AN APPLICATION OF NONLINEAR MODEL MATCHING TO THE DYNAMIC CONTROL OF ROBOT ARM WITH ELASTIC JOINTS

A. De Luca, A. Isidori and F. Nicolo
Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza".
Via Eudossiana 18, 00184 Roma, Italy

Abstract. The paper is concerned with dynamic control of robot arm with non negligible joint elasticity. In this case nonlinear control techniques based on decoupling and nonlinearity compensation via static state-feedback cannot be applied for the most common robot structures. This is because the mathematical model associated with these structures is such that the necessary and sufficient conditions for the existence of the decoupling control laws fail to hold. Motivated by these facts, the problem of using a dynamic state-feedback is considered. The main purpose of the paper is to show that one can design a dynamic state-feedback compensator, which makes the external behavior of the controlled robot identical to the one of a prescribed decoupled linear model. The particular example considered is a planar robot arm with two elastic joints.

Keywords. Nonlinear control systems; decoupling; model matching; robots; system graphs.

INTRODUCTION

The problem of decoupling the dynamics of robot arms via static state-feedback has received large attention in the last years and is an appealing approach for achieving a complete control of the external behavior of an anthropomorphic manipulator.

For robots with rigid joints this control technique has been labeled in a number of different ways in the past (see e.g. Brady and others, 1982) and it has been shown that the resulting control is robust (Nicola, Nicola and Lentini, 1981); moreover the nonlinear decoupling has the nice property that it completely linearizes the state dynamics (Tarn and others, 1984).

For robots with elastic transmission between actuators and arms, as belts or harmonic drives, the effect of joint elasticities is such that the decoupling technique cannot be applied for the most common robot mechanical structures. This is because the mathematical model associated with these structures is such that the necessary and sufficient conditions for the existence of a decoupling law fail to hold. There are special but rather unusual mechanical structures for which such conditions hold (De Simone and Nicola, 1985); however, in these cases, the computation of the relevant control laws is quite cumbersome. In alternative, an approximate decoupling may be obtained using a controller based on singular perturbation theory as done by Ficola, Mariano and Nicola (1983).

In this paper we show that exact decoupling can be achieved for robot with elastic joints via the nonlinear model matching theory developed by Di Benedetto and Isidori (1984). In this case the resulting control law is a dynamic state-feedback which makes the external behavior of the controlled robot identical to the one of a prescribed decoupled linear model.

Starting from mechanical parameters, the model (1) is given automatically by the DYMIR code both for rigid and elastic robots (Cesareo, Nicola and Nicola, 1993). Eq. (1) may be rewritten in the standard form

\[ \dot{\bf x} = f(\bf x) + g(\bf x) \bf u \]

\[ y = h(\bf x) \]

with state \( \bf x = \begin{bmatrix} \bf x_1^T & \bf x_2^T \end{bmatrix}^T \), \( \bf u \in \mathbb{R}^m \) and output \( y \in \mathbb{R}^n \). In the elastic case \( n = 2N \), \( n = 2N \). The vector \( f \) and the \( n \) columns \( \bf e_1, \ldots, \bf e_n \) of \( f \) and the \( n \) columns \( \bf e_1, \ldots, \bf e_n \) of \( g \) are given by

\[ f = \begin{bmatrix} \bf f_1^T & \bf f_2^T \end{bmatrix}^T \]

\[ g = \begin{bmatrix} \bf g_1^T & \bf g_2^T \end{bmatrix}^T \]

where \( \bf f_1 \) and \( \bf f_2 \) are \( n \times 1 \) vectors and \( \bf g_1 \) and \( \bf g_2 \) are \( n \times m \) matrices.
of the matrix $g$ are smooth vector fields defined on an open subset of $\mathbb{R}^n$; the expressions for $f$ and $g$ are given by:

$$
f(x) = \begin{bmatrix} x_1 \\ -B(x)h_1 - (C(x) + \delta(x))v(x) \\ \vdots \\ -B(x)h_m - (C(x) + \delta(x))v(x) \\ \end{bmatrix},
$$

$$
g(x) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \end{bmatrix}.
$$

Moreover, in (2) $h$ is a smooth vector-valued function; in our case the output $y$ may be defined as the vector of link displacements $q_i = q_{2i}$ $(i = 1, \ldots, n)$. The input $u$ collects only the non-zero components of $m(x)$.

An attractive approach to the control of nonlinear systems is the one based on decoupling state-feedback laws. The static state-feedback decoupling problem is defined as follows: given a system in the form (2), find a feedback $a(x)$ and a state dependent change of coordinates $\beta(x)$ in the input space such that the closed-loop system formed by the composition of (2) with $u = a(x) + h(x)v$ has the $i$-th output dependent only on the $i$-th component of the new input.

The problem is now well understood (Isidori and co-workers, 1981; Sinha, 1977); the basic concepts of decoupling theory are summarized in what follows. Given a vector field $f$ on $\mathbb{R}^n$ and a smooth function $h$, recall that the Lie derivative of $h$ in the direction of the field $f$ is the function

$$
L_h f = \{f, h\}
$$

The k-th order Lie derivative is $L^k h = L_i L^i h$ while $L^0 h = h$.

**Definition 1.** The characteristic number $\gamma_i$ associated with the output $y_i$ is the largest integer such that for all $k < \gamma_i$:

$$
L^k h = 0, \quad \forall \xi \in \{1, \ldots, m\},
$$

In the nondegenerate case ($\gamma_i \neq m, \forall j$) we have:

**Definition 2.** The decoupling matrix $A(x)$ associated with the system (2), with $m = \gamma_i$, is the $k \times \ell$ matrix with smooth entries

$$
a_{ij}(x) = L_i L_j h.
$$

The main result of static state-feedback decoupling theory given by Isidori and co-workers (1981) is:

**Theorem 1.** A necessary and sufficient condition for the existence of $(\alpha(x), \beta(x))$ which solves the decoupling problem is that the decoupling matrix $A(x)$ is nonsingular.

When Theorem 1 applies, a possible decoupling control is given by (3) with:

$$
\alpha(x) = A(x)^{-1} A(x)h_1, \ldots, A(x)^{-1} h_m,
$$

$$
\beta(x) = A(x)^{-1}.
$$

It has been shown (Freund, 1982; Nicolosi, Nicolò and Luntini, 1985; Tarn and others, 1984) that for the rigid robot the corresponding matrix $A(x)$ is nonsingular so that (4) provides a solution to the decoupling problem. On the contrary, in the elastic case considered here the matrix $A(x)$ is in general singular (De Simone and Nicolò, 1985), except for particular mechanical structures. We will see later on that the condition of Theorem 1 fails to hold for a configuration present in most robots. As a consequence, static state-feedback is no longer sufficient in order to solve the decoupling problem when joint elasticity is taken into account.

**MODEL MATCHING VIA DYNAMIC STATE FEEDBACK**

In this section we summarize some recent results concerned with the use of dynamic feedback compensation in order to match a prescribed input-output behavior and we find, as a byproduct, a sufficient condition for decoupling via dynamic feedback.

A dynamic state-feedback is a control mode in which the inputs $v_1, \ldots, v_n$ are related to the state $x$ of the process (2) and to other input variables $v_{n+1}, \ldots, v_{n+n}$ by means of equations of the form

$$
u = a(x) + h(x)v
$$

These equations characterize a dynamical system = the state-feedback compensator = whose state $x$ evolves on an open subset of $\mathbb{R}^n$. The $\ell \times 1$ vector $a(x,E;)$, the $m \times \ell$ matrix $b(x,E;)$, the $\ell \times 1$ vector $c(x,E;)$ and the $m \times 1$ matrix $d(x,E;)$ have entries which are smooth functions defined on an open subset of $\mathbb{R}^n = \mathbb{R}^n$.

The composition of (2) with (5) defines a new dynamical system with inputs $v_1, \ldots, v_{n+n}$ and state $\hat{x} = (x,v)$ given by equations of the form

$$
\dot{\hat{x}}(t) = f(\hat{x}(t)) + g(\hat{x}(t))v
$$

with

$$
\dot{\hat{x}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{v}(t) \end{bmatrix},
$$

$$
\begin{bmatrix} \dot{x}(t) \\ \dot{v}(t) \end{bmatrix} = \begin{bmatrix} f(x(t), v(t)) \\ g(x(t), v(t)) \end{bmatrix}.
$$

$$(\hat{x}(E;)) = \hat{h}((x,E;))
$$

We shall see that dynamic compensation enables us to match, if some conditions are verified, a prescribed behavior between inputs and outputs in the composed (closed-loop) system (6). In particular, we will be able to obtain a response of the form

$$
y(t) = Q(t,E;)(E;\tilde{x}(t))
$$

where $Q(t)$ is the impulse-response of a fixed linear model

$$
y(t) = \mathcal{L} \{\mathcal{L}^{-1} \{y(t)\} + \mathcal{L}^{-1} \{y(t)\} + \mathcal{L}^{-1} \{y(t)\}\}
$$

On the first term $Q(t,E;)(E;\tilde{x}(t))$ of the right-hand-side of (7), which corresponds to the zero-input behavior (and clearly depends - possibly in a non-linear manner - on the initial state), we do not impose at this point any particular constraint.

If the process (2) and the compensator (5) were linear, then a well known necessary and sufficient condition for the existence of solutions to this problem would be the one based on the comparison of the behavior of the transfer functions of the process and of the model as $s \to (\text{Malabre}, 1982)$. We recall that the behavior of a strictly proper transfer function $W(s)$ for $s \to (\text{Malabre}, 1982)$ is described by the so-called Smith-McMillan factorization at the infinity (Dion and Commault, 1982). $W(s)$ is given by:

$$
W(s) = \frac{Q(s)}{P(s)}
$$

in which $Q(s)$ and $P(s)$ are biproper rational matrices (i.e., proper rational matrices with an inverse which is also proper) and
The sequence \( \{c_1, c_2, \ldots\} \) is said to characterize the structure at infinity of the given linear system. Sometimes, one considers a sequence \( \{r_0, r_1, \ldots\} \) related to the former in this way:

\[
{r_0} = \delta_1, \quad {r_i} = \delta_i + {r_{i-1}}, \quad i \geq 1
\]

and whose computation is rather easy on a realization \((A, B, C)\) of the transfer function \(\mathcal{W}(s)\).

If the process to be compensated is nonlinear, one may still establish a sufficient condition for the existence of solutions to a model matching problem by means of a suitable extension of the notion of structure at infinity, see Almeijer and Schumacher (1985). This extension, which is based upon the consideration of the so-called maximal controlled invariant subspace Algorithm of Isidori and coworkers (1981) and is described in full in Appendix 1, associates with any nonlinear system of the form (2), i.e. with any triplet \((f, g, h)\), a string of integers \(\{r_0, r_1, \ldots\}\) which shares much of the properties of the sequence defined by means of (9).

A (sufficient) condition for the solvability of the problem of matching the external behavior of a prescribed linear model can be found in the following way. With the given process (2), i.e. with the triplet \((f, g, h)\), we associate its structure at the infinity \(\{r_0, r_1, \ldots\}\) and the triplet \((A, B, C)\) which characterizes the model to be matched, we define an enlarged system as follows:

\[
\begin{align*}
\dot{x} &= f(x) + g(x)u \\
\dot{z} &= Ax + By \\
o &= h(x) - Cz.
\end{align*}
\]

These equations may be written in more condensed form as

\[
\begin{align*}
x &\in E = \begin{bmatrix} f & g \
\end{bmatrix} y_E \\
o &\in \mathbb{R}^n,
\end{align*}
\]

letting \(x^E = (x, z)\), \(u^E = (u, v)\) and

\[
\begin{align*}
f^E(x, z) &= f(x) \\
g^E(x, z) &= g(x) 0
\end{align*}
\]

with the triplet \((f^E, g^E, h^E)\) we associate the structure at infinity, denoted \(\{r_0, r_1, \ldots\}\).

The coincidence between the structure at infinity of the triplet \((f, g, h)\) and that of the triplet \((f^E, g^E, h^E)\) is exactly the condition we were looking for. As a matter of fact, the following result has been shown to hold.

**Theorem 2.** Let \((f, g, h)\) and \((A, B, C)\) be given. If

\[
{r_0} = {r_k}
\]

for all \(k \geq 0\), then there exists a dynamic feedback compensator under which the input-output behavior of (2) becomes of the form (7) with \(\mathcal{W}(s) = \exp(At)\).

The proof of this Theorem, which is constructive, may be found in Di Benedetto and Isidori (1986). It may be worth observing that the compensator (5) thus determined incorporates the dynamics of the model to be followed (i.e. \(z = A^*\phi(t)\)). In particular, its dimension \(\nu\) is equal to that of the linear model.

The conditions provided by this Theorem may be used, in particular, in order to check the possibility of matching the external behavior of a linear model with transfer function

\[
\mathcal{W}_A(s) = \begin{bmatrix} \text{diag}(i \frac{1}{\delta_1}) & 0 \end{bmatrix}
\]

where \(\delta\) is a suitable integer.

The external behavior of a linear system with transfer function (12) is clearly decoupled. Thus, the dynamic-state feedback compensator which enables us to match this model is such as to impose a decoupled behavior between inputs and outputs. In the closed-loop system, each component of the output is influenced only by the corresponding component of the input, independently from the specific form of the zero-input response \(\mathcal{W}(s) = \exp(At)\).

**APPLICATION OF THE MODEL MATCHING THEORY TO THE ROBOT ARM**

In this section we describe the application of the previous theoretical results to the problem of controlling a robot arm with elastic joints. In particular we will consider as an example the two-link planar robot arm whose complete mathematical model is described in Appendix 2.

By means of the Algorithm of Appendix 1 we will first compute the sequence \(\{r_0, r_1, \ldots\}\) and then see that it is possible to match a linear model with transfer function of the form (12), with \(\delta = k^* + 1\). This is done by imposing the same structure at infinity of the original system on the system (10) which includes the model to be matched.

We begin with the computation of the sequence \(\{r_0, r_1, \ldots\}\) for the triplet \((f, g, h)\) which describes the robot arm system. In the initial step we make use of the output functions \(h_1 = x_2, h_2 = x_4\) we have

\[
\{r_0\} = \{0, 0, 0, 0, 0, 0\},
\]

so that \(r_0 = 6\). Then \(r_1 = 2\), and \(h_0 = [h_{1,2}]^T\).

To start the 0-th iteration we need to find \(r_1\), i.e. we just have to compute the rank of the matrix

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

from which \(r_1 = 4\).

Since \(r_0 = 6\), the equations (A1) in Appendix 1 are trivial. To construct \(h_2\) we have to take functions from the set \(\mathcal{H} = \{L_{f_1}, L_{f_2}, L_{f_3}, L_{f_4}, L_{f_5}, L_{f_6}\}\) such that their differentials are linearly independent with respect to each other and to the set of differentials which span \(\mathcal{E}_0\). Since we have \(L_{f_1} = x_6, L_{f_2} = x_8, L_{f_3} = 0\), we obtain directly

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}
\]

and hence \(s = \dim \mathcal{E}_0 = 4\). Thus, we set

\[
h_2 = [h_{1,2,4}]^T.
\]

In the 1-st iteration, \(r_1\) is given by the rank of the matrix

\[
\begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

At this point consider the 4 by 4 permutation matrix \(P_1\) with \(P_1[0 0 1 0]\) which is such that \(P_1A_1\) selects \(r_1\) linearly independent rows from \(A_1\). In this case \(r_1 = 1\) and the third row of \(A_1\) is chosen. Define further \(h_3 = x_1 e_1^T\) and solve for an \(m\)-vector \(\mathcal{h}\) and an \(m \times m\) invertible matrix \(\mathcal{I}\) (\(m = 2\)) the equations

\[
\mathcal{I}(L \mathcal{h}_1, L \mathcal{h}_2, L \mathcal{h}_3)^T = \begin{bmatrix}
1 & 1 & 1 \\
0 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 0 \\
0 & 1 & 0 \\
0 & 1 & 0
\end{bmatrix}
\]

where for \(K\) we can choose any matrix of real numbers of rank equal to \(r_1\). We obtain:

\[
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]
\[ J_1 = 0, \quad J_2 = \frac{r_6/8_r^2}{11} \quad i = 1, 2, 22 = \frac{1}{8_r^2} \quad \text{from which we can construct a static feedback law} \]
\[ u = 3(x) + 2(y)w \quad (13) \]
that gives the new vector fields \( f = r_8, \quad g_1 = (g_3) \)
(see Appendix 2). Next, select a maximal number of functions from the set
\[ \{ L_1(y), L_2(y) \}; \quad i = 1, 2; \quad j = 1, 2 \]
with linearly independent differentials. Noting that \( L_1(y) = L_2(y) \), we have
\[ L_1(y) = 0, \quad L_2(y) = \frac{2}{8_r^2}L_3(y) = 0 \]
\[ L_1(y) = 1, \quad L_2(y) = \frac{2}{8_r^2}L_3(y) = 0 \]
so that the only candidate is \( L_3(y) \). Since \( g = 0 \), we can depend on \( x_2 \) only. We obtain in fact
\[ \alpha_2 = \beta_2 \theta \text{sp}(dL_3(y)) = \beta_2 \theta \text{sp}(0) = 0 \quad \text{O} \]
where \( (dL_3(y)) \) = \( g_1/3_L(y) \neq 0 \) assures the linear independence of the new row w.r.t. the previous ones (* denotes non relevant terms). So \( \alpha_2 = \beta_2 = 5 \) and \( \lambda_2 = [h_1, h_2, L_1(h_1), L_1(h_2), L_2(h_1), L_2(h_2)] \).

For the 2nd iteration \( A_2 = dL_2(y) \) is
\[ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + r_2 = 1 \]
Again we need to find \( (a, b) \) as a solution of a matrix equation similar to the one considered in the previous step. However, as long as \( r_2 = 1 \) we obtain the same \( (a, b) \) i.e. the same \( 1, 2 \) and so we will bypass this part of the computations; thus, the new searching set is:
\[ \lambda_2 = [L_1(y)h_1, L_1(y)h_2, L_1(y)h_3, L_2(y)h_1, L_2(y)h_2, L_2(y)h_3] \]
We have
\[ L_1(y)h_1 = 0, \quad L_2(y)h_2 = \frac{2}{8_r^2}L_3(y) = -2 \]
\[ L_1(y)h_2 = (27y/8_r^2)h_2 \]
and then
\[ \alpha_3 = \theta \text{sp}(dL_3(y)) = \theta \text{sp}(0) = 0 \quad \text{O} \]
so that \( \alpha_3 = \beta_3 = 5 \) and \( \lambda_3 = [h_1, h_2, L_1(h_1), L_1(h_2), L_2(h_1), L_2(h_2)] \).

The 3rd step of the algorithm starts computing \( A_3 = dL_3(y) \) is
\[ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + r_3 = 1 \]
Furthermore
\[ \lambda_3 = [L_1(y)h_1, L_1(y)h_2, L_1(y)h_3, L_2(y)h_1, L_2(y)h_2, L_2(y)h_3] \]
\[ \text{where} \quad L_1(y)h_1 = 0, \quad L_1(y)h_2 = 0, \quad L_2(y)h_1 = 0 \quad \text{is a function of} \quad x_1, x_2 \]
\[ x_4, x_5 \quad \text{while} \quad L_2(y)h_2 = \frac{(27y/8_r^2)h_2}{h_2} \]
Notice that \( (dL_2(y)/h_2) = g_1/3_L(y) \neq 0 \); thus, we obtain
\[ \alpha_4 = \theta \text{sp}(dL_2(y)) = \theta \text{sp}(0) = 0 \quad \text{O} \]
and \( \alpha_4 = \beta_4 = 5 \) and \( \lambda_4 = [h_1, h_2, h_1, L_1(h_1), L_1(h_2), L_2(h_1), L_2(h_2)] \).

In the 4th step, \( A_4 = dL_4(y) \) is
\[ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + r_4 = 1 \]
Furthermore
\[ \lambda_4 = [L_1(y)h_1, L_1(y)h_2, L_1(y)h_3, L_2(y)h_1, L_2(y)h_2, L_2(y)h_3] \]
\[ \text{where} \quad L_1(y)h_1 = 0, \quad L_1(y)h_2 = 0, \quad L_2(y)h_1 = 0 \quad \text{is a function of} \quad x_1, x_2 \]
\[ x_4, x_5 \quad \text{while} \quad L_2(y)h_2 = \frac{(27y/8_r^2)h_2}{h_2} \]
Notice that \( (dL_2(y)/h_2) = g_1/3_L(y) \neq 0 \); thus, we obtain
\[ \alpha_5 = \theta \text{sp}(dL_2(y)) = \theta \text{sp}(0) = 0 \quad \text{O} \]
and \( \alpha_5 = \beta_5 = 5 \) and \( \lambda_5 = [h_1, h_2, h_1, L_1(h_1), L_1(h_2), L_2(h_1), L_2(h_2)] \).

Thus, \( \alpha_5 = \beta_5 = 5 \) and \( \lambda_5 = [h_1, h_2, h_1, L_1(h_1), L_1(h_2), L_2(h_1), L_2(h_2)] \).

which gives \( r_5 = 2 \).

Notice that, for the particular example considered, we have found that the set of functions whose differentials span \( \lambda_5 \) is obtained by taking the Lie derivatives of the system outputs only with respect to the vector field \( f \); these functions may be used to define a new set of local coordinates in terms of which the system is described by much simpler expressions.

To conclude this section, we consider the problem of finding what kind of linear and decoupled model it is possible to match. Let the linear model be expressed in the parameterized form (8), with the condition that \( c_A b_2 = 0 \), if \( i \neq j \), i.e. having a decoupled structure of input-output channels. The model matching problem is solvable if the enlarged system (robot arm plus model, see (10)) has the same structure at the infinity as the original system (the robot arm alone).

The procedure is then the following: apply the Algorithm to (\( f, g, h \)) and compute the matrices \( A_k \) of the enlarged system; then, imposing the equality between \( r_k = \text{rank of} \ A_k \) and \( r_k \) for \( k = 1, \ldots, 5 \), derive conditions on the triplet \( (A, B, C) \). The computation of Lie derivatives is greatly simplified being the model to be matched a linear one and will be omitted here. After \# = 5 steps of the Algorithm, we find that the constraint
\[ C^T A = 0 \quad \text{for} \quad k = 0, 1, \ldots, 4 \]
is such as to make the condition (11) satisfied for all \( k \geq 0 \).

This means that the input-output behavior of the considered robot arm can be made equal via dynamic feedback to that of a decoupled linear system constituted by two chains of six integrators each. In this case (12) becomes \( \text{rank of} \ A_k = 5 \), and, then, a dynamic controller can be constructed just following the arguments of the proof of Theorem 2. The obtained controller will be of order \( \# + 1 = 6 \), with a dynamics inherited from the one of the matched model. However, the explicit derivation of this controller will be avoided here.

**ANALYSIS OF THE MODEL STRUCTURE VIA SYSTEM GRAPH**

In order to explore the decoupling problem and the possibility of reducing the dynamic order of the controller we recall the notion of graph representation of a system, introduced by Slijak (1977).

The system graph of nonlinear system (2) is a weighted oriented graph \( G(S, L) \), where \( S \) is the set of nodes representing the input, state and output variables and \( L \) is the set of weighted arcs representing the influences among variables.

More precisely, the weight of an arc \( (x_j, x_k) \) is given by \( g_{jk} \) where \( g_{jk} \) is the \( k \)-th element of \( g_j \); the weight of an arc \( (s_j, y_j) \) is given by \( h_j/s_j \), where \( h_j \) is the...
Consider now, in terms of system graph, the effects of adding chains of integrators to the inputs of the system. Connect one integrator to input $u_i$, i.e., $u_i = \dot{y}_i$. Modifications occur in the graph only at arcs ending in state nodes directly connected with $u_i$. More precisely, in the new system graph the weight of an arc $(x_i, y_i)$, with $x_i$ connected to $u_i$, becomes $3f(x_i) + g(x_i)$, $g(x_i)$. New arcs are created if $3f(x_i) + g(x_i) = 0$ but $g(x_i) \neq 0$. As a matter of fact, the addition of an integrator modifies consistently the structure of the graph.

At this point one possibility in order to modify the graph structure is to apply first a feedback transformation, which use the results of the previous section, and then to work on the obtained graph with the above dynamic extension.

**Decoupling Dynamic Controller**

We saw that during each step of the algorithm a particular static feedback is computed. Whenever the system is statically decoupled, the feedback obtained at the last step is a decoupling one. In our case (see (13)) we found:

$$\begin{align*}
\hat{u} = \hat{x} &= \lambda(x) + \hat{y}(x) = \\
&= \left[ \begin{array}{c}
0 \\
-16 \\
0 \\
-18
\end{array} \right] \cdot \hat{x} + \left[ \begin{array}{c}
0 \\
1 \\
0 \\
1
\end{array} \right] \cdot \hat{y}, \quad (14)
\end{align*}$$

The application of this feedback law to the robot obviously does not achieve a decoupled structure.

However, in the system

$$\begin{align*}
\dot{x} &= f(x)g(x) + \hat{g}(x)\hat{y}(x) = \hat{y}(x)
\end{align*}$$

and the vector fields $\hat{y}_i(u)$ are given respectively by:

$$\begin{align*}
\hat{y}_1(u) &= \left[ \begin{array}{c}
-16 \\
0 \\
0 \\
0
\end{array} \right], \\
\hat{y}_2(u) &= \left[ \begin{array}{c}
0 \\
1 \\
0 \\
0
\end{array} \right],
\end{align*}$$

and the vector fields $\hat{y}_i(u)$ are given respectively by:

$$\begin{align*}
\hat{y}_1(u) &= \left[ \begin{array}{c}
-16 \\
0 \\
0 \\
0
\end{array} \right], \\
\hat{y}_2(u) &= \left[ \begin{array}{c}
0 \\
1 \\
0 \\
0
\end{array} \right],
\end{align*}$$

where $D = \begin{bmatrix} D_1 & D_2 \end{bmatrix}$ is the desired output feedback of the system obtained in the previous section.

Applying (12) and using the explicit expressions of the terms involved (see Appendix 2) we obtain:

$$\begin{align*}
\hat{x}(u) &= \hat{A}(\hat{x}) + \hat{B}(\hat{y}),
\end{align*}$$

where $D = \begin{bmatrix} D_1 & D_2 \end{bmatrix}$ is the desired output feedback of the system obtained in the previous section.

Applying (12) and using the explicit expressions of the terms involved (see Appendix 2) we obtain:

$$\begin{align*}
\hat{x}(u) &= \hat{A}(\hat{x}) + \hat{B}(\hat{y}),
\end{align*}$$

where $D = \begin{bmatrix} D_1 & D_2 \end{bmatrix}$ is the desired output feedback of the system obtained in the previous section.

Applying (12) and using the explicit expressions of the terms involved (see Appendix 2) we obtain:

$$\begin{align*}
\hat{x}(u) &= \hat{A}(\hat{x}) + \hat{B}(\hat{y}),
\end{align*}$$

where $D = \begin{bmatrix} D_1 & D_2 \end{bmatrix}$ is the desired output feedback of the system obtained in the previous section.
The order of the controller has been reduced from twelve to four by the joint application of static pre-feedback (19), dynamic extension (16) and static decoupling feedback (18) from the extended state. The proposed dynamic controller exhibits a further nice property. Static feedback decoupling creates a closed-loop system which has an unobservable part with a possibly nonlinear dynamics of dimension $n = \frac{n}{2} - 1$, see Isidori and co-workers (1981); the remaining part of the system is equivalent to a linear controllable and observable subsystem. In our case we have $\frac{\delta n}{2} = 3$, i.e. two chains of six integrators, $n = \dim \mathbb{R} = 12$ and hence $\dim n = 0$, so that the decoupling law is also a linearizing one for the extended system. Thus, the composition of the control law (19) with the robot arm equations (2) yields a dynamical system whose external behavior as decoupled and which is diffeomorphic to a linear and controllable system. As a matter of fact, in the coordinates

$$
\begin{align*}
&z_1, ..., z_{23}, \quad z_2 = h_1, \quad z_3 = L f h_2, \quad z_4 = L f h_3, \\
&z_5, ..., z_9 = L f h_7, \quad z_{10} = L f h_8, \quad z_{11} = L f h_9, \quad z_{12} = L f h_{10}, \\
&z_{13}, ..., z_{22} = L f h_{11}, \quad z_{23}, ..., z_{52} = L f h_{12}, \\
&z_{53}, ..., z_{54} = L f h_{13}
\end{align*}
$$

the closed-loop system is described by the equations

$$
\dot{x} = A_1 x + B_1 u, \quad y = c_1 x + c_2
$$

with the triple $\{A_1, B_1, C_1\}$, $i = 1, 2$, in Brunovsky canonical form.

We conclude that in the examined elastic robot we obtain, as a byproduct of the decoupling, the full state linearization. This allows to assign all the dynamic behavior by standard techniques.

CONCLUSIONS

In this paper we have shown how nonlinear model matching theory can be applied for the dynamic decoupling control of industrial robots with joint elasticity. The existence of a decoupling controller is guaranteed for the planar two-link robot with elastic joints. The model matching approach leads to a twelve-order dynamic controller; the resulting closed-loop system matches the input-output behavior of a prescribed decoupled linear system but includes an unobservable part with a possibly nonlinear dynamics. However, the analysis of the robot model structure allows both to reduce the order of the controller down to four and to fully linearize the closed-loop state dynamics.

We note also that similar results may be obtained by means of an algorithm for dynamic feedback decoupling recently proposed by Dussacce and Mogg (1985).

REFERENCES


APPENDIX 1

In this Appendix we describe the algorithm for the computation of the structure of the structure at the infinity of a nonlinear system (Krener, 1985). We recall that if $\frac{\delta n}{2}$ is a smooth function, its differential $\psi$ is the $1 \times n$ row vector with $j$th component given by $\psi_j(x) = \frac{\partial}{\partial x_j} \psi$.

Let $\Psi(x)$ be a matrix $\Psi(x)$ of the map $x$ one construct first of all the $n \times \text{vec}$ (subspace of row vectors)

$$
\Psi(x) = \text{span} \{ \psi_1(x), \ldots, \psi_n(x) \}.
$$

Suppose $\psi_i(x)$ has dimension $\frac{\delta n}{2} = i$ in a neighborhood of a point $x_0$. Then there exists an $n \times n$ column vector $\gamma$, whose entries $\gamma_{1i}, \ldots, \gamma_{ni}$ are entries of $\gamma$, with the properties that the differentials $\psi_i(x) = \gamma_i(x)$ are linearly independent at all $x$ in a neighborhood of $x_0$. The algorithm consists of a finite number of iterations, each one defined as follows.

Iteration $k$. Consider the $n \times n$ matrix $A_k(x)$ whose $(i,j)$-entry is $d_{ij}(x)$; $\psi_i(x)$. Suppose that in a neighborhood of $x_0$ the rank of $A_k(x)$ is constant and equal to $r_k$. Then it is possible to find $r_k$ rows of $A_k(x)$ which, for all $x$ in a neighborhood of $x_0$, are linearly independent. Let $\beta_k = \{ \beta_{k1}, \ldots, \beta_{kr} \}$ with $\beta_{kj} \in \mathbb{R}$.
be an \( n \times n \) permutation matrix, such that the \( r_k \) rows of \( P_{k+1} A_k(x) \) are linearly independent. Let \( B_k(x) \) be an \( n \times n \) vector whose i-th element is \( d A_{k+1}(x)f(x) \). As a consequence of the assumptions on \( P_{k+1} \), the equations
\[
B_{k+1}A_k(x)u(x) = -P_{k+1}B_k(x)
\]
(Al)
where \( K \) is a matrix of real numbers, of rank \( r_k \) may be solved for \( u \) and \( B \), an \( m \)-vector and an \( m \times m \) invertible matrix whose entries are real-valued smooth functions defined in a neighborhood of \( x^0 \).

Consider the set of functions \( \{ dA_k(x) \} \) spanned by the first \( r_k \) derivatives of \( A_k(x) \) for all \( k \) such that the differentials \( dA_k(x) \) are linearly independent at all \( x \) in a neighborhood of \( x^0 \). Define the \( r_k \)-vector \( \lambda_{k+1}(x) \) whose i-th entry is the function \( \lambda_{k+1,i} \). This concludes the \( k \)-th iteration. We repeat the process with the \( k+1 \)th set of functions until \( r_k \) is reached.

The set \( \{ dA_k(x) \} \) provides the so-called structured infinity associated with the triple \((f,g,h)\).

**APPENDIX 2**

We report here the dynamic model of a two-link robot arm with joint elasticity, whose possible configurations lie in a vertical plane:

\[ x = f(x) = \begin{bmatrix} x_2 \\ x_3 \\ x_4 \end{bmatrix}, \quad y = h(x) = \begin{bmatrix} x_2 \\ x_3 \\ x_4 \end{bmatrix} \]

with
\[
f(x) = \begin{bmatrix} x_5 & x_6 & x_7 & x_8 \end{bmatrix}^T, \quad g(x) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}^T
\]

where
\[
A_{51} = 0, \quad A_{52} = \begin{bmatrix} A_1 \cos^2 x_4 - A_2 \sin^2 x_4 \end{bmatrix}, \quad A_{53} = A_1^2 / \sqrt{m^2 - 1}, \quad A_{54} = A_2^2 / \sqrt{m^2 - 1}, \quad A_{55} = 0, \quad A_{56} = -A_1 \cos x_4, \quad A_{57} = A_2 \sin x_4, \quad A_{58} = 0
\]

and the two \((n \times n)\) subspaces (of row vectors)
\[
\lambda_k(x) = \text{span} \{d\lambda_1(x), \ldots, d\lambda_{k-1}(x)\}, \quad \lambda_k(x) = \text{span} \{d\lambda_1(x), \ldots, d\lambda_k(x)\}
\]

\( \lambda_k(x) \) has constant dimension \( r_k \), for all \( k \neq k^* \). The sequence \( \{ r_0, r_1, \ldots \} \) provides the so-called structure at the joint \( i \). Since \( r_k \leq k \), \( \lambda_k(x) \) is the reduction ratio of the gear box, \( K_i \) is the elastic constant, \( J_{RZ_2} \) is the inertia of the rotor. The constants \( A_1, A_2, \ldots, A_m \) and \( G_i \) include all the data of the robot (mass and inertia of links and rotors, length of links and their mass center).

Finally, we collect here the expressions of the vector fields \( f, g, h \) computed with the Algorithm:
\[
f(x) = \begin{bmatrix} f_1 \end{bmatrix} = \begin{bmatrix} x_5 \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}^T
\]

We note explicitly that \( f_0 = 0 \) and that \( f_k \) is a function of \( x_k \) and \( x_k \) only, this is reflected in the system graph of Fig. 2.