MODELING, IDENTIFICATION AND CONTROL, 1987, VOL. 8, NO. 3, 175–184
doi:10.4173/mic.1987.3.5

Coordinating control of a special joint structure with more servos than degrees of freedom

FREDRIK DESSEN†

Keywords: *robot manipulators, redundancy, coordinate transformation, force control, wire drive mechanism.*

A joint mechanism for use as a general building block in manipulators with a very high number of degrees of freedom is introduced. It consists of 3 servos driving a 2 d.o.f. universal joint by means of wire. A coordinate transformation set is developed, which includes positional, velocity and force transformations. Both direct and inverse transformations are presented. Special attention is given to the inverse force transformation which is obtained using linear optimization. The solution in this case is also shown to be valid for a more general class of constrained non-linear optimization problems. An example is given of the use of the coordinate transformation set; a joint control system including servos under internal force control.

1. Introduction

The coordinate transformations necessary to describe movements of a complete robotic or remote controlled manipulator can be separated into at least two levels. The higher level concerns transformations between the generalized coordinates of the joints and some external coordinate system which is convenient for task description (Whitney 1969; Paul 1981). At this level the corresponding generalized forces are often considered as input to the system's dynamic description (Luh, Walker and Paul 1980a, b; Hollerbach 1980; Bejczy, Tarn and Chen 1985). The lower level describes relations between the joints and their driving servos. These transformations seem to have been given less attention in the literature, perhaps because structural designs at this level vary more than the designs at the higher level. This article develops the lower level description for one specific servo to joint connection design.

The design arises from an idea developed at the Norwegian Institute of Technology concerning a manipulator with a high degree of kinematic redundancy. With this type of manipulator it is necessary to reduce the weight of each joint as much as possible. This is done by mounting all servos in the manipulator base and by using a light weight two degrees of freedom joint as a standard building block. The arm consists of a number of straight links connected by universal joints. Each joint is driven by linear hydraulic servos. The joints and servos are connected by steel wire as shown in Fig. 1. Since the wire can support pulling forces only, each 2 d.o.f. joint requires a minimum of three pulling wires. This paper will describe such a joint in mathematical terms. In the two following sections the forward and inverse positional transformations are developed. Then, differential transformations are treated.

Received 12 March 1987.

† Norwegian Institute of Technology, Division of Engineering Cybernetics, N-7034 Trondheim-NTH, Norway.

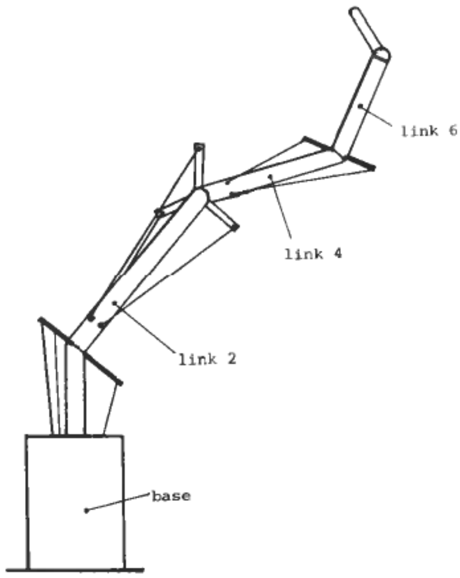

Figure 1. Wire driven manipulator. The 9 servos driving 6 d.o.f. for positioning are mounted in the base. When including a servo rotating base and a 3 d.o.f. spherical wrist joint, the total number of d.o.f. is 10.

Special attention is given to the problem of inverting the relationship between wire forces and joint torques. This is solved as a linear programming problem.

2. Description of the mechanism

A 2 d.o.f. mechanism joining links j and $j + 2$ of a manipulator is shown in Fig. 2. The wires driving link $j + 2$ are connected to moment arms mounted on the same

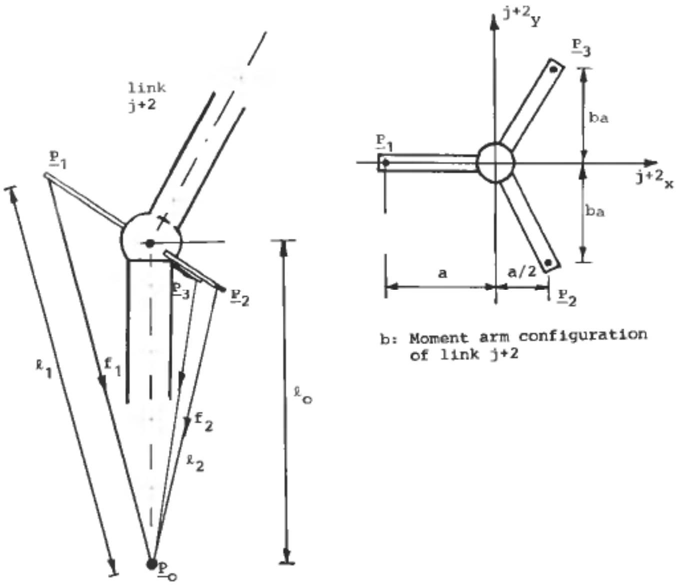

Figure 2. Wire drive mechanism.

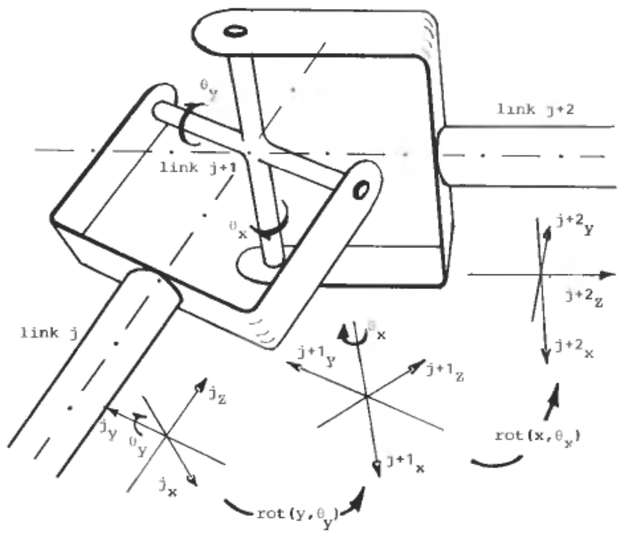

Figure 3. Rotations of a universal joint. All coordinate systems have origin in the pivot intersection point of link $j + 1$. Any coordinate system i is fixed with respect to link i.

link (Fig. 2(*b*)). The connecting point positions are denoted by $\boldsymbol{p}_1$, $\boldsymbol{p}_2$, $\boldsymbol{p}_3$. From these points the wires are led through point $\boldsymbol{p}_0$ on link j and down to the manipulator base. Note that in practice it is impossible to lead three wires through the same point. However, as this approximation simplifies the development of the analytic expressions, it will be used throughout this paper. Figure 2(*b*) shows the mounting of moment arms on link $j + 2$. The points $\boldsymbol{p}_i$ lie in the plane ${}^{j+2}x = 0$. Their positions in the plane are given by the constant a and b. Usually one would choose $b = 1/2\sqrt{3}$ so that all three moment arms have length a, and the angle between each pair is 120°.

The kinematic structure of the link is defined in Fig. 3. The total rotational transformation of a universal joint can be represented by a rotation θ_y about a y-axis fixed in link j coordinates and a rotation θ_x about the x-axis resulting from the first rotation. The rotational matrix from j to link $j + 2$ becomes:

$$\boldsymbol{R} = \text{rot}\,(y, \theta_y)\,\text{rot}\,(x, \theta_x) = \begin{bmatrix} c_y & s_y s_x & s_y c_x \\ 0 & c_x & -s_x \\ -s_y & c_y s_x & c_y c_x \end{bmatrix} \tag{1}$$

where $s_y = \sin\theta_y$, $c_y = \cos\theta_y$ etc.

The values θ_y and θ_x are defined as the generalized coordinates describing the movement of the 2 d.o.f. joint.

3. Positional transformations

Displacements of the three servos driving the joint can be defined by lengths $[l_1, l_2, l_3]^{\mathrm{T}}$ of the wires in Fig. 2. Knowing the positions, $\boldsymbol{p}_i$, of each wire end point in link j coordinates and using point $\boldsymbol{p}_0$ as reference, l_i will be found as the vector magnitude of $\boldsymbol{p}_i$. The wire end points, as functions of joint variables are:

$$\boldsymbol{p}_1 = \begin{bmatrix} 0 \\ 0 \\ l_0 \end{bmatrix} + \boldsymbol{R}\begin{bmatrix} -a \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -ac_y \\ 0 \\ l_0 + as_y \end{bmatrix} \tag{2}$$

$$\boldsymbol{p}_2 = \begin{bmatrix} 0 \\ 0 \\ l_0 \end{bmatrix} + \boldsymbol{R}\begin{bmatrix} a/2 \\ -ba \\ 0 \end{bmatrix} = \begin{bmatrix} ac_y/2 - bas_y s_x \\ -bac_x \\ l_0 - as_y/2 - bac_y s_x \end{bmatrix} \tag{3}$$

$$\boldsymbol{p}_3 = \begin{bmatrix} 0 \\ 0 \\ l_0 \end{bmatrix} + \boldsymbol{R}\begin{bmatrix} a/2 \\ ba \\ 0 \end{bmatrix} = \begin{bmatrix} ac_y/2 + bas_y s_x \\ bac_x \\ l_0 - as_y/2 + bac_y s_x \end{bmatrix} \tag{4}$$

so the vector magnitudes become:

$$\begin{aligned} l_1 &= \sqrt{(a^2 + l_0^2 + 2l_0 as_y)} \\ l_2 &= \sqrt{[(b^2 + \tfrac{1}{4})a^2 + l_0^2 - l_0 as_y - 2l_0 bac_y s_x]} \\ l_3 &= \sqrt{[(b^2 + \tfrac{1}{4})a^2 + l_0^2 - l_0 as_y + 2l_0 bac_y s_x]} \end{aligned} \tag{5}$$

Eqn. (5) expresses the servo positions necessary to give a certain joint displacement value $\boldsymbol{\theta} = [\theta_y, \theta_x]^{\mathrm{T}}$. In some cases the inverse expression might be useful, if for instance one has decided to measure the wire lengths $\boldsymbol{l}$ to calculate $\boldsymbol{\theta}$. The following two linear combinations are obtained after squaring (5):

$$\begin{aligned} 6l_0 as_y &= 2(b^2 - \tfrac{3}{4})a^2 + 2l_1^2 - l_2^2 - l_3^2 \\ 4l_0 bac_y s_x &= -l_2^2 + l_3^2 \end{aligned} \tag{6}$$

Though many other linear combinations of (5) squared can be used, (6) exploits $\boldsymbol{l}$ reasonably well, and is therefore recommended. $\boldsymbol{\theta}$ is now found by:

$$\sin\theta_y = \frac{(b^2 - 3/4)a}{3l_0} + \frac{1}{6l_0 a}(2l_1^2 - l_2^2 - l_3^2) \tag{7}$$

$$\sin\theta_x = (-l_2^2 + l_3^2)/[4l_0 ba\sqrt{(1 - \sin^2\theta_y)}] \tag{8}$$

When θ_y or θ_x approaches $\pi/2$ radians, numerical problems may arise. However, since the joint design (Figs. 2 and 3) sets a practical limit of approx. $\pi/4$ on θ_y and θ_x, this situation will never occur. Note that if $b = 1/2\sqrt{3}$ then Eqns. (5)–(8) become somewhat simpler.

4. Direct differential transformations

Given an infinitesimal change in the joint variable one can find the corresponding change in servo positions:

$$d\boldsymbol{l} = \boldsymbol{J}d\boldsymbol{\theta} \tag{9}$$

where $\boldsymbol{J}$, the Jacobian, is given by

$$\boldsymbol{J} = \begin{bmatrix} \dfrac{\partial l_1}{\partial \theta_y} & \dfrac{\partial l_2}{\partial \theta_y} & \dfrac{\partial l_3}{\partial \theta_y} \\ \dfrac{\partial l_1}{\partial \theta_x} & \partial l_2 \dfrac{\partial l_2}{\partial \theta_x} & \dfrac{\partial l_3}{\partial \theta_x} \end{bmatrix}^{\mathrm{T}}$$

The elements of this matrix are found by differentiating (5):

$$\begin{aligned} &\frac{\partial l_1}{\partial \theta_y} = \frac{l_0 a}{l_1} c_y; \quad \frac{\partial l_1}{\partial \theta_x} = 0 \\ &\frac{\partial l_2}{\partial \theta_y} = \frac{l_0 a}{l_2}(b s_y s_x - \tfrac{1}{2} c_y); \quad \frac{\partial l_2}{\partial \theta_x} = \frac{l_0 a}{l_2}(-b c_y c_x) \\ &\frac{\partial l_3}{\partial \theta_y} = \frac{l_0 a}{l_3}(-b s_y s_x - \tfrac{1}{2} c_y); \quad \frac{\partial l_3}{\partial \theta_x} = \frac{l_0 a}{l_3}(b c_y c_x) \end{aligned} \tag{10}$$

The Jacobian can be used for velocity transformations as described in (Whitney 1969):

$$\dot{\boldsymbol{l}} = \boldsymbol{J}\dot{\boldsymbol{\theta}} \tag{11}$$

where the dots denote time derivatives.

In addition one can find joint torques from wire forces using the virtual work method as in (Paul 1981):

$$\boldsymbol{m} = -\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f} \tag{12}$$

where $\boldsymbol{m}$: joint torques; $\boldsymbol{f}$: wire forces. The negative sign is a result of the definition of force direction against servo displacement shown in Fig. 2.

Note that the Jacobian, as defined in (10), can be conveniently written:

$$\boldsymbol{J} = a\boldsymbol{L}^{-1}\boldsymbol{H} \tag{13}$$

where a: moment arm defined in Fig. 2(*b*), and

$$\boldsymbol{L} = \frac{1}{l_0} \operatorname{diag} \boldsymbol{l}\text{: diagonal matrix of normalized wire lengths}$$

$$\boldsymbol{H} = \begin{bmatrix} c_y & b s_y s_x - \tfrac{1}{2} c_y & -b s_y s_x - \tfrac{1}{2} c_y \\ 0 & -b c_y c_x & b c_y c_x \end{bmatrix}^{\mathrm{T}}$$

The effect of collecting the wires in $\boldsymbol{p}_0$ is thus represented by the diagonal matrix $\boldsymbol{L}$. Also note that, according to Fig. 2, if $l_0 \gg a$ then $\boldsymbol{L} \approx \boldsymbol{I}$, the identity matrix. These facts will be used to simplify the development of an inverse force transformation.

$\boldsymbol{J}$ is a 2 by 3 matrix. This creates a problem when trying to solve (11) or (12). A glance at the equations will show that, in general, (11) has no solution and (12) has an infinite number of solutions. The following two sections describe how to deal with these two problems.

5. Inverse velocity transformation

The inverse of (11) will be of use when measuring $\dot{\boldsymbol{l}}$ in order to calculate $\dot{\boldsymbol{\theta}}$. In this case we have a redundant set of measurements which could be reduced simply by neglecting one of the elements in $\dot{\boldsymbol{l}}$. However, one would obtain a better estimate of $\dot{\boldsymbol{\theta}}$ by exploiting all three measurements. A reasonably good estimate of $\dot{\boldsymbol{\theta}}$ is given by

$$\dot{\boldsymbol{\theta}} = \boldsymbol{J}^{+}\dot{\boldsymbol{l}} \qquad (14)$$

where $\boldsymbol{J}^{+} = (\boldsymbol{J}^{\mathrm{T}}\boldsymbol{J})^{-1}\boldsymbol{J}^{\mathrm{T}}$: a pseudoinverse of $\boldsymbol{J}$.

$\boldsymbol{J}^{+}$ can be expressed in detail using (10) or (13). It can be shown that (14) gives a minimum variance estimate of $\dot{\boldsymbol{\theta}}$ if the measuring errors for each element in $\dot{\boldsymbol{l}}$ are independent and equal in variance.

6. Inverse force transformation

The main problem when inverting (12) is how to choose one $\boldsymbol{f}$ among those who satisfy the equation. A simple solution involves constructing a 3×3 matrix by augmenting $-\boldsymbol{J}^{\mathrm{T}}$ with an artificial constraint. This constraint may specify a bias force keeping a weighted sum of the wire forces constant. A reasonable servo force vector $\boldsymbol{f}$ may also be found using some optimization procedure.

The method used in this section minimizes the force vector components. This tends to keep joint wear and dissipated power low without explicitly minimizing these effects. Since wires support pulling forces only, the solution must only contain positive force values. See Fig. 2. In fact all wire forces should be greater than a predefined positive value. This is both because the hydraulic servos often have bad characteristics near zero force and because the wire might behave badly when slack. In addition to these constraints, (12) must be satisfied. If the function to be minimized is linear, the optimization problem can be solved using the simplex method (Phillips *et al.* 1976). As argued at the end of this section, the exact shape of this function is not important. For this reason a linear function will be used. Furthermore the function to be used will be especially designed to obtain the solution in a simple way.

Before stating the optimization problem, matrix $\boldsymbol{J}$ is separated as shown in (13). A minimum force vector is chosen as

$$\boldsymbol{f}_{\mathrm{min}} = \boldsymbol{L}\boldsymbol{f}_{\mathrm{m}} \qquad (15)$$

where $\boldsymbol{f}_{\mathrm{m}} = [f_{\mathrm{m}}, f_{\mathrm{m}}, f_{\mathrm{m}}]^{\mathrm{T}}$: predefined constant vector.
Defining the variable $\boldsymbol{k}$,

$$\boldsymbol{k} = a\boldsymbol{L}^{-1}\boldsymbol{f}, \qquad (16)$$

an optimization problem, which is independent of a and $\boldsymbol{L}$, can be stated:

$$\begin{aligned} &\text{minimize} && B = k_1 + k_2 + k_3 \quad \text{(A)} \\ &\text{subject to} && -\boldsymbol{H}^{\mathrm{T}}\boldsymbol{k} = \boldsymbol{m} \quad \text{(B)} \\ &&& \boldsymbol{k} \geqslant \boldsymbol{k}_{\mathrm{m}} \quad \text{(C)} \end{aligned} \qquad (17)$$

where $\boldsymbol{H}$: defined in (13); $\boldsymbol{m}$: desired joint torque vector

$$\boldsymbol{k}_{\mathrm{m}} = a\boldsymbol{L}^{-1}\boldsymbol{f}_{\mathrm{min}} = a\boldsymbol{f}_{\mathrm{m}}$$

The problem can be solved numerically using the simplex method for linear programming problems.

Before using the simplex method, (17(C)) must be formulated as an equality constraint by adding three new variables, $\boldsymbol{k}' = [k'_1, k'_2, k'_3]^{\mathrm{T}}$, where $\boldsymbol{k}' \geqslant \boldsymbol{0}$. Then

$$\boldsymbol{k} + \boldsymbol{k}' = \boldsymbol{k}_{\mathrm{m}} \qquad (18)$$

The problem now has 5 linear equality constraints and 6 variables. According to the theory of linear programming, any optimal solution contains at least one variable equal to zero. Since $\boldsymbol{k} \geqslant \boldsymbol{k}_\mathrm{m}$ (definitely positive) one of the elements in $\boldsymbol{k}'$ has to be zero. According to (18) the corresponding element in $\boldsymbol{k}$ equals k_m. As a result the solution can be divided into 3 cases.

case i: $\quad k_i = k_\mathrm{m} \quad$ and $\quad k_i' = 0$

The solution in each case can be represented symbolically. Before developing the three cases, three matrices $\boldsymbol{H}_i$ will be defined: $\boldsymbol{H}_i$ is the 2×2 matrix obtained by omitting row no. i of $\boldsymbol{H}$. $\boldsymbol{H}$ is defined in (13). E.g.:

$$\boldsymbol{H}_1 = \begin{bmatrix} bs_y s_x - \tfrac{1}{2}c_y & -bs_y s_x - \tfrac{1}{2}c_y \\ -bc_y c_x & bc_y c_x \end{bmatrix}^\mathrm{T} \tag{19}$$

CASE 1:

Inserting $k_1 = k_\mathrm{m}$ in (17(B)), the restriction becomes:

$$-\begin{bmatrix} k_\mathrm{m} c_y \\ 0 \end{bmatrix} - \boldsymbol{H}_1^\mathrm{T} \begin{bmatrix} k_2 \\ k_3 \end{bmatrix} = \begin{bmatrix} m_y \\ m_x \end{bmatrix}$$

$$\begin{bmatrix} k_2 \\ k_3 \end{bmatrix} = -\boldsymbol{H}_1^{-\mathrm{T}} \begin{bmatrix} m_y + k_\mathrm{m} c_y \\ m_x \end{bmatrix}$$

where by (19)

$$\boldsymbol{H}_1^{-\mathrm{T}} = \begin{bmatrix} -\dfrac{1}{c_y} & -\dfrac{1}{2bc_y c_x} - \dfrac{s_y s_x}{c_y^2 c_x} \\ -\dfrac{1}{c_y} & \dfrac{1}{2bc_y c_x} - \dfrac{s_y s_x}{c_y^2 c_x} \end{bmatrix}$$

so

$$\begin{aligned} k_1 &= k_\mathrm{m} \\ k_2 &= k_\mathrm{m} + \frac{1}{c_y} m_y + \left(\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \\ k_3 &= k_\mathrm{m} + \frac{1}{c_y} m_y + \left(-\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \end{aligned} \tag{20}$$

For case 1 to be valid, the conditions

$$k_2 \geqslant k_\mathrm{m} \quad \text{and} \quad k_3 \geqslant k_\mathrm{m}$$

must be satisfied. Using (20) this can be written

$$\begin{aligned} &\frac{m_y}{c_y} + \left(\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \geqslant 0 \\ &\frac{m_y}{c_y} + \left(-\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \geqslant 0 \end{aligned} \tag{21}$$

CASE 2:

Here the solution is:

$$\begin{aligned} k_1 &= k_m - \frac{1}{c_y} m_y - \left(\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \\ k_2 &= k_m \\ k_3 &= k_m - \frac{1}{bc_y c_x} m_x \end{aligned} \tag{22}$$

And conditions for case 2 to be valid:

$$-\frac{m_y}{c_y} - \left(\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \geqslant 0$$

$$-\frac{1}{bc_y c_x} m_x \geqslant 0$$

CASE 3:

Here the solution is:

$$\begin{aligned} k_1 &= k_m - \frac{1}{c_y} m_y - \left(\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \\ k_2 &= k_m + \frac{1}{bc_y c_x} m_x \\ k_3 &= k_m \end{aligned} \tag{24}$$

And the conditions for case 3 to be valid:

$$-\frac{m_y}{c_y} - \left(-\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) m_x \geqslant 0$$

$$\frac{1}{bc_y c_x} m_x \geqslant 0 \tag{25}$$

Using (21), (23) and (25) to design a decision rule, an algorithm to solve (17) can be stated:

BEGIN

case 1 ← true; case 2 ← true; case 3 ← true

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \leftarrow \begin{bmatrix} 0, & \frac{1}{bc_y c_x} \\ \frac{1}{c_y}, & \left(-\frac{1}{2bc_y c_x} + \frac{s_y s_x}{c_y^2 c_x}\right) \\ \frac{-1}{c_y}, & \left(-\frac{1}{2bc_y c_x} - \frac{s_y s_x}{c_y^2 c_x}\right) \end{bmatrix} \begin{bmatrix} m_y \\ m_x \end{bmatrix}$$

IF $d_1 > 0$ THEN case 2 ← false
ELSE case 3 ← false

IF $d_2 > 0$ THEN case 3 ← false
ELSE case 1 ← false

IF $d_3 \geqslant 0$ THEN case 1 $\leftarrow$ false
ELSE case 2 $\leftarrow$ false

IF case 1 THEN $\boldsymbol{k} \leftarrow \boldsymbol{k}_m + [0, -d_3, d_2]^T$

IF case 2 THEN $\boldsymbol{k} \leftarrow \boldsymbol{k}_m + [d_3, 0, -d_1]^T$

IF case 3 THEN $\boldsymbol{k} \leftarrow \boldsymbol{k}_m + [-d_2, d_1, 0]^T$

END (26)

After finding the optimal vector $\boldsymbol{k}$ the optimal vector $\boldsymbol{f}$ is computed using (16):

$$\boldsymbol{f} \leftarrow \frac{1}{a} \boldsymbol{L}\boldsymbol{k} \tag{26B}$$

When developing the optimal solution, the criterion $B = k_1 + k_2 + k_3$ was never mentioned explicitly. B was implicitly considered as an increasing linear function of $\boldsymbol{k}$, so the solution also applies to the more general objective function:

$$B = a_1 k_1 + a_2 k_2 + a_3 k_3,\ a_i > 0$$

or

$$B = a_1 f_1 + a_2 f_2 + a_3 f_3,\ a_i > 0$$

In fact the solution applies to any objective function which is monotonically increasing in $\boldsymbol{k}$ or $\boldsymbol{f}$, so quite generally it can be said to minimize 'the wire forces' subject to the constraints given in (17).

7. Application example

A system involving the developed expressions is shown in Fig. 4. Here Eqn. (12), (11) and (5) are used to describe the transformations in the mechanical part of the system. These are well defined direct expressions. The inverse expressions are calcu-

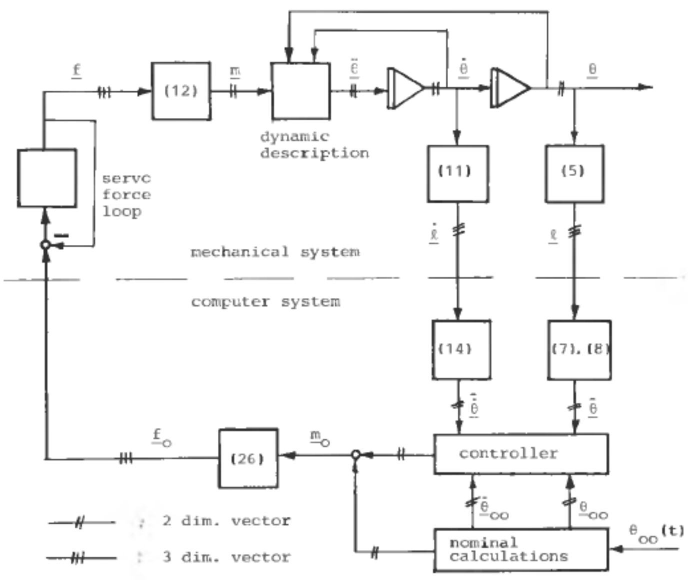

Figure 4. Universal joint control system.

lated in a real time computer, supporting a positional controller working in joint coordinates. In Fig. 4 the state vector, $[\boldsymbol{\theta}^T, \dot{\boldsymbol{\theta}}^T]^T$, is estimated using (14), (17) and (18) on a redundant set of measurements on the wires. The controller output is converted into servo force reference using the algorithm (26).

8. Conclusion

This paper has pointed out the use of lower level coordinate conversion for one special servo mechanism which illustrates problems concerning both non-linear transmissions and redundancy.

Note that the velocity conversion expression (11) was inverted using a traditional pseudoinverse while the force/torque relation (12) was solved as a linear programmimg problem. The difference in solution methods reflects differences in nature of the two problems at hand. The inversion method for the force expression is especially interesting since it solves a quite general class of optimization problems at the same time.

Obviously such a set of transformations must be supported by a powerful computer. For most applications however, it is recommended that simpler, but less accurate, transformation expressions are used. In this case the given expressions will only serve as a guide. It is of course also possible to avoid the problem by using kinematic structures that are simpler to describe mathematically but perhaps more expensive to manufacture. This approach will become less attractive in the future since the price of computing power is decreasing in relation to the price of mechanical structures.

References

Bejczy, A. K., Tarn, T. J. and Chen, Y. L. (1985). Robot Arm Dynamic Control by Computer. *1985 IEEE International Conference on Robotics and Automation, St. Louis*, March 25–28, 960–970.

Hollerbach, J. M. (1980). A recursive Lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity. *IEEE Trans. Syst., Man & Cybernetics*, **10**, 730–736.

Luh, J. Y. S., Walker, M. W. and Paul, R. P. C. (1980a). Resolved acceleration control of mechanical manipulators. *IEEE Trans. Autom. Control*, **25**, 468–474.

Luh, J. Y. S., Walker, M. W. and Paul, R. P. C. (1980b). On-line computational scheme for mechanical manipulators. *J. Dynamic Systems, Measurement and Control*, **102**, 69–76.

Paul, R. P. (1981). *Robot Manipulators.* (MIT Press: Cambridge, MA).

Phillips, Don T., Ravindran, A. and Solberg, J. (1976). *Operations Research: Principles and Practice.* (John Wiley & Sons: New York).

Whitney, D. E. (1969). Resolved motion rate control of manipulators and human prostheses. *IEEE Trans. Man-Machine Syst.* **10**, 47–53.