DYNAMIC CONTROL OF ROBOTS WITH JOINT ELASTICITY

Alessandro De Luca

Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Eudossiana 18 , 00184 Roma, Italy

Abstract. The problem of controlling the dynamic behavior of robots with rigid links but in presence of joint elasticity is much more complex than the one in which the manipulator is assumed to be rigid in both the links and the transmission elements. When the full nonlinear and interacting dynamics is taken into account in the model, it is known that the resulting system may not be feedback linearizable using nonlinear *static* state-feedback. It is shown here that use of the more general class of *dynamic* nonlinear state-feedback allows to solve both the feedback linearization and the input-output decoupling problems. A constructive procedure for the decoupling and linearizing feedback is given which is based on generalized system inversion and on the properties of the so-called zero-dynamics of the system. A case study of a planar two-link robot with elastic joints is included. The role of dynamic feedback for this class of robots is discussed.

1. Introduction

In the process of modeling the dynamic behavior of robot manipulators, some physical phenomena like backlash, stiction, link flexibility and joint elasticity are usually neglected. In general, when a reliable model of such disturbances is available, one can reduce the control effort and achieve better performances. Various experimental and simulation studies [1,2,3] have shown that elasticity of the transmission elements between actuators and links may have a relevant influence on robot dynamics. In particular, robots using transmission belts, long shafts or harmonic drives show a typical resonant behavior in the same range of frequencies used for control; this effect can be reconducted to the presence of elasticity at the joints between the rigid links of the arm.

From the modeling point of view, joint elasticity implies that the position of the actuator (i.e. the angle of the motor shaft) is not uniquely related to the position of the driven link. This *internal* deflection is taken into account by inserting a linear torsional spring at each joint. As a consequence, the rigid arm dynamic model has to be modified in order to describe completely the relation between applied torques and links motion. For quasi-static applications, simplified models which consider only the dynamics of the drive system have been used widely, see e.g. [4]. Models including the full nonlinear dynamic interactions among joint elasticities and inertial properties of links and actuators have been first introduced in [5]. There exist programs that generate automatically the dynamic equations for these arms, using symbolic manipulation languages [6].

Control of robot arms with joint elasticity has recently become an active area of research. Several advanced control approaches have been proposed using singular perturbation techniques [7], integral manifold design [8,9], sliding mode [10], pseudo-linearization [11], and model reference adaptive control [12]. It is worth to point out that none of these methods achieves an *exact* design, in the sense that is able to mimic the results obtained in the nominal case for fully rigid manipulators.

The reason for such an outgrow of methods stands in the peculiar control theoretic feature that distinguishes robots with joint elasticity from rigid manipulators: in fact, this class of nonlinear systems does not satisfy the necessary conditions for obtaining linearization and noninteraction using nonlinear static state-feedback [13,14]. *In general*, the so-called inverse dynamics or computed torque method, which is the standard nominal trajectory control for rigid arms, cannot be extended directly. There are however simple types of robots with elastic joints, like the single-link [15] and the two-link cylindric arm [16], where use of nonlinear static feedback still leads to a linear system; in these cases, the closed-loop equivalent behavior is that of, respectively, one or two independent chains of *four* integrators each.

A detailed analysis of several kinematic types of arms with elastic joints is contained in [17,18]. The resulting picture is quite intriguing. Feedback linearization may or may not be achieved, depending on the specific kinematic arrangement (i.e. the set of Denavit-Hartenberg parameters) of the robot. In particular, it is never possible for structures in which two or more elastic joints have parallel axes of rotation, like for the common case of a two-link planar arm. As an effort to achieve feedback linearization properties, a simplified model which neglects the inertial couplings between actuators and links has been investigated in [19].

Only recently a unifying perspective has been found for extending exact methods of nonlinear control to the full model of robots with joint elasticity. The keypoint is to enlarge the class of allowed feedback rules to the use of *dynamic* nonlinear state-feedback. Exploiting this concept, input-output decoupling and exact state linearization have been obtained in the closed-loop control of two arms with joint elasticity, a two-link arm with gravity [20] and a three-link anthropomorphic one [21], both of which cannot be linearized using only static feedback.

These results can be applied to the whole class of robots with elasticity at the joints [17] and were obtained by extensive use of nonlinear differential-geometric concepts such as the properties of the maximal controlled invariant distribution [22] associated to the system. The complexity of the techniques and of the computations involved has limited to a certain extent the understanding of the generality of such a design.

In this paper, the introduction of dynamic compensation in the control scheme follows from a more natural approach. The control design is based on the capability of properly recovering the inputs to the system from the knowledge of the outputs and of their time derivatives, an issue which is related to system inversion. In particular, a full linearizing controller can be obtained if the system has *no zero-dynamics* [23], that is if there is no dynamics left in the system once the output is forced to be zero. This concept is a quite useful extension to nonlinear systems of the notion of transfer function zeros for linear systems. The *generalized inverse dynamics* approach that is presented here incorporates the use of either static or dynamic state-feedback. In the multi-input multi-output case, the need of the latter arises only if the so-called decoupling matrix [22] of the system is singular.

Dynamic modeling of robot arms is briefly reviewed next. Section 3 outlines how to construct a full linearizing controller for nonlinear systems without zero-dynamics [24,25], a result which is used in Section 4 for a planar two-link robot with joint elasticity. Discussion of the obtained results and comparison with other approaches completes the paper.

2. Dynamic model of robots with joint elasticity

As a whole, a robot arm with N elastic joints can be seen as a set of 2N elastically coupled rigid bodies, the N actuators and the N links. In face of 2N mechanical degrees of freedom, only N independent control inputs are available, the motor torques acting on the actuator side of the elastic joints.

The dynamic equations of motion are obtained following a Lagrangian approach. Two variables are associated to the generic i-th elastic joint (see Figure 1): q_{2i-1}, the rigid position of the i-th actuator with respect to the (i-1)-th link, and q_{2i}, the elastic position of the i-th link with respect to the previous one. The 2N-dimensional vector $\mathbf{q}$ of generalized coordinates is partitioned into two N-vectors $\mathbf{q}_r$ and $\mathbf{q}_e$ containing respectively the odd and the even components.

The potential energy $U(\mathbf{q})$ and the kinetic energy $T(\mathbf{q},\mathbf{q}')$ are computed in the usual way, considering the arm as an open chain of 2N rigid bodies, links and actuators (see the example in Appendix). Beside the gravitational contribution $U_g(\mathbf{q})$, $U(\mathbf{q})$ contains also the elastic energy stored in the joints:

$$U_e(\mathbf{q}) = \sum_{i=1}^{N} U_{e,i}(q_{2i-1}, q_{2i}) = \sum_{i=1}^{N} \frac{1}{2} K_i \left[q_{2i} - \frac{q_{2i-1}}{N_i} \right]^2$$

K_i is the elastic constant and $N_i \geq 1$ is the transmission ratio of the i-th drive. No damping is given to the springs modeling joint elasticity. The Euler-Lagrange equations particularize as:

$$\frac{d}{dt}\left[\frac{\partial T}{\partial \dot{q}_j}\right] - \frac{\partial T}{\partial q_j} + \frac{\partial U_g}{\partial q_j} - \frac{K_i}{N_i}\left[q_{2i} - \frac{q_{2i-1}}{N_i}\right] = \tau_i \qquad \text{for } j = 2i-1$$

$$\frac{d}{dt}\left[\frac{\partial T}{\partial \dot{q}_j}\right] - \frac{\partial T}{\partial q_j} + \frac{\partial U_g}{\partial q_j} + K_i\left[q_{2i} - \frac{q_{2i-1}}{N_i}\right] = 0 \qquad \text{for } j = 2i$$

with $i = 1,\ldots,N$. τ_i is the torque supplied by the actuator at the i-th joint. Rhs zeros in the even equations mean lack of local control action at the link side of elasticity.

The dynamic model for robot arms with joint elasticity can be rewritten as:

$$B_E(\mathbf{q})\,\ddot{\mathbf{q}} + c_E(\mathbf{q},\dot{\mathbf{q}}) + e_E(\mathbf{q}) + r_E(\mathbf{q}) = \tau_E$$

In vector form the model looks similar to the one of a rigid arm but there are twice the number of second order nonlinear differential equations. The 2N-vector forcing term τ_E has even components equal to zero and odd components equal to the motor torques τ. The 2Nx2N generalized inertia matrix $B_E(\mathbf{q})$ is still symmetric and positive definite for all $\mathbf{q}$. $c_E(\mathbf{q},\mathbf{q}')$ collects the centrifugal and Coriolis terms and is related to the elements of the matrix $B_E(\mathbf{q})$ by the same relations holding for the rigid model [6]. $e_E(\mathbf{q})$ contains gravitational forces while the elastic terms are grouped in $r_E(\mathbf{q})$.

Assume now a symmetric mass distribution of the motor around its rotation axis (this implies also that its center of mass is located on the motor shaft). Under this mild assumption it can be shown that B_E, c_E and e_E depend only on link variables:

$$B_E = B_E(\mathbf{q}_e), \quad c_E = c_E(\mathbf{q}_e, \dot{\mathbf{q}}_e), \quad e_E = e_E(\mathbf{q}_e)$$

The motor joint variables $\mathbf{q}_r$ enter the nonlinear equations only through r_E, that is in a linear way like the input motor torques τ_i.

To obtain state and output equations, define the state as $\mathbf{x} = (\mathbf{x}_1,\mathbf{x}_2) = (\mathbf{q},\mathbf{q}')$ and the input as $\mathbf{u} = \tau$. For compactness set $n_E(\mathbf{q},\mathbf{q}') = c_E(\mathbf{q},\mathbf{q}') + e_E(\mathbf{q}) + r_E(\mathbf{q})$ and define a 2NxN odd-columns selection matrix B_S = block diag { $[1\ 0]^T$ } and a Nx2N even-rows selection matrix C_S = block diag { [0 1] }. Then

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{x}_2 \\ -B_E(\mathbf{x}_1)^{-1} n_E(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} 0 \\ B_E(\mathbf{x}_1)^{-1} B_S \end{bmatrix} \mathbf{u}$$

which is of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\,\mathbf{u} \qquad \mathbf{x} \in \mathbb{R}^n, \ \mathbf{u} \in \mathbb{R}^m$$

with n = 4N and m = N, where N is the number of elastic joints. Limiting ourself to joint space strategies, the outputs will be

$$\mathbf{y} = h(\mathbf{x}) = C_S\,\mathbf{x}_1 \qquad \mathbf{y} \in \mathbb{R}^p, \ p = N$$

or, in a scalar notation, $y_i = x_{1,2i} = q_{2i}$, $i = 1,\ldots,N$. The proper "joint" variables to be set under control are indeed the ones that specify the link positions. Notice that a nonlinear system of the square type (p = m = N) is obtained.

3. Linearization of systems without zero-dynamics

Consider a nonlinear system described by the equations:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + \sum_{i=1}^{m} g_i(\mathbf{x})\,u_i \qquad y_i = h_i(\mathbf{x}), \quad i = 1,\ldots,m$$

with state $\mathbf{x}$ in R^n, input $\mathbf{u}$ and output $\mathbf{y}$ in R^m. All the functions are assumed to be smooth. The problem of transforming a nonlinear system into a linear one by means of state-feedback and change of coordinates has been studied by several authors; necessary and sufficient conditions exist when static state-feedback is used [26,27]. If dynamic state-feedback laws are considered, the necessary conditions for linearization may be relaxed. In particular, the problem of full linearization for systems *with outputs* is of special interest, like in the present robotic application. It will be briefly described how constructive conditions for full linearization and input-output decoupling can be derived in the generalized case of dynamic state-feedback. Further details can be found in [24,25].

Some notation is needed first. Given a scalar function $\lambda(\mathbf{x})$ and a vector function $f(\mathbf{x})$, the Lie derivative of $\lambda(\mathbf{x})$ along $f(\mathbf{x})$, denoted by $L_f\lambda(\mathbf{x})$, is a new scalar function defined as:

$$L_f\lambda(\mathbf{x}) = \frac{\partial \lambda}{\partial \mathbf{x}} \cdot f(\mathbf{x}) = \left(\frac{\partial \lambda}{\partial x_1}, \ldots, \frac{\partial \lambda}{\partial x_n}\right) \cdot f(\mathbf{x})$$

$L_f^{\,k}\lambda(\mathbf{x})$ is defined iteratively as $L_f(L_f^{\,k-1}\lambda(\mathbf{x}))$, with $L_f^{\,0}\lambda(\mathbf{x}) = \lambda(\mathbf{x})$. This operation can be applied repeatedly and is used to define the relative degrees associated to the outputs of the system. An output y_i is said to have *relative degree* r_i (at $\mathbf{x}^\circ$) if:

$$L_{g_j} L_f^k h_i(\mathbf{x}) = 0 \quad \text{for } j = 1,\dots,m \quad \text{and} \quad k = 0,\dots,r_i - 2$$

for all $\mathbf{x}$ in a neighborhood of $\mathbf{x}^\circ$, and if for at least an index j:

$$L_{g_j} L_f^{r_i - 1} h_i(\mathbf{x}^\circ) \neq 0.$$

The relative degree of an output is connected to the integration structure of the given system; r_i is exactly the number of times one has to differentiate $y_i(t)$ at $t = t^\circ$ in order to let some of the input values $u_j(t^\circ)$ appear explicitly. Moreover, the set of relative degrees is used also to define the elements $a_{ij}(\mathbf{x})$ of the so-called decoupling matrix $A(\mathbf{x})$ associated to the system:

$$a_{ij}(\mathbf{x}) = L_{g_j} L_f^{r_i - 1} h_i(\mathbf{x})$$

Another relevant control concept which is intrinsic to a given nonlinear system with outputs is the so-called *zero-dynamics* [23]. This is the internal dynamics of the system obtained using as input a state-feedback law $\mathbf{u} = \mathbf{u}^*(\mathbf{x})$ which forces the output $\mathbf{y}$ to be constantly zero.

For SISO systems it is easy to see that, in order to have $y(t) \equiv 0$ for all times, it is required that:

$$h(\mathbf{x}(t)) = L_f h(\mathbf{x}(t)) = L_f^2 h(\mathbf{x}(t)) = \dots = L_f^{r-1} h(\mathbf{x}(t)) = 0$$

and

$$u(t) = -\frac{L_f^r h(\mathbf{x}(t))}{L_g L_f^{r-1} h(\mathbf{x}(t))} = u^*(\mathbf{x}(t))$$

The first set of conditions defines an hypersurface $M^*(\mathbf{x})$ of dimension n-r in the state space which is an *invariant set* of the closed-loop system obtained using $u(t) = u^*(\mathbf{x}(t))$. The zero-dynamics is exactly the dynamical behavior of this particular closed-loop system on $M^*(\mathbf{x})$. Note that the control $u^*(\mathbf{x}(t))$ is defined also *outside* M^*.

For a description of the problem of "zeroing the output" in a multivariable nonlinear system the reader is referred to [23,25]. However, the essential steps of the zero-dynamics algorithm can be easily recovered directly from the robot application described in the next section. Roughly speaking, it will be shown how it is possible to recover the input $\mathbf{u}(t)=\mathbf{u}^*(\mathbf{x}(t))$ from the output functions of the system and from other properly chosen functions which are *x-dependent* combinations of the output derivatives. If the algorithm terminates successfully, the system is said to be *invertible*. The set M^* is then found directly by zeroing the above functions.

It is well-known [22] that any system for which:
- the decoupling matrix $A(\mathbf{x})$ is nonsingular at $\mathbf{x}^\circ$,
- the sum of the relative degrees equals the dimension of $\mathbf{x}$,

can be transformed, locally near $\mathbf{x}^\circ$, into a linear system by means of a suitable change of coordinates and of the feedback:

$$\mathbf{u}(\mathbf{x}) = -A(\mathbf{x})^{-1} \begin{bmatrix} L_f^{r_1-1} h_1(\mathbf{x}) \\ \dots \\ L_f^{r_m-1} h_m(\mathbf{x}) \end{bmatrix}$$

The first is a necessary and sufficient condition for the existence of *noninteracting control* via static state-feedback. From the previous discussion, the second condition (i.e. $r_1 + \dots + r_m = n$) can be interpreted as the condition that the system has *no zero-dynamics* ; the hypersurface $M^*(\mathbf{x})$ degenerates into a single point, $\mathbf{x}^\circ$. The above result can be rephrased by saying that a system "without zeros" for which noninteracting control exists can always be transformed into a linear system via static state-feedback. This is the classic case of rigid manipulators.

In this perspective, milder sufficient conditions for full linearization can be obtained when dynamic state-feedback is allowed. More specifically, the assumption on the existence of noninteracting control can be relaxed; assuming only that the system is without zeros, it is possible to obtain linearity via feedback. The constructive technique that makes this possible follows the philosophy of adding a suitable number of integrators on the input channels until a system is obtained which has a nonsingular decoupling matrix (see [24]). Note that the input $\mathbf{u}^*(\mathbf{x})$ that displays the zero-dynamics can always be written as a product where the inverse of a matrix $Q(\mathbf{x})$ appears:

$$\mathbf{u}^*(\mathbf{x}) = -Q^{-1}(\mathbf{x})\, \mathbf{b}(\mathbf{x})$$

(compare with the equation above for the linearizing controller). Whenever the system has a nonsingular decoupling matrix $A(\mathbf{x})$, then $Q(\mathbf{x}) = A(\mathbf{x})$.

Consider the following *pre-processing* of the system:
- apply the static state feedback

$$\mathbf{u} = \mathbf{u}^*(\mathbf{x}) + Q^{-1}(\mathbf{x})\, \mathbf{w}$$

i.e. the control displaying the zero-dynamics of the system, with an additional external input $\mathbf{w}$;
- perform a dynamic extension, i.e. add to each of the inputs $\mathbf{w}$ a proper number of integrators, or

$$\overline{w}_i = \frac{d^{\mu_i}}{dt^{\mu_i}} w_i \qquad i = 1,\dots, m$$

where the integer μ_i is the difference between the highest and the lowest order of derivation of the output y_i, which appears in the computation of $\mathbf{u}^*(\mathbf{x})$.

Under the *invertibility* hypothesis on the original system, it is possible to show that the extended system obtained with the above pre-processing has a nonsingular decoupling matrix. Moreover, its zero-dynamics is left unchanged. Thus, if the original system had no zero-dynamics, then the extended system can be transformed into a fully linear one by means of static-state feedback and change of coordinates. In other words, denoting by $\mathbf{z}$ the states of the added integrators, the extended system will have relative degrees (with overbars) such that

$$\bar{r}_1 + \bar{r}_2 + \dots + \bar{r}_m = \bar{n} \qquad \text{where } \bar{n} = \dim \bar{\mathbf{x}} = \dim \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix}$$

Therefore, the standard static noninteracting feedback will also be a linearizing one for the extended system. By composing this feedback from the extended state with the pre-processing, the full linearizing *dynamic* controller is obtained.

4. Controller design for a two-link arm

To illustrate the above methodology, the control problem for a two-link planar robot arm with the two rotary joints being elastic (see Figure 2) is considered here as a case study. The derivation of the dynamic model and the state-equations are reported in Appendix.

For this arm, which has a kinematic structure of two rotational joints with parallel axes of rotation, neither the conditions for exact linearization nor the ones for input-output decoupling are satisfied [7,17], if *static* state-feedback is used. In the following it will be shown how to construct a linearizing and decoupling *dynamic* state-feedback law for this example.

From the output and state equations, since $L_g h(\mathbf{x}) = 0$, one gets

$$\mathbf{y} = h(\mathbf{x}) = \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} \qquad \dot{\mathbf{y}} = L_f h(\mathbf{x}) = \begin{bmatrix} x_6 \\ x_8 \end{bmatrix}$$

$$\ddot{\mathbf{y}} = L_f^2 h(\mathbf{x}) + L_g L_f h(\mathbf{x})\,\mathbf{u} \quad \Rightarrow \quad A(\mathbf{x}) = \begin{bmatrix} 0 & g_{62}(x_4) \\ 0 & g_{82}(x_4) \end{bmatrix}$$

and hence $r_1 = r_2 = 2$. The decoupling matrix $A(\mathbf{x})$ is singular and so input-output decoupling is not possible using static state-feedback. The input $\mathbf{u}$ cannot be recovered only from the knowledge of $\mathbf{y}$, $\mathbf{y}'$ and $\mathbf{y}''$. To proceed further, one has to avoid the introduction of the time derivative of the second input u_2 at the next step of the algorithm. This is done as follows. Find the coefficient of linear dependence (on the field of smooth analytic functions) of the rows of $A(\mathbf{x})$

$$\begin{bmatrix} \gamma(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} 0 & g_{62}(x_4) \\ 0 & g_{82}(x_4) \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \quad \Rightarrow \quad \gamma(x_4) = -\frac{g_{82}(x_4)}{g_{62}(x_4)}$$

and define a function $\lambda(\mathbf{x})$ as

$$\lambda(\mathbf{x}) = \begin{bmatrix} \gamma(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} L_f^2 h_1(\mathbf{x}) \\ L_f^2 h_2(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} -\dfrac{g_{82}(x_4)}{g_{62}(x_4)} & 1 \end{bmatrix} \begin{bmatrix} f_6(\mathbf{x}) \\ f_8(\mathbf{x}) \end{bmatrix}$$

or, using the system expressions in the Appendix,

$$\lambda(\mathbf{x}) = f_8(\mathbf{x}) + \left(1 + \frac{A_3}{A_2}\cos x_4\right) f_6(\mathbf{x}) =$$

$$= -\frac{K_2}{A_2 N_2}(N_2 x_4 - x_3) - \frac{A_3}{A_2} x_6^2 \sin x_4 = \lambda(x_3, x_4, x_6)$$

This function $\lambda(\mathbf{x})$ of the state will be treated as a new "output" of the system. Note that, dropping x-dependence, this function can be alternatively rewritten as $\lambda = \gamma\, y''_1 + y''_2$. At the next step, its time derivative is

$$\dot{\lambda} = L_f\lambda + L_{g_1}\lambda\, u_1 + L_{g_2}\lambda\, u_2 =$$

$$= \left[\frac{\partial\lambda}{\partial x_3} x_7 + \frac{\partial\lambda}{\partial x_4} x_8 + \frac{\partial\lambda}{\partial x_6} f_6\right] + [0]\, u_1 + \left[\frac{\partial\lambda}{\partial x_6} g_{62}\right] u_2$$

Since the input u_1 still does not appear, the above reasoning has to be repeated. Solving

$$\begin{bmatrix} \delta(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} L_g L_f h_1(\mathbf{x}) \\ L_g \lambda(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \delta(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} 0 & g_{62} \\ 0 & \dfrac{\partial\lambda}{\partial x_6} g_{62} \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

for $\delta(\mathbf{x})$ gives

$$\delta(\mathbf{x}) = -\frac{\partial\lambda}{\partial x_6} = \frac{2A_3}{A_2} x_6 \sin x_4 = \delta(x_4, x_6)$$

Another function $\mu(\mathbf{x})$ is built as

$$\mu(\mathbf{x}) = \begin{bmatrix} \delta(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} f_6 \\ L_f\lambda \end{bmatrix} =$$

$$= -\frac{K_2}{A_2 N_2}(N_2 x_8 - x_7) - \frac{A_3}{A_2} x_6^2 x_8 \cos x_4 = \mu(x_4, x_6, x_7, x_8)$$

Again, μ can be given an alternative expression in terms of $\mathbf{x}$-dependent combinations of time derivatives of the original outputs, and in which the control $\mathbf{u}$ does not appear. Taking the time derivative of this function

$$\dot{\mu} = L_f\mu + L_{g_1}\mu\, u_1 + L_{g_2}\mu\, u_2 =$$

$$= \left[\frac{\partial\mu}{\partial x_4} x_8 + \frac{\partial\mu}{\partial x_6} f_6 + \frac{\partial\mu}{\partial x_7} f_7 + \frac{\partial\mu}{\partial x_8} f_8\right] + \left[\frac{\partial\mu}{\partial x_6} g_{62} + \frac{\partial\mu}{\partial x_7} g_{72} + \frac{\partial\mu}{\partial x_8} g_{82}\right] u_2$$

gives again no information on u_1. The coefficient $\varepsilon(\mathbf{x})$ of linear dependence between the rows of the matrix multiplying $\mathbf{u}$ at this step is found by solving

$$\begin{bmatrix} \varepsilon(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} 0 & g_{62} \\ 0 & L_{g_2}\mu \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \quad \Rightarrow \quad \varepsilon(\mathbf{x}) = -\frac{L_{g_2}\mu}{g_{62}}$$

that gives

$$\varepsilon(\mathbf{x}) = -\frac{K_2}{A_2 N_2}\left(N_2 \frac{g_{82}}{g_{62}} - \frac{g_{72}}{g_{62}}\right) + \frac{A_3}{A_2} x_6 \left(2x_8 + \frac{g_{82}}{g_{62}} x_6\right) \cos x_4$$

so that $\varepsilon(\mathbf{x}) = \varepsilon(x_4, x_6, x_8)$ since

$$\frac{g_{82}}{g_{62}} = -\left(1 + \frac{A_3}{A_2}\cos x_4\right) \qquad \frac{g_{82}}{g_{62}} = \frac{G_2}{A_2}\left(A_3^2 \cos^2 x_4 + A_4\right) - 1.$$

The coefficient $\varepsilon(\mathbf{x})$ is used for defining a third function $v(\mathbf{x})$ as

$$v(\mathbf{x}) = \begin{bmatrix} \varepsilon(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} f_6(\mathbf{x}) \\ L_f\mu(\mathbf{x}) \end{bmatrix} =$$

$$= \frac{\partial\mu}{\partial x_4} x_8 + \frac{\partial\mu}{\partial x_7}\left[f_7 - f_6 \frac{g_{72}}{g_{62}}\right] + \frac{\partial\mu}{\partial x_8}\left[f_8 - f_6 \frac{g_{82}}{g_{62}}\right] =$$

$$= \frac{A_3}{A_2}(x_6 x_8)^2 \sin x_4 + \frac{G_2 K_2}{A_2 N_2}\Big[\frac{K_1}{N_1}(N_1 x_2 - x_1) - \frac{K_2}{N_2}(N_2 x_4 - x_3)\left(1 + \frac{A_3}{A_2}\cos x_4\right)$$

$$- A_3 \sin x_4 \left(\frac{A_3}{A_2} x_6^2 \cos x_4 + (x_6 + x_8)^2\right)\Big]$$

$$+ \frac{K_2 + A_3 x_6^2 \cos x_4}{A_2}\left[\frac{K_2}{A_2 N_2}(N_2 x_4 - x_3) + \frac{A_3}{A_2} x_6^2 \sin x_4\right] =$$

$$= v(x_1, x_2, x_3, x_4, x_6, x_8)$$

Note that this "output" function does not depend on x_5; this implies that its time derivative

$$\dot{v} = L_f v + L_{g_1} v\, u_1 + L_{g_2} v\, u_2 =$$

$$= \left[\frac{\partial v}{\partial x_1} x_5 + \frac{\partial v}{\partial x_2} x_6 + \frac{\partial v}{\partial x_3} x_7 + \frac{\partial v}{\partial x_4} x_8 + \frac{\partial v}{\partial x_6} f_6 + \frac{\partial v}{\partial x_8} f_8\right] + \left[\frac{\partial v}{\partial x_6} g_{62} + \frac{\partial v}{\partial x_8} g_{82}\right] u_2$$

is still independent from input u_1. An $\mathbf{x}$-dependent coefficient $\zeta(\mathbf{x})$ is found as

$$\zeta(\mathbf{x}) = -\frac{L_{g_2} v}{g_{62}} = \zeta(x_3, x_4, x_6, x_8)$$

where the functional dependencies follow from the expression of $L_{g_2} v$. A fourth function $\xi(\mathbf{x})$ is computed as

$$\xi(\mathbf{x}) = \begin{bmatrix} \zeta(\mathbf{x}) & 1 \end{bmatrix} \begin{bmatrix} f_6(\mathbf{x}) \\ L_f v(\mathbf{x}) \end{bmatrix} = \zeta(\mathbf{x})\, f_6(\mathbf{x}) + L_f v(\mathbf{x}) = \frac{\partial v}{\partial x_1} x_5 + \phi(\mathbf{x})$$

where $\phi(\mathbf{x})$ contains all terms which do not depend on x_5. From the expression of $v(\mathbf{x})$

$$\frac{\partial v}{\partial x_1} = -\frac{K_1 K_2}{A_2 N_1 N_2} \neq 0$$

so that x_5 appears linearly in $\xi(\mathbf{x})$. The time derivative of the function $\xi(\mathbf{x})$ yields

$$\dot{\xi} = L_f\xi + L_{g_1}\xi\ u_1 + L_{g_2}\xi\ u_2 = L_f\xi + [\frac{\partial \xi}{\partial x_5} g_{51}]\ u_1 + L_{g_2}\xi\ u_2$$

where finally the control input u_1 is explicitly present. The full input $\mathbf{u}$ can be recovered from the equation

$$\begin{bmatrix} \ddot{y}_1 \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} L_f^2 h_1(\mathbf{x}) \\ L_f\xi(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} L_{g_1}L_f h_1(\mathbf{x}) & L_{g_2}L_f h_1(\mathbf{x}) \\ L_{g_1}\xi(\mathbf{x}) & L_{g_2}\xi(\mathbf{x}) \end{bmatrix} \mathbf{u} = \mathbf{b}(\mathbf{x}) + \mathbf{Q}(\mathbf{x})\,\mathbf{u}$$

where $Q(\mathbf{x})$ is a nonsingular (here, triangular) matrix. Since this equation is solvable in $\mathbf{u}$, then the robotic system is invertible. As already mentioned, all functions introduced in the algorithm may be expressed in terms of y_1, y_2 and of their time derivatives. In particular, ξ' contains combinations of derivatives of the first output, from $y_1^{(2)}$ to $y_1^{(6)}$, but only the sixth derivative of the second one (i.e. $y_2^{(6)}$).

Setting the lhs of the above equation to zero yields the input $\mathbf{u} = -Q^{-1}(\mathbf{x})\ b(\mathbf{x}) = \mathbf{u}^*(\mathbf{x})$ which displays the zero-dynamics of the system. It is easy to see that the given robot with joint elasticity has no zero-dynamics: the hypersurface $M^*(\mathbf{x})$ is reduced to a single point, $\mathbf{x}° = \mathbf{0}$. In fact, M^* is defined by imposing a zero-constrained output behavior. This is equivalent to find all $\mathbf{x}$'s which solve the following set of eight equations:

$$\begin{bmatrix} \mathbf{y}(\mathbf{x}) & \dot{\mathbf{y}}(\mathbf{x}) & \lambda(\mathbf{x}) & \mu(\mathbf{x}) & \nu(\mathbf{x}) & \xi(\mathbf{x}) \end{bmatrix} = \mathbf{0} \ \Rightarrow\ M^* \equiv \mathbf{0}$$

The above implication is checked by looking at the triangular structure of these equations. Using the unique solution of the first four ($x_2 = x_4 = x_6 = x_8 = 0$) into the fifth equation (λ=0) yields $x_3 = 0$; using these values into the sixth one (μ=0) gives $x_7 = 0$, and so on.

In order to construct the linearizing and decoupling control for the given two-link robot arm, first apply the *static* state-feedback law $\mathbf{u} = Q^{-1}(\mathbf{x})\ [\mathbf{w} - b(\mathbf{x})] = \mathbf{u}^*(\mathbf{x}) + Q^{-1}(\mathbf{x})\ \mathbf{w}$, or

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 & \dfrac{A_2}{A_3^2\cos^2 x_4 + A_4} \\ -\dfrac{G_1K_1K_2}{A_2N_1N_2} & L_{g_2}\xi(\mathbf{x}) \end{bmatrix}^{-1} \left\{ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} - \begin{bmatrix} f_6(\mathbf{x}) \\ L_f\xi(\mathbf{x}) \end{bmatrix} \right\} =$$

$$= \alpha(\mathbf{x}) + \beta(\mathbf{x})\ \mathbf{w}$$

which is the feedback derived directly from the computation of the zero-dynamics. The obtained system can be represented in a nice form as in Figure 3. On this graph it is easy to verify that the system has no zero-dynamics; it is also simple to see which additional steps have to be taken in order to get decoupling and linearization.

The system is then extended by adding a proper number of integrators to the inputs. In this case, $\mu_1 = 4$ while $\mu_2 = 0$. Note that for two-input two-output systems, one input is *always* left unchanged. So, *four* integrators have to be added on input w_1

$$w_1 = z_1 \quad \dot{z}_1 = z_2 \quad \dot{z}_2 = z_3 \quad \dot{z}_3 = z_4 \quad \dot{z}_4 = \bar{w}_1$$

$$w_2 = \bar{w}_2$$

As a result of this preprocessing, a system is obtained with state

$$\bar{\mathbf{x}} = \begin{bmatrix} \mathbf{x}^T & \mathbf{z}^T \end{bmatrix}^T \in \mathbb{R}^{12}$$

and described by the following equations (see also Figure 3):

$$\dot{\bar{\mathbf{x}}} = \bar{f}(\bar{\mathbf{x}}) + \bar{g}(\bar{\mathbf{x}})\ \bar{\mathbf{w}} \qquad \mathbf{y} = \bar{h}(\bar{\mathbf{x}}) = h(\mathbf{x})$$

where

$$\bar{f}(\bar{\mathbf{x}}) = \begin{bmatrix} x_5 & x_6 & x_7 & x_8 & \bar{f}_5 & z_1 & \bar{f}_7 & \bar{f}_8 & z_2 & z_3 & z_4 & 0 \end{bmatrix}^T$$

$$\bar{g}(\bar{\mathbf{x}}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \bar{g}_{52} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

and

$$\bar{f}_5(\bar{\mathbf{x}}) = f_5(\mathbf{x}) + \frac{L_{g_2}\xi(\mathbf{x})}{L_{g_1}\xi(\mathbf{x})}\frac{g_{51}}{g_{62}(\mathbf{x})}(f_6(\mathbf{x}) - z_1) - \frac{L_f\xi(\mathbf{x})}{L_{g_1}\xi(\mathbf{x})}g_{51}$$

$$\bar{f}_7(\bar{\mathbf{x}}) = f_7(\mathbf{x}) - \frac{g_{72}(\mathbf{x})}{g_{62}(\mathbf{x})}(f_6(\mathbf{x}) - z_1)$$

$$\bar{f}_8(\bar{\mathbf{x}}) = f_8(\mathbf{x}) - \frac{g_{82}(\mathbf{x})}{g_{62}(\mathbf{x})}(f_6(\mathbf{x}) - z_1) \qquad \bar{g}_{52}(\bar{\mathbf{x}}) = \frac{g_{51}}{L_{g_1}\xi(\mathbf{x})}$$

It is easy to check that the decoupling matrix of this extended system is

$$\bar{A}(\bar{\mathbf{x}}) = L_{\bar{g}}L_{\bar{f}}^5\bar{h}(\bar{\mathbf{x}}) = \begin{bmatrix} 1 & 0 \\ \gamma(x_4) & 1 \end{bmatrix}$$

which is *globally* nonsingular. The relative degrees are both equal to six and their sum gives the dimension of the extended state. Hence, the standard noninteracting control [22]

$$\bar{\mathbf{w}} = \bar{A}(\bar{\mathbf{x}})^{-1}[\ \mathbf{v} - L_{\bar{f}}^6\bar{h}(\bar{\mathbf{x}})\ ] = \bar{\alpha}(\bar{\mathbf{x}}) + \bar{\beta}(\bar{\mathbf{x}})\ \mathbf{v}$$

will also linearize the extended system. The composition of the preprocessing and of this additional static feedback yields the linearizing *dynamic* state-feedback controller.

The input-state-output behavior between the reference inputs v_1, v_2 and the original outputs y_1 and y_2 is decoupled and linear. This is made evident once a change of coordinates is performed [22]. In the new coordinates θ specified by

$$\theta_i = L_{\bar{f}}^{i-1}\bar{h}_1(\bar{\mathbf{x}}), \quad i = 1,\ldots,6 \qquad \theta_i = L_{\bar{f}}^{i-1}\bar{h}_2(\bar{\mathbf{x}}), \quad i = 7,\ldots,12$$

the system takes on the canonical form of two strings of *six* integrators each. This set of linear coordinates is given just by the system outputs, namely the angular positions of the two links of the robot, together with their time derivatives up to the sixth order. As usual, the design of a linear stabilizing feedback for trajectory control has to be done in terms of this set of coordinates.

5. Discussion

It has been shown that a planar two-link robot with joint elasticity can be fully linearized and input-output decoupled using dynamic rather than only static feedback. control laws are allowed. As a matter of fact, this is a more general result: the *whole* class of robots with joint elasticity can be linearized once the larger class of dynamic state-feedback control laws is considered. The control problem for a series of robots with different kinematic types, including structures with mixed sets of rigid and elastic joints, has been solved in [17,18]. Similar linearization results hold also if the outputs are chosen in the task space, except for the presence of kinematic singularities.

Like for any rigid manipulator, the resulting closed-loop system is linear and decoupled. However, the main difference is in the length of the input-output chains of integrators which, for each elastic joint, is variable but is never less than four, as opposed to the constant double-integration structure of the rigid case. For the two-link planar robot, this length is six and is obtained with the help of the states of the compensator. Note that the lower bound of four input-output integrations is intrinsic to the purely elastic coupling between bodies. This can be checked by analyzing the simple one-dimensional case of two rigid masses elastically coupled, with a force acting at one side while observations are taken on the other.

The variability of this integration structure is a rather surprising aspect [28]. The reason of such behavior is related to the physics of the problem, namely to the kind of interactions that arise between the elastic and the rigid degrees of freedom of a multi-jointed arm. Including joint elasticity in the model creates two sources of dynamic interaction among links and actuators: elastic reaction forces $r_E(\mathbf{q})$, and inertial couplings (i.e. the off-diagonal terms of the inertia matrix $B_E(\mathbf{q})$). The actuator torques affect the motion of the robot links using both these dynamic pathways. Usually, the one going through inertial couplings has lower control authority. In any case, the paths which are faster, measured in terms of integration structure, dominate over the others. It may happen that one of the input torques is "felt" at different links before all of the other ones. No *instantaneous* feedback action performed on the system is capable of uncoupling these interaction effects. Moreover, the possibility of full compensation of model nonlinearities is lost. This happens whenever there exist inertial coupling paths which are faster than the ones due to elastic forces, as in the examined planar case. There are particular kinematic structures in which these inertial paths are not present, already at a *mechanical* level. This is true, for example, for pairs of elastic joints having orthogonal axes. This explains the results obtained for a two-link cylindric [16] and a polar [11,17] arm. It is also evident that the "racing" situation among torques does not occurr for one-dimensional systems. This idea is consistent with the result of feedback linearizability obtained in the single-joint elastic case [15].

The above discussion helps to understand the role of dynamic feedback in the control of robots with joint elasticity. In particular, dynamic compensation *delays* the contributions of the inertial interacting effects so that each input torque affects the relative output (the link position) after the same number of integrations. By slowing down these fast but "weak" actions, the high authority control paths are brought into play and cancellation of the nonlinearities becomes possible. Of course, if the inertial interaction paths are neglected as in the approximate modeling proposed in [19], decoupling and linearization are again possible using only static feedback, although a more complex one than in the rigid case. Roughly speaking, dynamic compensation gives robustness w.r.t. these parasitic effects. Moreover, the *balancing* role of dynamic feedback becomes even clearer when arms with mixed types of joints are considered (see [18]). In any case, there is essentially a trade-off between modeling accuracy and use of more sophisticated control laws.

Some remarks are in order on the choice of reference trajectories for this class of robot arms. The length of the linear chains of closed-loop integrators puts obvious smoothness constraints on the class of desirable time evolutions. If the outputs have to reproduce exactly a given reference trajectory, this should possess continuous time derivatives at least up to the third order. For the two-link planar arm this requirement increases up to the fifth order; the reference inputs $\mathbf{v}$ may be, for instance, the piecewise constant sixth derivative of the desired link motion. If the initial state is properly set, this allows exact output tracking of the trajectory. Otherwise, only asymptotic tracking is guaranteed, once the system has been externally stabilized. As intuition suggests, robots with joint elasticity have to be driven with very smooth reference commands.

Finally, it should be mentioned that the whole approach requires availability of the full state of the system. The measure of position and velocity at both sides of the elastic joint is tecnically feasible, although additional instrumentation is needed. Except for [12], full-state availability is a common request of all the proposed control methods [7-11]. Some work is in progress also on the design of exact linear observer for this class of robots [29].

References

[1] Good,M.C.,Sweet,L.M.,Strobel,K.L., Dynamic Models for Control System Design of Integrated Robot and Drive Systems,*ASME J. Dyn.Systems, Meas, and Control*, **107**, 1985.

[2] Rivin, E.I., Zeid, A., Rastgu-Ghamsari, A., Dynamic Effects Associated with Joint Compliance in Cartesian-Frame Manipulators, *1st IFAC Symp. Robot Control*, Barcelona, 1985.

[3] Sweet, L.M., Good, M.C., Redefinition of the Robot Motion Control Problem,*IEEE Control Systems Mag.*, **5**, 3,1985.

[4] Kuntze, H.B., Jacubasch, A.H.K., Control Algorithms for Stiffening an Elastic Industrial Robot, *IEEE J. Robotics and Automation*, **RA-1**, 2, 1985.

[5] Nicosia, S., Nicolò, F., Lentini, D., Dynamical Control of Industrial Robots with Elastic and Dissipative Joints, *8th IFAC World Congress*, Kyoto, 1981.

[6] Cesareo, G., Nicolò, F., Nicosia, S., DYMIR: A Code for Generating Dynamic Model of Robots, *1st IEEE Int. Conf. Robotics and Automation* , Atlanta, 1984.

[7] Marino, R, Nicosia, S., On the Feedback Control of Industrial Robots with Elastic Joints: A Singular Perturbation Approach, *Università di Roma "Tor Vergata"*, Rap.84.01, 1984.

[8] Khorasani, K., Spong, M.W., Invariant Manifolds and Their Application to Robot Manipulators with Flexible Joints, *2nd IEEE Int. Conf. Robotics and Automation*, St.Louis, 1985.

[9] Khorasani, K., Kokotovic, P.V., Feedback Linearization of a Flexible Manipulator Near its Rigid Body Manifold, *Systems & Control Lett.*, 6, 1985.

[10] Slotine, J.J., Hong, S., Two-Time Scale Sliding Control of Manipulators with Flexible Joints, *American Control Conference*, Seattle, 1986.

[11] Nicosia, S.,Tomei, P., Tornambé, A., Feedback Control of Elastic Robots by Pseudo-Linearization Techniques, *25th IEEE Conf. Decision and Control*, Athens, 1986.

[12] Tomei, P., Nicosia, S., Ficola, A., An Approach to the Adaptive Control of Elastic at Joints Robots, *3rd IEEE Int. Conf. Robotics and Automation*, S.Francisco, 1986.

[13] Cesareo, G., Marino, R., On the Controllability Properties of Elastic Robots, *6th Int. Conf. Analysis and Optimization of Systems*, Nice, 1984.

[14] De Simone, C., Nicolò, F., On the Control of Elastic Robots by Feedback Decoupling, *IASTED Int. J. Robotics and Automation*, **1**, 2, 1986.

[15] Marino,R., Spong, M.W.,Nonlinear Control Techniques for Flexible Joint Manipulators: A Single Link Case Study, *3rd IEEE Int. Conf. Robotics and Automation*, S.Francisco, 1986.

[16] Forrest-Barlach, M.G., Babcock, S.M., Inverse Dynamics Position Control of a Compliant Manipulator, *3rd*

IEEE Int. Conf. Robotics and Automation, S.Francisco, 1986.
[17] De Luca A., Control of Robot Arms with Joint Elasticity: A Differential-Geometric Approach, Ph.D.Thesis (in Italian), *Università di Roma "La Sapienza"*, December 1986.
[18] De Luca A., Control Properties of Robot Arms With Joint Elasticity, *Symp.Math.Theo.of Networks Systems*, Phoenix, 1987
[19] Spong M.W., Modeling and Control of Elastic Joint Robots, *ASME Winter Annual Meeting*, Anaheim, 1986.
[20] De Luca, A., Isidori, A., Nicolò, F., An Application of Nonlinear Model Matching to the Control of Robot Arm with Elastic Joints, *1st IFAC Symp. Robot Control*, Barcelona, 1985.
[21] De Luca, A., Isidori, A., Nicolò, F., Control of Robot Arms with Elastic Joints Via Nonlinear Dynamic Feedback, *24th IEEE Conf. Decision and Control*, Ft. Lauderdale, 1985.
[22] Isidori, A., Nonlinear Control Systems: An Introduction, *Lect. Notes in Control and Information Sci.*, **72**, Springer, 1985.
[23] Isidori, A., Moog, C.H., On the nonlinear equivalent of the notion of transmission zeros, *Modeling and Adaptive Control* (C.I.Byrnes, A.H.Kurszanski Eds.), Springer, 1987.
[24] Isidori, A., Moog, C.H., De Luca, A., A Sufficient Condition for Full Linearization Via Dynamic State Feedback, *25th IEEE Conf. Decision and Control*, Athens, 1986.
[25] De Luca, A., Isidori, A., Feedback Linearization of Invertible Systems, *Colloquium on Robotics and Automation*, Duisburg, 1987.
[26] Jakubczyk, B., Respondek, W., On Linearization of Control Systems, *Bull.Acad.Pol.Sci.Ser.Math.Phys.*, **28**, 1980.
[27] Hunt, L.R., Su, R., Meyer, G., Design for Multi-Input Nonlinear Systems, in: *Differential Geometric Control Theory* (R.W.Brockett, R.S.Millman, H.Sussmann Eds.), Birkhauser, 1983.
[28] Vidyasagar, M., Systems Theory and Robotics, *IEEE Control Systems Mag.*, April 1987.
[29] Nicosia, S., Tomei, P., Tornambé, A., A Nonlinear Observer for Elastic Robots, to appear on *IEEE J. Robotics and Automation*.

Appendix. A two-link robot with joint elasticity

The kinetic energy T of the arm in Figure 2 is the sum of the terms:

$$T_{mot1} = \frac{1}{2} JR_1 \dot{q}_1^{\,2} \qquad T_{link1} = \frac{1}{2} m_1 d_1^2 \dot{q}_2^{\,2} + \frac{1}{2} J_1 \dot{q}_2^{\,2}$$

$$T_{mot2} = \frac{1}{2} mr_2 l_1^2 \dot{q}_2^{\,2} + \frac{1}{2} JR_2 (\dot{q}_2 + \dot{q}_3)^2$$

$$T_{link2} = \frac{1}{2} m_2 [l_1^2 \dot{q}_2^{\,2} + d_2^2 (\dot{q}_2 + \dot{q}_4)^2 + 2 d_2 l_1 \dot{q}_2 (\dot{q}_2 + \dot{q}_4) \cos q_4] + \frac{1}{2} J_2 (\dot{q}_2 + \dot{q}_4)^2$$

The expressions of f(**x**), g(**x**) and h(**x**) are given below:

$$f(\mathbf{x}) = \begin{bmatrix} x_5 & x_6 & x_7 & x_8 & f_5(\mathbf{x}) & f_6(\mathbf{x}) & f_7(\mathbf{x}) & f_8(\mathbf{x}) \end{bmatrix}^T$$

$$g(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 0 & 0 & g_{51}(\mathbf{x}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & g_{62}(\mathbf{x}) & g_{72}(\mathbf{x}) & g_{82}(\mathbf{x}) \end{bmatrix}^T$$

$$h(\mathbf{x}) = \begin{bmatrix} x_2 & x_4 \end{bmatrix}^T$$

with

$$g_{51} = G_1 \qquad g_{62}(x_4) = \frac{A_2}{A_3^2 \cos^2 x_4 + A_4}$$

$$g_{72}(x_4) = G_2 - g_{62}(x_4) \qquad g_{82}(x_4) = -\frac{A_3 \cos x_4 + A_2}{A_3^2 \cos^2 x_4 + A_4}$$

$$f_5(x_1, x_2) = \frac{G_1 K_1}{N_1^2} [N_1 x_2 - x_1]$$

$$f_6(x_1,x_2,x_3,x_4,x_6,x_8) = \frac{1}{A_3^2 \cos^2 x_4 + A_4} \{ \frac{K_1 A_2}{N_1} [N_1 x_2 - x_1]$$

$$+ \frac{K_2 [A_2 - N_2 (A_3 \cos x_4 + A_2)]}{N_2^2} [N_2 x_4 - x_3] - A_3 \sin x_4 [A_3 x_6^2 \cos x_4 + A_2 (x_6 + x_8)^2] \}$$

$$f_7(x_1,x_2,x_3,x_4,x_6,x_8) = -f_6(\mathbf{x}) + \frac{G_2 K_2}{N_2^2} [N_2 x_4 - x_3]$$

$$f_8(x_1,x_2,x_3,x_4,x_6,x_8) = \frac{1}{A_3^2 \cos^2 x_4 + A_4} \{ -\frac{K_1 (A_3 \cos x_4 + A_2)}{N_1} [N_1 x_2 - x_1] \cdot$$

$$+ \frac{K_2 [(N_2 - 1)(A_3 \cos x_4 + A_2) + N_2 (A_3 \cos x_4 - \frac{A_4}{A_2})]}{N_2^2} [N_2 x_4 - x_3]$$

$$+ A_3 \sin x_4 [(A_3 \cos x_4 - \frac{A_4}{A_2}) x_6^2 + (A_3 \cos x_4 + A_2)(x_6 + x_8)^2] \}$$

Besides $G_i = 1/JR_i$, the following constants were introduced:

$$A_1 = (J_1 + m_1 d_1^2) + (J_2 + m_2 d_2^2) + JR_2 + (m_2 + mr_2) l_1^2$$

$$A_2 = J_2 + m_2 d_2^2 \qquad A_3 = m_2 l_1 d_2 \qquad A_4 = A_2 (\frac{1}{G_2} + A_2 - A_1)$$

Figure 1

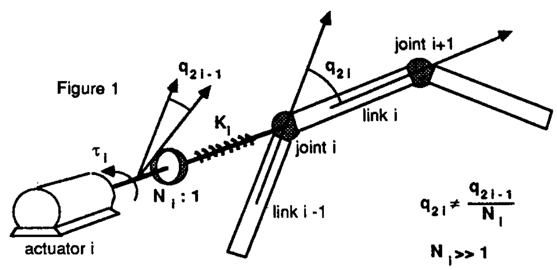
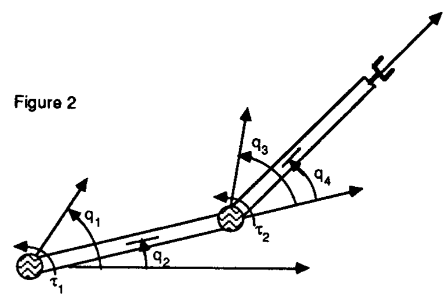


Figure 2



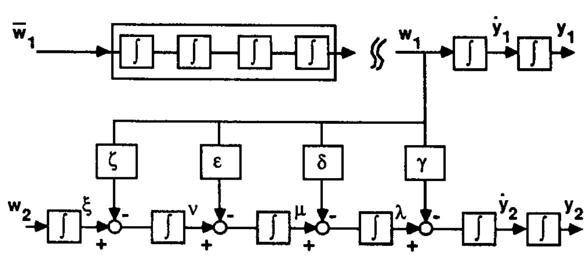


Figure 3