{3\times3}. \tag{55}$$

The parameters θ, s, $\mathbf{u}$, $\mathbf{Q}$ are then found form R and $\mathbf{d}$.

Measures of Error

Errors in scalars $a = \theta$ or s are found as

$$\%\text{ error in ``}a\text{''} = \frac{|a_2 - a_1|}{|a_1|} \times 100\%. \tag{56}$$

Error in vectors $\mathbf{v} = \mathbf{d}$, $\mathbf{u}$ or $\mathbf{Q}$ are found by using L_2 or L_∞ vector norms as

$$\%\text{ error in ``}\mathbf{v}\text{''} = \frac{\|\mathbf{v}_2 - \mathbf{v}_1\|}{\|\mathbf{v}_1\|} \times 100\%. \tag{57}$$

Errors in matrices $M = R_{3\times3}$ or $A_{4\times4}$ are found by using L_2 or L_∞ matrix norms as

$$\%\text{ error in ``}M\text{''} = \frac{\|M_2 - M_1\|}{\|M_1\|} \times 100\%. \tag{58}$$

In these definitions, the first value (a_1 or $\mathbf{v}_1$ or M_1) is either the known exact value or a *priori* estimate of the exact value. It is only available in numerically created perturbation tests. The second value (a_2 or $\mathbf{v}_2$ or M_2) is obtained by the proposed methods.

Results form the two new methods, using MATLAB, were compared with the following "basic" method assembled from textbooks (Suh and Radcliffe, 1978; Bottema and Roth, 1979; Gupta, 1997):

For $n = 4$, solve Eq. (23), symbolically as

$$A_{4\times4} = BC^{-1}. \tag{59}$$

or, for $n > 4$, solve Eq. (40) by least squares as

$$A_{4\times4} = BC^t(CC^t)^{-1}. \tag{60}$$

In numerical examples, Eq. (60) was used for $n \geq 4$. From 3 × 3 principal minor $T_{3\times3}$ of $A_{4\times4}$, use standard formulae (Beggs, 1966; Bottema and Roth, 1979) to find

$$\theta = \text{arccosine}\left(\frac{t_{11} + t_{22} + t_{33} - 1}{2}\right). \tag{61}$$

$$u_x = \frac{t_{32} - t_{23}}{2\sin\theta}, \; u_y = \frac{t_{13} - t_{31}}{2\sin\theta}, \; u_z = \frac{t_{21} - t_{12}}{2\sin\theta}. \tag{62}$$

Finally, solve the following four linear equations in s, Q_x, Q_y, Q_z:

$$s\mathbf{u} - (R - I)\mathbf{Q} = \mathbf{d}$$
$$\mathbf{u}^t\mathbf{Q} = 0 \qquad (63)$$

Examples and Discussion

To test the robustness of these new methods, the following parametric example was developed. A set of eleven body points ($\mathbf{p}_i$) were chosen on or near the plane defined by $x + y + z = 1$. These points were:

(1, 0, 0), (0, 1, 0), (0, 0, 1)
$(0.391932601, 0.409863047, 0.198204352 + \epsilon)$
$(0.211229909, 0.068227469, 0.720542622 + \epsilon)$
$(0.141077943, 0.199071845, 0.659860212 + \epsilon)$
$(0.096983736, 0.300505054, 0.602511210 + \epsilon)$
$(0.005658113, 0.088440184, 0.905901702 + \epsilon)$
$(0.095911919, 0.414177359, 0.489910723 + \epsilon)$
$(0.491617899, 0.078865590, 0.429516511 + \epsilon)$
$(0.122027123, 0.493968500, 0.384004376 + \epsilon)$

The first three points lie exactly on the plane $x + y + z = 1$, and the remaining eight points on the plane $x + y + z = 1 + \epsilon$. The value of ϵ was changed from 0.1 to 10^{-16}. The machine precision was 10^{-16}. A known displacement matrix D_{exact},

$$D_{\text{exact}} = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 & 1 \\ 0 & 0 & 1 & 2 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

was used to generate the corresponding global coordinates ($\mathbf{P}_i$) by using equation 938). With matrices B and C thus formed, D_{approx} was obtained by the "basic" method (Eq. (60)), the SVD/QR decomposition method (Eqs. (38–49)) and SVD/QS decomposition method (equations (38–45) and (51–53)). The parameters θ, s, $\mathbf{u}$, $\mathbf{Q}$ were obtained by Eqs. (61–62), the errors in D, R, θ, s, $\mathbf{u}$, $\mathbf{Q}$ were found by using Eqs. (56–58), and the "worst percent error" was flagged.

Since the eleven chosen points ($\mathbf{p}_i$ or $\mathbf{P}_i$) were nearly coplanar, numerical problems were expected for small values of ϵ. For the values of ϵ in the range 10^{-1} to 10^{-4}, all three methods gave identical results. For $\epsilon = 10^{-5}$ and 10^{-6}, the "basic" method had (worst) errors of the order of 0.002 percent and 0.02 percent, respectively, while the SVD/QR and SVD/QS decomposition methods worked fine (i.e. negligible percent errors). For $\epsilon = 10^{-7}$, the "basic" method produced (worst) errors of the order of 37 percent, while the SVD/QR and SVD/QS decomposition methods continued to work fine.

From $\epsilon = 10^{-8}$, the "basic" method broke down and the D_{approx} matrix that it produced no longer represented a rigid body transformation/displacement. However, the SVD/QR and SVD/QS decomposition methods produced the (worst) errors of the orders of 0.000004 percent only.

From $\epsilon = 10^{-8}$ to 10^{-16} (machine precision), the SVD/QR and SVD/QS decomposition methods worked well. The (worst) errors increased in the range of $\epsilon = 10^{-8}$ to 10^{-12} up to a value of 0.00005%. Then, unexpectedly, they declined in the range of $\epsilon = 10^{-13}$ to 10^{-16} (machine precision); for $\epsilon = 10^{-16}$ (machine precision), the (worst) error for the SVD/QR method was only 0.000011 percent, and for the SVD/QS method it was only 0.0000094 percent.

Although both the SVD/QR and SVD/QS decomposition gave excellent results on an absolute basis, the latter method gave slightly better results for $\epsilon \leq 10^{-5}$.

The maximum element of $\mathcal{D}^+$ in SVD decomposition (45) was also monitored. It increased form 14.923 at $\epsilon = 10^{-1}$ to 1,191,133,519.166 at $\epsilon = 10^{-16}$. It changed rapidly in the range of $\epsilon = 10^{-1}$ to 10^{-9}, but did not change significantly in the range of $\epsilon = 10^{-10}$ to 10^{-16} where it was slightly above the value of 1.19×10^9. In $\mathcal{D}^+$, we left the largest element alone. In SVD theory, it is recommended that if the smallest element of $\mathcal{D}$ is zero, then the corresponding element in $\mathcal{D}^+$ is also zero, not infinity. We found that if the largest element of $\mathcal{D}^+$, even when it became as large as 10^9, was replaced by zero, the A matrix in Eq. (45) lost its rigid body transformation/displacement structure. Our recommendation is therefore to keep $\mathcal{D}^+$ as it is found naturally by the SVD decomposition.

This special example was constructed to test the limits of the three methods. With machine precision of 10^{-16}, and eleven nearly coplanar points, we have found that the "basic" method does not work beyond $\epsilon = 10^{-7}$, while both SVD/QR and SVD/QS decomposition methods work fine up to the machine precision ($\epsilon = 10^{-16}$). On a relative basis, the SVD/QS method gives slightly better results than the SVD/QR method and that is probably due to a better (symmetric) distribution of errors in the error matrix of equation (55).

In a second example to test reliability, eleven exact noncoplanar body point coordinates ($\mathbf{p}_i$) and the corresponding exact coordinates ($\mathbf{P}_i$) were randomly perturbed with errors of ± 0.05. The D_{exact} used was same as that in the previous example. Matrix D_{approx} and the corresponding parameters θ, s, $\mathbf{u}$, $\mathbf{Q}$ were found by the "basic" method, the SVD/QR method and the SVD/QS method, and compared with the known exact values. It was found that the (worst) error was 59.25 percent for the "basic" method, 5.1 percent for the SVD/QR method and 3.97 percent for the SVD/QS method. Therefore, in the presence of small random errors in the point coordinate data ($\mathbf{p}_i$ and $\mathbf{P}_i$), both SVD/QR and SVD/QS methods gave results which were much closer to the known exact values (of D, R, θ, s, $\mathbf{u}$, $\mathbf{Q}$) than the "basic" method; on a relative basic, the SVD/QS method gave slightly better results than the SVD/QR method. Other examples, including a statistical evaluation, are presented in Chutakanonta (1998).

Conclusion

Two new methods for obtaining object position from imprecise and excess point coordinate data have been presented. Both methods exploit the sophisticated matrix algebra capabilities of the readily available software such as MATLAB, IMSL and EISPACK. The first method is called SVD/QR Decomposition Method because it uses the stable SVD decomposition, followed by qr-decomposition. The second method is called SVD/QS Decomposition. Both methods have been shown to be robust by considering an example with nearly dependent point coordinates. In the presence of small random errors in the point coordinate data (body and global), both methods can approximate the underlying position matrix and parameters with good accuracy. While both methods produced good results on an absolute basis, between the two methods, the SVD/QS method gave slightly better results than the SVD/QR method due to better distribution of the extracted errors in the former.

References

Angeles, J., *Spatial Kinematic Chains,* Springer-Verlag, Berlin, 1982.

Angeles, J., "Automatic Computation of the Screw Parameters of Rigid Body Motions. Part I: Finitely-Separated Positions," ASME Journal of Dynamic Systems, Measurement and Control, Vol. 108, No. 1, pp. 323–38, 1986.

Beggs, J. S., *Advanced Mechanism,* Macmillan, New York, 1966.

Beggs, J. S., *Kinematics,* Hemisphere, Washington, 1983.

Bottema, O., and Roth, B., *Theoretical Kinematics,* North Holland, Amsterdam, 1979.

Challis, J. H., "Examination of Procedures for Determining Body Segment Attitude and Position Data From Noisy Biomechanical Data," *Medical Engineering and Physics,* Vol. 17, No. 2, pp. 83–90, 1995.

Cheng, H., and Gupta, K. C., "An Historical Note on Finite Rotations," ASME Journal Applied Mechanics, Vol. 56, pp. 139–145, 1989.

Chutakanonta, P., ''An Analysis of the Accuracy of Object Position Determined from the Point Coordinate Data,'' M. S. Thesis, Mechanical Engineering, University of Illinois at Chicago, Chicago, IL, 1998.

Fenton, R. G., and Shi, X., ''Comparison of Methods for Determining Screw Parameters of Finite Rigid Body Motion from Initial and Final Position Data,'' *Proc. ASME Design Automation Conf.—Advances in Design Automation,* DE19(3), pp. 433–439, 1989.

Gupta, K. C., ''Measures of Positional Error for a Rigid Body,'' ASME Journal of Mechancial Design, Vol. 119, pp. 346–348, 1997.

Gupta, K. C., *Mechanics and Control of Robots,* Springer-Verlag, New York, 1997.

Heeren, T. A. G., and Veldpaus, F. E., ''Optical System to Measure the End Effector Position for On-Line Control Purposes,'' *International Journal of Robotics Research,* Vol. 11, No. 1, pp. 53–63, 1992.

Laub, A. J., and Shiflett, G. R., ''Linear Algebra Approach to the Analysis of Rigid Body Displacement from Initial and Final Position Data,'' ASME Journal of Applied Mechanics, Vol. 49, No. 1, pp. 213–216, 1982.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P., *Numerical Recipes in FORTRAN—The Art of Scientific Computing,* Cambridge University Press, Cambridge, 1992.

Shiflett, G. R., and Laub, A. J., ''The Analysis of Rigid Body Motion from Measured Data,'' ASME Journal of Dynamic Systems, Meausrement and Control, Vol. 117, No. 4, pp. 578–584, 1995.

Spoor, C. W., and Veldpaus, F. E., ''Rigid Body Motion Calculated from Spatial Coordinates of Markets,'' *Journal of Biomechanics,* Vol. 13, No. 4, pp. 391–393, 1980.

Strang, G., *Linear Algebra and its Applications,* Harcourt Brace Jovanovich, San Diego, 1988.

Suh, C. H., and Radcliffe, C. W., *Kinematics and Mechanisms Design,* John Wiley and Sons, New York, 1978.

Veldpaus, F. E., Woltring, H. J., and Dortmans, L. J. M. G., ''Least-Squares Algorithm for the Equiform Transformation from Spatial Marker Coordinates,'' *Journal of Biomechanics,* Vol. 21, No. 1, pp. 45–54, 1988.