[2] R. B. McGhee *et al.* "Real-time computer control of a hexapod vehicle," in *Proc 3rd CISM-IFToMM Symp. on Theory and Practice of Robots and Manipulators.* Amsterdam, The Netherlands: Elsevier, pp. 323-339, 1978.
[3] R. B. McGhee and G. I. Iswandhi, "Adaptive locomotion of a multilegged robot over rough terrain," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-9, no. 4, pp. 176-182, 1979.
[4] M. H. Raibert and I. E. Sutherland, "Machines that walk," *Scientific Amer.*, vol. 248, no. 1, pp. 44-53, 1983.
[5] M. H. Raibert and F. C. Wimberly, "Tabular control of balance in a dynamic legged system," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-14, no. 3, 1984.
[6] M. H. Raibert *et al.*, "Dynamically stable legged locomotion," Carnegie-Mellon Univ., Robotics Inst., Third Annual Report, Tech. Rep. CMU-RI-TR-83, 1983.
[7] M. H. Raibert *et al.*, "Experiments in balance with a 3D one-legged hopping machine," *Int. J. Robotics Res.*, vol. 3, no. 2, pp. 75-92, 1984.
[8] D. M. Wilson, "Insect walking," *Annu. Rev. Entymol.*, vol. 11, pp. 103-121, 1966.
[9] S. Hirose, "A study of design and control of a quadruped walking vehicle," *Int. J. Robotics Res.*, vol. 3, no. 2, pp. 113-133, 1984.
[10] H. Hemani and R. L. Farnsworth, "Postural and gait stability of a planar five link biped by simulation," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 452-458, 1977.
[11] D. E. Orin, "Interactive control of six-legged vehicle with optimization of both stability and energy," Ph.D. dissertation, The Ohio State Univ., Columbus, 1976.
[12] ——, "Supervisory control of a multilegged robot," *Int. J. Robotic Res.*, vol. 1, no. 1, pp. 79-91, 1982.
[13] S. S. Sun, "A theoretical study of gaits for legged locomotion systems," Ph.D. dissertation, The Ohio State Univ., Columbus, 1974.
[14] A. P. Bessonov and N. V. Umnov, "The analysis of gaits in six-legged vehicles according to their static stability," in *Proc. Symp. on Theory and Practice of Robots and Manipulators* (Udine, Italy, 1973).
[15] F. Ozguner, S. J. Tsai, and R. B. McGhee, "An approach to the use of terrain-preview information in rough-terrain locomotion by a hexapod walking machine," *Int. J. Robotics Res.*, vol. 3, no. 2, pp. 134-146, 1984.
[16] E. I. Kugusheve and V. S. Jaroshevskij, "Problems of selecting a gait for an integrated locomotion robot," paper presented at the 4th Int. Conf. on Artificial Intelligence, Tbilisi, Georgian SSR, USSR, 1975.
[17] K. Ikeda *et al.*, "Finite state control of quadruped walking vehicle—Control by hydraulic digital actuator" (in Japanese), *Biomechanism*, vol. 2, pp. 164-172, 1973.
[18] T. T. Lee and C. L. Shy, "A study of the gait control of quadruped walking vehicle," *IEEE J. Robotics Automat.*, vol. RA-2, no. 2, pp. 61-69, 1986.
[19] S. M. Song, "Kinematic optimal design of a six-legged walking machine," Ph.D. dissertation, The Ohio State University, Columbus, 1984.
[20] T. T. Lee and C. M. Liao, "On the hexapod crab walking tripod gaits," in *Proc. 15th Int. Symp. on Industrial Robots* (Tokyo, Japan, 1985), vol. 1, pp. 263-270.

# Closed-Loop Manipulator Control Using Quaternion Feedback

JOSEPH S.-C. YUAN

*Abstract*—**Euler parameters, a form of normalized quaternions, are used here to model the hand orientation errors in resolved rate and resolved acceleration control of manipulators. The quaternion formulation greatly simplifies the stability analysis of the orientation error dynamics. Two types of quaternion feedback have been considered. The first type uses only the vector portion of the quaternion error, while the second one is based on a Euler rotation representation. The quaternion vector approach leads to a linear feedback control law for which the global asymptotic convergence of the orientation error is readily established. The Euler rotation approach also results in asymptotic error convergence in the large except for a singularity where the hand orientation differs from its desired orientation by a rotation of 180°.**

Manuscript received February 26, 1987; revised October 28, 1987.
The author is with Spar Aerospace Ltd., Weston, Ont., Canada M9L 2W7.
IEEE Log Number 8820162.

## I. Introduction

Manipulators with six or more degrees of freedom are generally required to follow preplanned paths of hand position and orientation defined as a function of time in Cartesian (or task space) coordinates. For closed-loop control of resolved motion, the instantaneous motion of the hand, or end-effector, must be monitored continuously either by using direct endpoint sensing techniques (e.g., [1]) or via a kinematic model of the manipulator which computes the hand position and orientation from the joint variables. This information is, in turn, used to produce corrective control action from the joint actuators in the manipulator (Fig. 1).

The hand position and orientation of a manipulator are typically represented by the position vector and rotation matrix, respectively, between reference coordinate frames fixed to the base and the last link of the manipulator [2]. The rotation matrix has the general form

$$R_{OH} = [\boldsymbol{n}\ \boldsymbol{s}\ \boldsymbol{a}] \tag{1}$$

where $\boldsymbol{n}$, $\boldsymbol{s}$, and $\boldsymbol{a}$ are the ***normal***, ***slide***, and ***approach*** (unit) vectors of the hand frame expressed in base frame coordinates.

It is clear that the position vector $\boldsymbol{p}$ and its derivatives ($\dot{\boldsymbol{p}}$ and $\ddot{\boldsymbol{p}}$ for velocity and acceleration, respectively) completely describe the translational motion of the hand. The position tracking error may be defined as

$$e_p = \boldsymbol{p} - \boldsymbol{p}_d \tag{2}$$

where $\boldsymbol{p}_d$ denote the desired hand position vector. The velocity and acceleration errors can be defined accordingly as $\dot{e}_p = (\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d)$ and $\ddot{e}_p = (\ddot{\boldsymbol{p}} - \ddot{\boldsymbol{p}}_d)$, respectively.

If $\boldsymbol{\omega}$ and $\dot{\boldsymbol{\omega}}$ denote the angular velocity and acceleration of the hand, then the corresponding error terms may be defined as $\dot{e}_0 = (\boldsymbol{\omega} - \boldsymbol{\omega}_d)$ and $\ddot{e}_0 = (\dot{\boldsymbol{\omega}} - \dot{\boldsymbol{\omega}}_d)$, where $\boldsymbol{\omega}_d$ and $\dot{\boldsymbol{\omega}}_d$ denote the desired angular velocity and angular acceleration, respectively, of the hand. The question that arises now is: What is an appropriate counterpart for $\boldsymbol{p}$ which represents $\int \boldsymbol{\omega}\, dt$ in the following definition of orientation tracking error?

$$e_0 = \int \boldsymbol{\omega}\, dt - \int \boldsymbol{\omega}_d\, dt. \tag{3}$$

This question takes on particular significance in the context of closed-loop manipulator control since the position and orientation errors of the hand are used explicitly in the feedback loop.

When the manipulator is controlled in its joint coordinates [3, ch. 7], there is no need to generate the hand orientation error as in (3) since the desired trajectory, represented by the hand position and rotation matrix, may be converted directly into a corresponding trajectory in the joint coordinates. The inverse kinematic models used for such a conversion, however, are typically very complex and exist as closed-form solutions only for manipulators with special configurations, such as parallel adjacent joints or spherical wrists [3, ch. 3].

For orientation error feedback, the rotation matrix representation of (1) is clearly impractical simply because there are too many elements in the matrix. More importantly, not all of its elements (which are direction cosines) are independent due to the requirement of orthogonality among the unit vectors $\boldsymbol{n}$, $\boldsymbol{s}$, and $\boldsymbol{a}$.

Despite their nonuniqueness, Euler angles are frequently used to represent orientation [4, ch. 2.1.1] mainly because of their physical manifestations, such as roll, pitch, and yaw angles. But they are undesirable in feedback control due to singularities and computational complexity [3, ch. 3.2]. Furthermore, because the differential

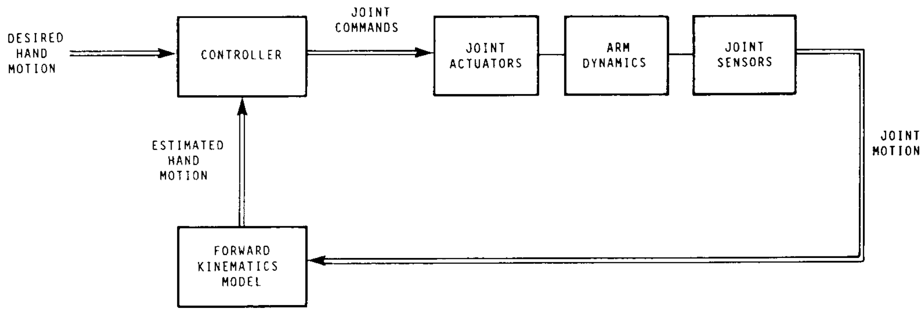


Fig. 1. Closed-loop resolved motion control.

equations relating the rates of change of these angles to $\boldsymbol{\omega}$ are highly nonlinear [4, p. 30], it is extremely difficult to analyze the stability of the closed-loop system without using some form of small-angle linear approximation.

Alternatively, the difference in orientation between two coordinate systems may be expressed as a single rotation through some angle about a fixed axis as stated in Euler's Theorem [4, p. 13]. Such a rotation (known as *Euler rotation*) represents the minimal angular distance between the two systems. This formulation has a natural appeal from the control standpoint since the rotation vector coincides with the axis about which the control torque must be applied to the hand.

The Euler rotation representation has been used in both resolved rate [5] and resolved acceleration control [6]. Furthermore, it has been applied successfully to the control of the space shuttle manipulator [7]. In the latter case, the instantaneous error between the actual and the desired hand orientation is described as a rotation of $\varphi(t)$ about a unit vector $\boldsymbol{r}(t)$. The angular velocity command for the hand is then aligned with $\boldsymbol{r}$. The feedback gain, however, must vary continuously with the orientation error and is highly nonlinear which renders the stability analysis of the closed-loop system an intractable task.

It was stated in [6] that when the orientation error is small, it may be expressed in terms of the Euler rotation parameters ($\varphi$, $\boldsymbol{r}$) as

$$\boldsymbol{e}_0(t) = \sin\varphi(t)\boldsymbol{r}(t). \tag{4}$$

Unfortunately, the convergence analysis offered in [6] is based on linear approximation over small time intervals and does not apply to stability in the large (i.e., when there is a large initial orientation error). It will be shown in this communication that the Euler rotation representation (4) in fact leads to convergence even for large orientation errors.

Yet another representation for orientation is *Euler parameters* which are a form of normalized quaternions [4, p. 23]. (For economy in notation, the generic term "quaternion" will be used in this communication to denote Euler parameters.) Though less amenable to physical interpretation than either Euler angles or rotation matrices, quaternions are free of singularities and are computationally more efficient. Many applications of quaternions can be found in the literature on large-angle maneuvering and attitude control of space vehicles [8]–[13]. More recently, the quaternion formulation has been applied to the dynamic analysis of a general class of mechanisms [14], [15] which could conceivably include manipulators.

This communication extends the quaternion concept to the resolved motion control of manipulators. It will be shown that the quaternion formulation can greatly simplify the stability analysis of the orientation error dynamics.

## II. Basic Properties of Quaternions

Quaternions have many interesting properties, not all of which, however, are relevant to the development of this communication. (A list of general identities involving quaternions can be found in [14].) Only some of their fundamental features are discussed below.

### A. Definition

Let two coordinate systems ($\mathfrak{F}_0$ and $\mathfrak{F}_1$) be separated by a rotation of $\varphi$ about a unit vector $\boldsymbol{r}$ as defined in Euler's Theorem. Then the quaternion description of the orientation difference between the two systems is given by a scalar $\eta$ and a vector $\boldsymbol{q}$ defined as follows:

$$\eta = \cos(\varphi/2) \qquad \boldsymbol{q} = \sin(\varphi/2)\boldsymbol{r}. \tag{5}$$

Note that if $\{\eta, \boldsymbol{q}\}$ is the quaternion representation of $\mathfrak{F}_1$ relative to $\mathfrak{F}_0$, then $\{\eta, -\boldsymbol{q}\}$ represents the orientation of $\mathfrak{F}_0$ relative to $\mathfrak{F}_1$.

### B. Normality

It is clear from (5) that

$$\eta^2 + \boldsymbol{q}^T\boldsymbol{q} = 1 \tag{6}$$

where a superscript "$T$" denotes vector or matrix transpose.

### C. Uniqueness

Both $\{\eta, \boldsymbol{q}\}$ and $\{-\eta, -\boldsymbol{q}\}$ describe the same orientation. But if the rotation angle $\varphi$ is confined to the range $-180° \le \varphi \le 180°$, then the scalar $\eta$ is nonnegative and the quaternion representation is unique.

### D. Relation to Rotation Matrix

The direction cosine matrix describing the rotation sequence that brings $\mathfrak{F}_0$ onto $\mathfrak{F}_1$ is given by [16, p. 421]

$$\boldsymbol{R}_{10} = \cos\varphi\, \boldsymbol{I} + (1-\cos\varphi)\boldsymbol{r}\boldsymbol{r}^T - \sin\varphi\, \boldsymbol{r}^\times \tag{7}$$

where $\boldsymbol{I}$ denotes a unit matrix and

$$\boldsymbol{r} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} \qquad \boldsymbol{r}^\times = \begin{bmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{bmatrix}. \tag{8}$$

It can be seen, from (5), that the rotation matrix may also be written in terms of the quaternion parameters as

$$\boldsymbol{R}_{10} = (\eta^2 - \boldsymbol{q}^T\boldsymbol{q})\boldsymbol{I} + 2\boldsymbol{q}\boldsymbol{q}^T - 2\eta\boldsymbol{q}^\times \tag{9}$$

where the matrix $\boldsymbol{q}^\times$ is defined in terms of the components of $\boldsymbol{q}$ in a similar manner to $\boldsymbol{r}^\times$ in (8). An efficient singularity-free algorithm for computing the quaternion from a rotation matrix is described in [17].

The quaternion vector $\boldsymbol{q}$ has the same coordinates in either $\mathfrak{F}_0$ or $\mathfrak{F}_1$. Indeed, it is the eigenvector of $\boldsymbol{R}_{10}$ with unit eigenvalue; that is,

$$\boldsymbol{R}_{10}\boldsymbol{q} = \boldsymbol{R}_{10}^T\boldsymbol{q} = \boldsymbol{q}. \tag{10}$$

### E. Quaternion Propagation

Suppose the coordinate system $\mathfrak{F}_1$ rotates at an instantaneous angular velocity of $\boldsymbol{\omega}$ about $\mathfrak{F}_0$. Then the quaternion $\{\eta, \boldsymbol{q}\}$ evolves in time according to the following differential equation [4, p. 32]:

$$\begin{bmatrix} \dot{\eta} \\ \dot{q} \end{bmatrix} = 1/2 \begin{bmatrix} 0 & -\omega^T \\ \omega & -\omega^\times \end{bmatrix} \begin{bmatrix} \eta \\ q \end{bmatrix} \tag{11}$$

where $\omega$ is expressed in the coordinates of $\mathfrak{F}_1$ and the matrix $\omega^\times$ is derived from $\omega$ in the same way $r^\times$ is defined for $r$ in (8). Further discussions on quaternion propagation can be found in [18].

### F. Relative Orientation

Consider two coordinate systems, $\mathfrak{F}_1$ and $\mathfrak{F}_2$, and let $R_{10}$ and $R_{20}$ denote the rotation matrices describing the orientation of each system referenced to a common base frame $\mathfrak{F}_0$. Suppose the corresponding quaternion representations are given by $\{\eta_1, q_1\}$ and $\{\eta_2, q_2\}$, respectively. Then the relative orientation between $\mathfrak{F}_1$ and $\mathfrak{F}_2$, denoted by the rotation matrix

$$R_{21} = R_{20} R_{10}^T \tag{12}$$

is given by $\{\delta\eta, \delta q\}$ where [9]

$$\delta\eta = \eta_1\eta_2 + q_1^T q_2 \qquad \delta q = \eta_1 q_2 - \eta_2 q_1 - q_1^\times q_2. \tag{13}$$

Note that $\delta q$ is expressed here in the coordinates of either $\mathfrak{F}_1$ or $\mathfrak{F}_2$ (since they are the same), but not of $\mathfrak{F}_0$.

### G. Orientation Error Representation

If $\mathfrak{F}_1$ and $\mathfrak{F}_2$ denote the desired and the actual hand orientation relative to the base of the manipulator, then (13) yields the quaternion for the orientation error. When the two frames coincides, $\eta_1 = \eta_2$ and $q_1 = q_2$, we get, through (6),

$$\delta\eta = 1 \qquad \delta q = 0. \tag{14}$$

Conversely, at $\delta q = 0$,

$$\eta_1 q_2 - \eta_2 q_1 = q_1^\times q_2. \tag{15}$$

But the vectors $(\eta_1 q_2 - \eta_2 q_1)$ and $q_1^\times q_2$ are orthogonal to each other. (The product $q_1^\times q_2$ is, in fact, a matrix-vector representation of the cross product $q_1 \times q_2$.) Therefore, (15) holds only if both vectors are zero; in other words,

$$q_2 = (\eta_2/\eta_1) q_1. \tag{16}$$

Furthermore, through the normality condition (6), $\delta q = 0$ implies

$$\delta\eta = \eta_1\eta_2 + q_1^T q_2 = \pm 1. \tag{17}$$

Substituting (16) into (17) and making use of the normality of $\{\eta_1, q_1\}$, we get

$$\{\eta_1, q_1\} = \{\pm\eta_2, \pm q_2\}$$

both of which describe the same orientation (cf. property in Subsection II-C, above).

Hence, we have established the following result which states that $\delta q$ is indeed a logical representation for the orientation error between two coordinate systems:

*Proposition 1:* Two coordinate systems coincide if, and only if, $\delta q = 0$, where $\delta q$ is the vector component of the quaternion defined in (13).

Note that, unlike the case cited in [6] with the Euler rotation representation (4), the above result holds regardless of the size of the orientation error.

## III. Closed-Loop Resolved Rate Control

Let $\dot{p}$ and $\omega$ be the linear and angular velocity, respectively, of the hand of a manipulator with $n$ links, and denote the joint rates by

$$\dot{\theta} = [\dot{\theta}_1 \cdots \dot{\theta}_n]^T.$$

Then the hand velocities are related to $\dot{\theta}$ by the expression [5]

$$\begin{bmatrix} \dot{p} \\ \omega \end{bmatrix} = J\,\dot{\theta} \tag{18}$$

where $J$ is a $6 \times n$ Jacobian matrix whose elements are functions of the joint variables

$$\theta = [\theta_1 \cdots \theta_n]^T.$$

In open-loop resolved rate control [19], the joint rate command is simply given by

$$\dot{\theta}_C = J^\dagger \begin{bmatrix} \dot{p}_d \\ \omega_d \end{bmatrix} \tag{19}$$

where $\dot{p}_d$ and $\omega_d$ are the desired linear and angular hand velocities, and $J^\dagger$ denotes either the direct inverse ($n = 6$) or the generalized inverse ($n > 6$) of $J$.

In closed-loop control [20], however, the control law (19) is replaced with

$$\dot{\theta}_C = J^\dagger \begin{bmatrix} \dot{p}_d - K_p e_p \\ \omega_d - K_0 e_0 \end{bmatrix} \tag{20}$$

where $K_p$ and $K_0$ are feedback gain matrices; $e_p$ and $e_0$ are the position and orientation errors of the hand defined in (2) and (3), respectively.

In most industrial robots, the command $\dot{\theta}_C$ is applied directly to the rate servos at the joints. This is known as kinematic control since the dynamics of the manipulator is completely ignored. For dynamic control, the rate command $\dot{\theta}_C$, together with the measured $\theta$ and estimated acceleration $\ddot{\theta}_C$, are used in a dynamic model of the manipulator (e.g., the Newton–Euler model of [21]), to compute the joint torques. This approach is commonly known as the *computed torque* method.

The computed torque algorithm of [21] accounts for the nonlinear joint inertia, Coriolis and centrifugal forces, as well as gravity. As a result, except for the effects of friction, actuator dynamics, and parametric errors, the linear and angular velocities of the hand can be represented, to the first order, by the following equations:

$$\dot{p} = \dot{p}_d - K_p e_p \tag{21}$$

$$\omega = \omega_d - K_0 e_0 \tag{22}$$

which have been obtained by applying (18) to (20). The convergence of the position and orientation errors is thus determined by the stability of the differential equations (21) and (22).

Since $\dot{e}_p = (\dot{p} - \dot{p}_d)$ and (21) is linear, it is easy to select the gain $K_p$ so that the position error $e_p$ converges to zero asymptotically (i.e., $e_p(t) \to 0$ as $t \to \infty$). We shall next examine the convergence of the orientation error in (22).

Let $\{\eta_d, q_d\}$ and $\{\eta, q\}$ denote, respectively, the quaternions for the desired and the actual orientation of the hand. Then each quaternion evolves in time according to a set of differential equations similar to (11) as follows:

$$\begin{bmatrix} \dot{\eta}_d \\ \dot{q}_d \end{bmatrix} = 1/2 \begin{bmatrix} 0 & -\omega_d^T \\ \omega_d & -\omega_d^\times \end{bmatrix} \begin{bmatrix} \eta_d \\ q_d \end{bmatrix} \tag{23}$$

$$\begin{bmatrix} \dot{\eta} \\ \dot{q} \end{bmatrix} = 1/2 \begin{bmatrix} 0 & -\omega^T \\ \omega & -\omega^\times \end{bmatrix} \begin{bmatrix} \eta \\ q \end{bmatrix}. \tag{24}$$

The asymptotic stability of the nonlinear system comprising (22)–(24) is best studied by using Lyapunov's second method [22]. For this we define the following positive-definite Lyapunov function:

$$V = (\eta - \eta_d)^2 + (q - q_d)^T (q - q_d). \tag{25}$$

It can be shown through substitution from (22)–(24) that the time

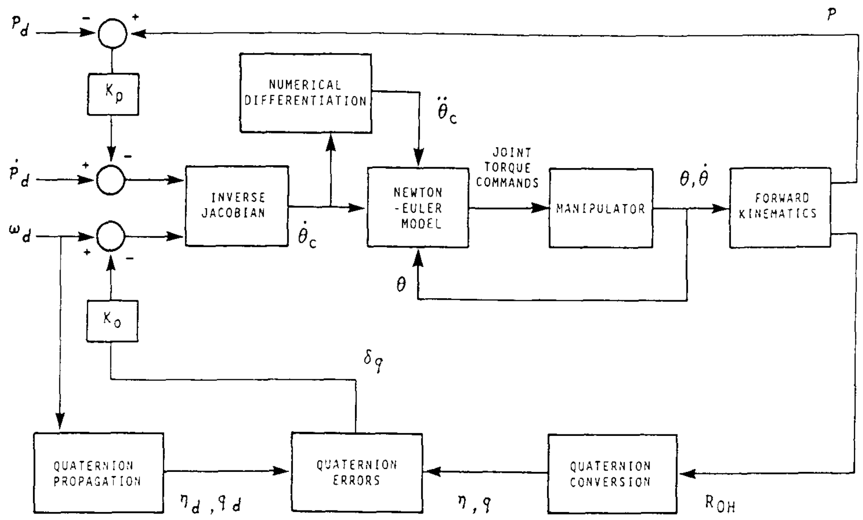


Fig. 2. Resolved rate control using quaternion feedback.

derivative of $V$ along any quaternion trajectory $\{\eta, q\}$ is given by

$$\dot{V} = \delta q^T(\omega - \omega_d) = -\delta q^T K_0 e_0 \tag{26}$$

where

$$\delta q = \eta_d q - \eta q_d - q_d^{\times} q. \tag{27}$$

One recognizes from (13) that the latter is simply the quaternion error vector between the desired and the actual hand orientation.

We are now faced with the task of selecting a representation for the orientation error feedback $e_0$ which will cause $\dot{V}$ in (26) to become negative-definite. An obvious choice is

$$e_0 = \delta q \tag{28}$$

with $K_0 > 0$ so that

$$\dot{V} = -\delta q^T K_0 \delta q. \tag{29}$$

Given an arbitrary nonzero $\delta q$, $\dot{V}$ is negative, which causes $V$ to approach zero since it is a positive-definite function. This will continue until an equilibrium point is reached where $\dot{V} = 0$, i.e., $\delta q = 0$. By Proposition 1, this also corresponds to zero orientation error. We therefore conclude that the quaternion error feedback given by (28) results in global asymptotic convergence of the orientation error.

The implementation of the quaternion error feedback (28) in resolved rate dynamic control is illustrated in Fig. 2. Given $\omega_d(t)$, the desired quaternion trajectory $\{\eta_d(t), q_d(t)\}$ can be either precomputed or generated on-line with (23). The instantaneous quaternion of the hand may be extracted from the rotation matrix (1) using the singularity-free algorithm described in [17].

Let us now study the stability of (22) with the error feedback $e_0$ expressed as a Euler rotation. When expressed in the form of (4), the relationship between $e_0$ and $(\omega - \omega_d)$ is not obvious and it becomes very difficult to determine the stability of (22). The convergence analysis given in [6] for the resolved acceleration case is valid only for short time intervals and small orientation errors. Described below is a stability analysis of (22) based on a quaternion formulation.

It follows from the definition of quaternions in (5) that the Euler rotation error (4) can also be expressed as

$$e_0 = \sin\varphi \, r = 2\cos(\varphi/2)\sin(\varphi/2) r = 2\delta\eta\delta q \tag{30}$$

where $\delta q$ is given by (27) and

$$\delta\eta = \eta_d \eta + q_d^T q. \tag{31}$$

Substituting (30) into (26), we have

$$\dot{V} = -2\delta\eta\delta q^T K_0 \delta q. \tag{32}$$

Since $\delta\eta$ is nonnegative (cf. property in Subsection II-C), $\dot{V} \le 0$ provided $K_0 > 0$. Thus there are two equilibrium points, $\delta\eta = 0$ and $\delta q = 0$, at which $\dot{V} = 0$. Of the two, however, only one ($\delta q = 0$) results in true convergence of the orientation error; $\delta\eta = 0$ occurs when the actual and the desired hand orientation are separated by a Euler rotation of 180°.

The above analysis clearly demonstrates the superiority of the quaternion error feedback (28) over the Euler rotation error feedback of (4) or (30).

## IV. Closed-Loop Resolved Acceleration Control

The acceleration of the hand can be obtained by differentiating (18) to yield

$$\begin{bmatrix} \ddot{p} \\ \dot{\omega} \end{bmatrix} = J\ddot{\theta} + \dot{J}\dot{\theta} \tag{33}$$

where $\dot{J}$ is the time derivative of the Jacobian matrix whose elements are functions of $\theta$ and $\dot{\theta}$. In resolved acceleration control [6], the joint acceleration command is given by

$$\ddot{\theta}_C = J^+ \begin{bmatrix} \ddot{p}_d - K_v(\dot{p} - \dot{p}_d) - K_p e_p \\ \dot{\omega}_d - K_\omega(\omega - \omega_d) - K_0 e_0 \end{bmatrix} - J^+ \dot{J} \dot{\theta} \tag{34}$$

where $K_v$, $K_\omega$, $K_p$, and $K_0$ are feedback gain matrices. $\ddot{p}_d$ and $\dot{\omega}_d$ are the desired linear and angular accelerations, and $J^+$ denotes the generalized inverse of $J$.

Equation (34) together with measured values of $\theta$ and $\dot{\theta}$ may then be used in a computed torque algorithm [21] to generate the joint torques. As a result, the linear and angular accelerations of the hand are given by (see comments just above (21) and (22))

$$\ddot{p} = \ddot{p}_d - K_v(\dot{p} - \dot{p}_d) - K_p e_p \tag{35}$$

$$\dot{\omega} = \dot{\omega}_d - K_\omega(\omega - \omega_d) - K_0 e_0. \tag{36}$$

Since the linear acceleration error is given by $\ddot{e}_p = (\ddot{p} - \ddot{p}_d)$, it is easy to choose the gain matrices $K_v$ and $K_p$ in (35) to ensure that $e_p(t) \to 0$ as $t \to \infty$. We shall now analyze the convergence of the orientation error in (36) by using a quaternion formulation.

Define the following Lyapunov function:

$$V = (\eta - \eta_d)^2 + (q - q_d)^T(q - q_d) + 1/2(\omega - \omega_d)^T(\omega - \omega_d). \tag{37}$$

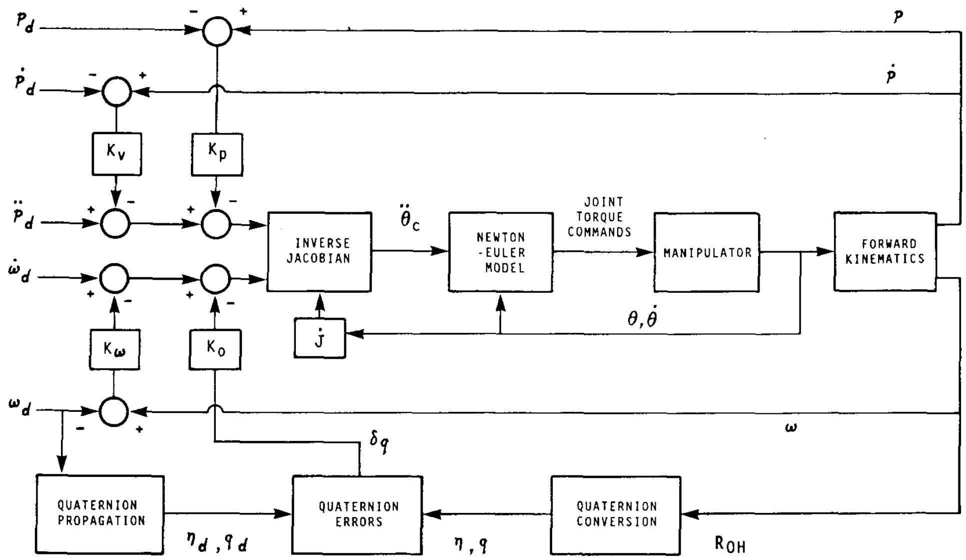


Fig. 3. Resolved acceleration control using quaternion feedback.

It can be shown by substitutions from (36) and the quaternion propagation equations (23) and (24) that the time derivative of $V$ along any quaternion trajectory $\{\eta, \boldsymbol{q}\}$ is

$$\dot{V} = -(\boldsymbol{\omega}-\boldsymbol{\omega}_d)^T \boldsymbol{K}_\omega(\boldsymbol{\omega}-\boldsymbol{\omega}_d) + (\boldsymbol{\omega}-\boldsymbol{\omega}_d)^T(\delta \boldsymbol{q} - \boldsymbol{K}_0 \boldsymbol{e}_0) \tag{38}$$

where the quaternion error vector $\delta \boldsymbol{q}$ is given by (27).

To make $\dot{V}$ negative-semidefinite, it is sufficient to set

$$\boldsymbol{K}_0 \boldsymbol{e}_0 = \delta \boldsymbol{q} \tag{39}$$

and $\boldsymbol{K}_\omega > 0$. In other words, let

$$\boldsymbol{e}_0 = \delta \boldsymbol{q} \qquad \boldsymbol{K}_0 = \boldsymbol{I} \tag{40}$$

where $\boldsymbol{I}$ is a unit matrix. We then have

$$\dot{V} = -(\boldsymbol{\omega}-\boldsymbol{\omega}_d)^T \boldsymbol{K}_\omega(\boldsymbol{\omega}-\boldsymbol{\omega}_d) \tag{41}$$

so that $\dot{V}(t) \leq 0$ for all $t$. In particular, whenever $\boldsymbol{\omega}$ differs from $\boldsymbol{\omega}_d$, $V$ will decrease in value until an equilibrium point is reached where $\dot{V} = 0$; i.e., $\boldsymbol{\omega} = \boldsymbol{\omega}_d$.

At $\boldsymbol{\omega} = \boldsymbol{\omega}_d$, we have from (36) and (39)

$$(d/dt)(\boldsymbol{\omega}-\boldsymbol{\omega}_d) = -\boldsymbol{K}_0 \boldsymbol{e}_0 = -\delta \boldsymbol{q}. \tag{42}$$

Suppose $\delta \boldsymbol{q} = 0$ at this equilibrium point. Then, by (42), $\boldsymbol{\omega}(t)$ will remain at $\boldsymbol{\omega}_d(t)$ for all $t$. By Proposition 1, the hand stays aligned with the desired orientation.

On the other hand, if $\boldsymbol{\omega} = \boldsymbol{\omega}_d$ but $\delta \boldsymbol{q}$ is nonzero, then $(\boldsymbol{\omega} - \boldsymbol{\omega}_d)$ will change according to (42) so that $\boldsymbol{\omega} = \boldsymbol{\omega}_d$ can only hold instantaneously. By (41), $\dot{V}$ becomes negative which causes $V$ to decrease further toward zero. Consequently, $\delta \boldsymbol{q}(t) \to 0$ as $t \to \infty$, which, by Proposition 1, corresponds to zero orientation error. Since this result holds for any initial value of $\delta \boldsymbol{q}$, we have global asymptotic convergence of the orientation error.

The quaternion feedback implementation of resolved acceleration control is illustrated in Fig. 3. As in the case of resolved rate control, the desired quaternion trajectory $\{\eta_d, \boldsymbol{q}_d\}$ can be either precomputed and recalled from memory or generated on-line using the propagation equation (23). The joint variables $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$ are assumed to be measurable and the hand velocities $\dot{\boldsymbol{p}}$ and $\boldsymbol{\omega}$ may be calculated from (18). Given $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, and $\ddot{\boldsymbol{\theta}}_C$ computed by (34) with (40) providing the orientation error feedback, the joint torques can then be calculated from a Newton–Euler dynamic model of the manipulator as in [21].

Let us now examine the stability of (36) when the orientation error feedback is given by a Euler rotation representation. A convergence analysis for this was given in [6] with the error expressed as in (4). But the results are only valid for short sampling intervals and small orientation errors. We shall show below that even large-angle stability can be established by using a quaternion formulation.

Substituting (30) into (39), we get

$$\boldsymbol{K}_0 \boldsymbol{e}_0 = 2\delta\eta \boldsymbol{K}_0 \delta \boldsymbol{q} = \delta \boldsymbol{q}. \tag{43}$$

Hence, provided the feedback gain matrix is set according to

$$\boldsymbol{K}_0 = (2\delta\eta)^{-1} \boldsymbol{I} \tag{44}$$

where $\boldsymbol{I}$ is a unit matrix, $\dot{V}$ is again given by (41). We, therefore, conclude that, despite what was claimed in [6], the orientation error converges *even* for large angles.

Note that, unlike the case in (40), the feedback gain $\boldsymbol{K}_0$ in (44) is nonlinear. This is due to the need to eliminate the second term in (38) so as to render $\dot{V}$ negative-semidefinite. The condition is only sufficient, not necessary. By keeping $\boldsymbol{K}_0$ constant as in (40), however, stability can no longer be guaranteed.

As a result of (44), the feedback gain has a singularity at $\delta\eta = 0$ which occurs when the hand orientation differs from its desired orientation by Euler rotation of 180°. This once again demonstrates the superiority of the direct quaternion feedback approach (40) over one using Euler rotations.

## V. Experimental Results

The resolved rate control law (20) using quaternion feedback was implemented on a Cincinnati Milacron $T^3$-776 industrial robot. The main objective of the experiment was to demonstrate large-angle stability of the two quaternion feedback formulations considered in this communication.

In the experiment the hand was commanded to follow a circular path of radius 200 mm in a vertical plane at a constant speed of 100 mm/s, while its initial orientation was displaced from the desired orientation by a rotation of 90° about the vertical (yaw) axis. Because this robot has very fast servo dynamics, only kinematic control was implemented; that is, the command calculated from (20) was applied directly to the inputs of the rate servos at the joints.

Figs. 4 and 5 describe the responses of the hand orientation with the feedback gain matrices set as follows:

$$\boldsymbol{K}_p = \boldsymbol{I} \qquad \boldsymbol{K}_0 = 0.2\boldsymbol{I}$$

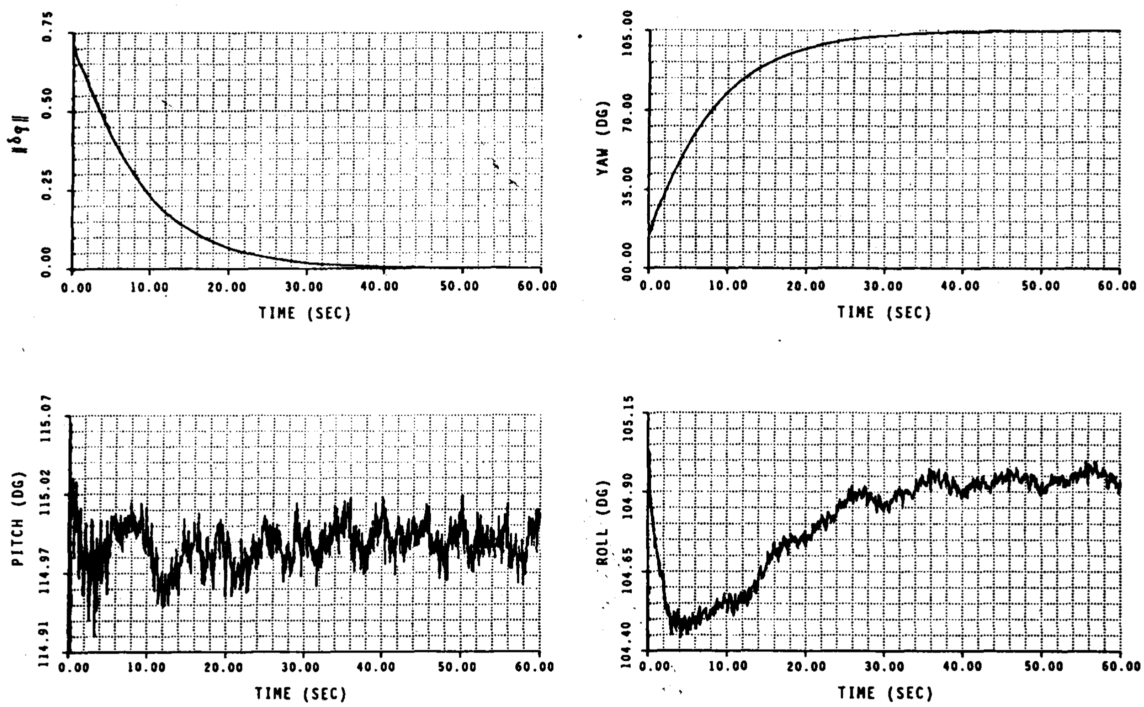


Fig. 4. Response with orientation error feedback given by $e_0 = \delta q$.

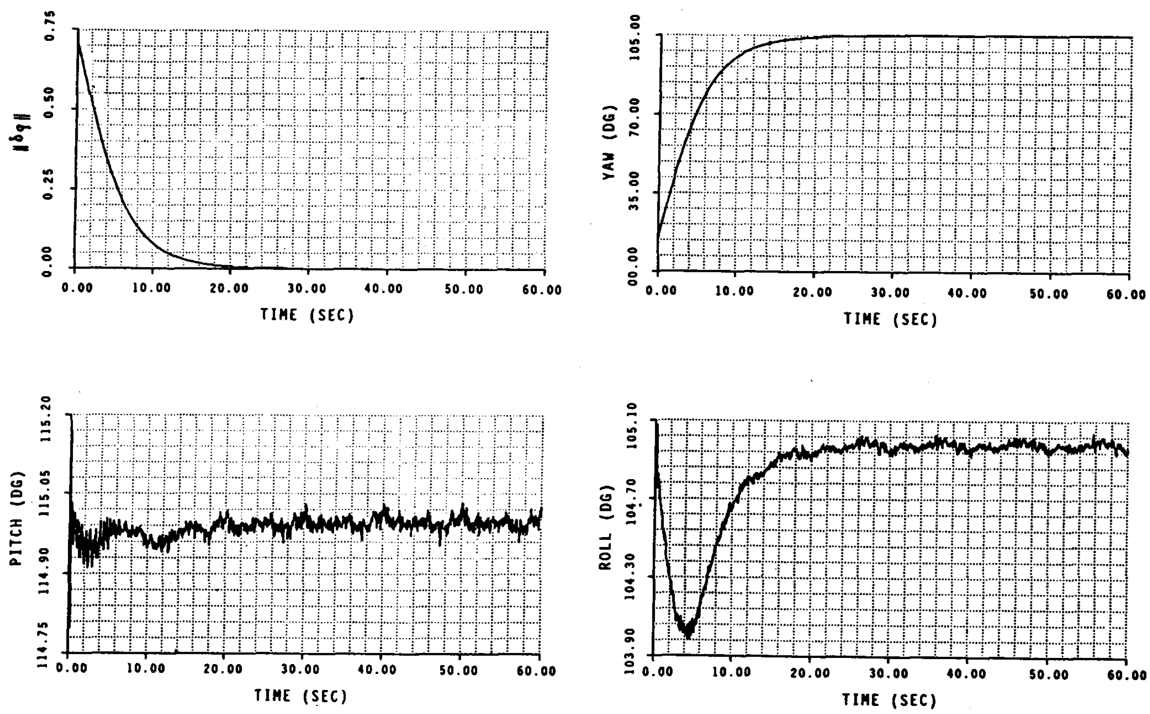


Fig. 5. Response with orientation error feedback given by $e_0 = 2\,\delta\eta\,\delta q$.

where $I$ is a $3 \times 3$ unit matrix. The orientation errors are presented in the figures both as $\|\delta q\|$, the Euclidean norm of $\delta q$, and as the conventional Euler angles (roll, pitch, and yaw).

Fig. 4 corresponds to the case with direct quaternion error feedback $e_0 = \delta q$, while in Fig. 5 the error feedback was based on Euler rotation $e_0 = 2\,\delta\eta\,\delta q$. The convergence rates for the two cases are different as expected due to the different $\dot{V}$ obtained in (29) and (32). In particular, the results of Fig. 5 demonstrate that the Euler rotation representation of (4) is valid even for large orientation errors.

## VI. Conclusions

Though quaternions are composed of a scalar and a vector, the orientation error is adequately represented by only the vector portion of the quaternion difference between the actual and the desired hand orientation. This vector quaternion error formulation greatly simplifies the stability analysis of the orientation error equations.

Of the two types of quaternion feedback considered, the approach using only $\delta q$ is far more superior to that derived from a Euler rotation representation ($2\,\delta\eta\,\delta q$) since the feedback gain is constant and stability is globally asymptotic. The Euler rotation feedback has a singularity when the actual hand orientation differs from its desired orientation by 180°. In resolved rate control, this singularity manifests itself as an equilibrium point with nonzero orientation error, while in resolved acceleration control, the feedback gain (which is nonlinear) becomes infinitely large.


## Acknowledgment

The author wishes to thank F. Keung for preparing the experimental results of this paper. He is also grateful for several insightful comments from the reviewers.


## References


[1] J. S. Schoenwald, M. S. Black, J. F. Martin, G. A. Arnold, and T. A. Allison, "Improved robot trajectory from acoustic range servo control," in *Proc. IEEE Int. Conf. on Robotics and Automation,* pp. 709-712, 1986.

[2] C. S. G. Lee, "Robot kinematics, dynamics and control," *IEEE Computer,* vol. 15, no. 12, pp. 62-80, Dec. 1982.

[3] R. P. Paul, *Robot Manipulators: Mathematics, Programming, and Control.* Cambridge, MA: MIT Press, 1981.

[4] J. Wittenburg, *Dynamics of Systems of Rigid Bodies.* Stuttgart, FRG: Teubner, 1977.

[5] D. E. Whitney, "The mathematics of coordinated control of prosthetic arms and manipulators," *Trans. ASME J. Dynamic Syst., Meas., Contr.,* pp. 303-309, Dec. 1972.

[6] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "Resolved-acceleration control of mechanical manipulators," *IEEE Trans. Automat. Contr.,* vol. AC-25, no. 3, pp. 468-474, June 1980.

[7] R. Ravindran and K. H. Doetsch, "Design aspects of the shuttle remote manipulator control," in *Proc. AIAA Guidance and Control Conf.* (San Diego, CA, 1982), paper 82-1581-CP.

[8] R. Mortensen, "A globally stable linear attitude regulator," *Int. J. Contr.,* vol. 8, no. 3, pp. 297-302, 1968.

[9] B. P. Ickes, "A new method for performing digital control system attitude computation using quaternions," *AIAA J.,* vol. 8, no. 1, pp. 13-17, Jan. 1970.

[10] B. Friedland, "Analysis strapdown navigation using quaternions," *IEEE Trans. Aerosp. Electron. Syst.,* vol. AES-14, no. 5, pp. 764-768, Sept. 1978.

[11] E. C. Wong and W. G. Breckenridge, "Inertial attitude determination for a dual-spin planetary spacecraft," *AIAA J. Guidance, Contr. Dyn.,* vol. 6, no. 6, pp. 491-498, Nov.-Dec. 1983.

[12] R. B. Miller, "A new strapdown attitude algorithm," *AIAA J. Guidenace, Contr. Dyn.,* vol. 6, no. 4, pp. 287-291, July-Aug. 1983.

[13] B. Wie and P. M. Barba, "Quaternion feedback for spacecraft large angle maneuvers," *AIAA J. Guidance, Contr. Dyn.,* vol. 8, no. 3, pp. 360-365, May-June 1985.

[14] P. E. Nikravesh, R. A. Wehage, and O. K. Kwon, "Euler parameters in comuptational kinematics and dynamics. Part 1," *ASME J. Mech., Transmiss., Automat. Des.,* vol. 107, pp. 358-365, Sept. 1985.

[15] P. E. Nikravesh, O. K. Kwon, and R. A. Wehage, "Euler parameters in computational kinematics and dynamics. Part 2," *ASME J. Mech., Transmiss., Automat. Des.,* vol. 107, pp. 366-369, Sept. 1985.

[16] B. Noble, *Applied Linear Algebra.* Englewood Cliffs, NJ: Prentice-Hall, 1969.

[17] S. W. Shepperd, "Quaternion from rotation matrix," *AIAA J. Guidance Contr.,* vol. 1, no. 3, pp. 223-224, May-June 1978.

[18] R. A. Mayo, "Relative quaternion state transition relation," *AIAA J. Guidance Contr.,* vol. 2, no. 1, pp. 44-48, Jan.-Feb. 1979.

[19] D. E. Whitney, "Resolved motion rate control of manipulators and human prostheses," *IEEE Tran. Man-Machine Syst.,* vol. MMS-10, no. 2, pp. 47-53, June 1969.

[20] C. H. Wu and R. P. Paul, "Resolved motion force control of robot manipulator," *IEEE Trans. Syst., Man Cybern.,* vol. SMC-12, no. 3, pp. 266-275, May/June 1982.

[21] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-line computational scheme for mechanical manipulators," *ASME J. Dynamic Syst., Meas. Contr.,* vol. 102, pp. 69-76, June 1980.

[22] W. Hahn, *Stability of Motion.* New York, NJ: Springer-Verlag, 1967.


# Kinematics of a Robot with Continuous Roll Wrist

KRISHNA C. GUPTA

*Abstract*—**Some operational details of the zero reference position method are presented in the context of deriving kinematical equations for a robot with a nonspherical continuous roll wrist.**

## I. Introduction

In a continuous roll wrist, all of the wrist joints have unrestricted rotational degree of freedom. The wrist axes are configured such that there is no mechanical interference among the links of the wrist as the wrist variables change continuously in the ranges [0, 360°]. There are several ways to induce the continuous roll feature in wrists and we discuss two important ways. Let the wrist axes be labeled $n-2$, $n-1$, and $n$ (last).

In a spherical wrist (i.e., $s_{n-1,n} = a_{n-2,n-1} = a_{n-1,n} = 0$) with serially orthogonal joint axes (i.e., $\alpha_{n-2,n-1} = \alpha_{n-1,n} = 90°$), there is mechanical interference when we turn about the $(n-1)$th joint. If the angles between successive joint axes are changed such that $\alpha_{n-2,n-1} = 90° + \beta$ and $\alpha_{n-1,n} = 90° - \beta$, then such a nonorthogonal spherical wrist has a continuous 3-roll property. Kinematic solutions of robots with spherical wrists are straightforward because of decoupling [1]-[3], [5]-[7]; among these, [1] and [7] also discuss the solutions of a variety of other robot-arm configurations.

Another way to modify the orthogonal spherical wrist is to introduce a small amount of offset ($s_{n-1,n} \neq 0$). We then have a nonspherical wrist in which the axes $(n-2)$ and $(n-1)$ intersect at one point (i.e., $a_{n-2,n-1} = 0$), but the axes $(n-1)$ and $n$ intersect at another point (i.e., $a_{n-1,n} = 0$), thus causing a small offset $s_{n-1,n} \neq 0$. Angles between the successive axes are $\alpha_{n-2,n-1} = \alpha_{n-1,n} = 90°$.

In this communication, we use the zero reference position method [2] for analyzing a robot with the second type of continuous roll wrist (nonspherical). The zero reference position method is a simple method for formulating robot kinematics just from the data on joint axes directions and locations in a conveniently chosen reference

Manuscript received April 24, 1987; revised August 28, 1987. This work was supported by the U. S. Army Research Office under Contracts 20125 EG and 24647 EG. Part of the material in this communication was presented at the 1987 International Conference on Robotics and Automation, Raleigh, NC, March 31-April 3, 1987.

The author is with the Department of Mechanical Engineering, University of Illinois at Chicago, Chicago, IL 60680.

IEEE Log Number 8820164.

0882-4967/88/0800-0440$01.00 