**B. Brogliato**
**D. Rey**
**A. Pastore**
**J. Barnier**
Laboratoire d'Automatique de Grenoble
Domaine Universitaire
38402 Saint-Martin d'Hères, France

# Experimental Comparison of Nonlinear Controllers for Flexible Joint Manipulators

## Abstract

*In this paper, the authors present an experimental comparison of several nonlinear controllers for flexible joint manipulators. The experimental setup is a 2-DOF manipulator with an elasticity at the second joint. This work follows and confirms a previous comparative study based on theoretical and numerical investigations. The main conclusions of the study are as follows. (1) Some nonlinear controllers may behave quite differently when applied to a particular device, despite their apparent theoretical resemblance. (2) The compensation of nonlinearities (Coriolis and centrifugal torques), as well as joint flexibilities, may improve the closed-loop performance for certain desired trajectories, despite a rather high joint stiffness. (3) However, when the joint stiffness is sufficiently large, the controllers that do not take into account the joint flexibility in the design can, in certain cases, behave as well as those that incorporate it. Thus, future experimental work with a smaller joint stiffness is needed. (4) Finally, the complex structure of certain nonlinear controllers for flexible joint robots (backstepping and energy-shaping schemes) may be an obstacle to their closed-loop behavior enhancement. The experimental results clearly show that the inputs of such schemes chatter and have a magnitude larger than those of simpler controllers (like singular perturbation-based or Slotine and Li controllers).*

## 1. Introduction

The last 10 years have witnessed significant theoretical progress in the control of flexible joint manipulators (see Spong 1990 for a survey). They were motivated by practical problems encountered, for instance, in industrial robots equipped with harmonic drives, that may decrease the tracking performance or even sometimes destabilize the closed-loop system. Several directions have been followed for control, among them singular perturbation techniques (the stability results then require a high enough stiffness value at the joints) (Spong 1987, 1989, 1995; Ghorbel, Hung, and Spong 1989; Ghorbel, Fitzmorris, and Spong 1990; Ghorbel and Spong 1992) and nonlinear global tracking controllers derived from new design tools such as the backstepping (Kanellakopoulos, Kokotovic, and Morse 1991) or the energy-shaping (Ortega and Espinosa 1993) techniques (see Lozano and Brogliato 1992; Brogliato, Ortega, and Lozano 1995).

The experimental comparison presented in this paper concerns seven different schemes. Four of them belong to the last class of algorithms, along with a simple PD controller, the classical Slotine and Li algorithm (designed as if the system were perfectly rigid), and a controller based on a singular perturbation analysis. The study that follows does not aim to classify several design techniques from a general point of view. Indeed, it is clear that the results depend on the nature of the process to be controlled, and that they might significantly vary if applied to another system (e.g., the integrator backstepping method has provided good results for control of a brush DC motor with a load; Dawson, Caroll, and Schneider 1994). The results show that, within the class of such mechanical systems, some design techniques that explicitly incorporate physical properties of the system, such as passivity, may provide better results. Moreover, reliability of the experimental conclusions is increased by the fact that previous theoretical and simulations results (the work in this paper completes a previous theoretical and numerical investigation in Brogliato, Ortega, and Lozano 1995) predicted rather well the obtained behaviors of the real closed-loop plant.

This paper is organized as follows. In Section 2, we briefly recall the general plant model and the design techniques that

The International Journal of Robotics Research,
Vol. 17, No. 3, March 1998, pp. 260–281,

have been applied. Section 3 presents the experimental setup. In Section 4, the obtained results are presented and discussed. Conclusions are given in Section 5. Some calculations are reported in the appendixes.

## 2. Plant Models and Controller Design

Two models for flexible joint manipulators have been used in the related literature. The first (and simplest) was introduced in Spong (1987):

$$\begin{aligned} D(q_1)\ddot{q}_1 + C(q_1,\dot{q}_1)\dot{q}_1 + g(q_1) &= K(q_2 - q_1) \\ J\ddot{q}_2 &= K(q_1 - q_2) + u, \end{aligned} \tag{1}$$

where $D(\cdot)$ is the inertia matrix, $C(\cdot,\cdot)$ is related to Coriolis and centrifugal efforts, $g(\cdot)$ is the generalized gravity vector, $K$ is the constant diagonal joint stiffness matrix, $q_1$ are the link angles, $q_2$ are the motor shaft angles, $J$ is the actuator inertia matrix, and $u$ is the control input. This model possesses some special properties (such as state feedback linearizability) but is in general a simplified form of a more general model (Nicosia and Tomei 1992) in which the effects of the links' angular velocity on the kinetic energy of the motor shafts have been neglected. It is worth noting that when the actuators are mounted at the fixed base of the robot—that is, for parallel drive manipulators (Yoshikawa 1990)—this model is the exact one. This is the case of our experimental 2-DOF manipulator.

Several controllers have been presented in Brogliato, Ortega, and Lozano (1995) that guarantee global tracking of any smooth desired trajectory. They require state feedback (i.e., measurement of $q_1$, $\dot{q}_1$, $q_2$, and $\dot{q}_2$). Although the perfect knowledge of the inertia parameters can be relaxed via adaptive techniques (Lozano and Brogliato 1992), only the fixed parameter control laws have been tested in the present work. This is mainly due to the fact that, first, adaptive techniques are essentially useful when the manipulator must carry loads of unknown mass and at the same time track some desired trajectory. The experiments conducted in this paper did not belong to such a robotic tasks class. Second, although adaptive control of flexible joint manipulators has been a challenging theoretical work, and has been solved in Khorasani (1992), Lozano and Brogliato (1995), Spong (1995), and Brogliato and Lozano (1996), its real practical usefulness is not really clear, and the obtained controllers may be too complicated to be easily implemented (e.g., several parametric projections are used in Lozano and Brogliato [1992], whose implementation does not seem trivial at all), or the results may be restricted to some sort of local ultimate boundedness of the state (Khorasani 1992; Spong 1995), and are therefore of limited applicability. Nevertheless, some experiments have been presented in Ghorbel, Hung, and Spong (1989) and Ghorbel, Fitzmorris, and Spong (1990).

Apart from the PD and the Slotine and Li controllers, the fixed parameter controllers tested in this paper originated from the following well-known design techniques:

- the application of singular perturbation analysis to flexible joint manipulators when the joint stiffness is high enough (Spong 1987, 1989);
- a result in Seibert and Suarez (1990) on stabilization of cascaded nonlinear systems for controller 1;
- the backstepping method (Kanellakopoulos, Kokotovic, and Morse 1991) for controllers 2 and 3; and
- the energy-shaping method that originated in Takegaki and Arimoto (1981) and was later extended in Slotine and Li (1988) for rigid robots, Lozano and Brogliato (1992) for flexible joint robots, and formalized in Ortega and Espinosa (1993) for controller 4.

For the sake of briefness, the theoretical aspects of the proposed controllers are not recalled. The interested readers are referred to the cited papers for details. Recall that the four nonlinear controllers for flexible joint manipulators which are presented in (Brogliato, Ortega, and Lozano 1995) can be written shortly as follows:

$$\text{Controller 1} \begin{cases} u = J\ddot{q}_{2d} - K_1\tilde{q}_2 - K_2\dot{\tilde{q}}_2 + K(q_2 - q_1) \\ q_{2d} = K^{-1}u_R + q_1 \end{cases} \tag{2}$$

$$\text{Controller 2} \begin{cases} u = J[\ddot{q}_{2d} - 2\dot{\tilde{q}}_2 - 2\tilde{q}_2 - K(\dot{s}_1 + s_1)] \\ \quad + K(q_2 - q_1) \\ q_{2d} = K^{-1}u_R + q_1 \end{cases} \tag{3}$$

$$\text{Controller 3} \begin{cases} u = J[\ddot{q}_{2d} - 2\dot{\tilde{q}}_2 - 2\tilde{q}_2 - (\dot{s}_1 + s_1)] \\ \quad + K(q_2 - q_1) \\ q_{2d} = K^{-1}u_R + q_1 \end{cases} \tag{4}$$

$$\text{Controller 4} \begin{cases} u = J\ddot{q}_{2r} + K(q_{2d} - q_{1d}) - B_2 s_2 \\ q_{2d} = K^{-1}u_R + q_{1d}, \end{cases} \tag{5}$$

where $u_R = D(q_1)\ddot{q}_{1r} + C(q_1,\dot{q}_1)\dot{q}_{1r} + g(q_1) - B_1 s_1$ is the Slotine and Li control law for the rigid part of the model. $\dot{q}_{1r} = \dot{q}_{1d} - \Lambda_1\tilde{q}_1$, $s_1 = \dot{\tilde{q}}_1 + \Lambda_1\tilde{q}_1$ are the classical signals used in the design of this controller (the same definitions apply with subscript 2). $B_1$, $B_2$, $K_1$, $K_2$ are positive definite constant gains. The subscript $d$ is for the smooth desired trajectories. Note that controllers 2 and 3 do not involve acceleration measurements, although the equivalent forms in (3) and (4) contain $\dot{s}_1$. Also, controller 1 (see (45) and (46)) decouples the response of $\dot{q}_2$ so that the rigid-body dynamics, driven by $q_2$, are the zero dynamics.

The last controller is in fact an improved version (in the sense that it is a static state feedback) of the dynamic state feedback proposed in Lozano and Brogliato (1992) that can

be written as

$$\begin{cases} u = J\ddot{q}_{2r} - K[q_{1d} - q_{2d} - \int_0^t (\Lambda_1 \tilde{q}_1 - \Lambda_2 \tilde{q}_2) d\tau] \\ \quad -B_2 s_2 \\ q_{2d} = p[pI + \Lambda_2]^{-1}\Big\{K^{-1}u_R + q_{1d} + K(\tilde{q}_1(0) \\ \quad - \tilde{q}_2(0)) - \int_0^t (\Lambda_1 \tilde{q}_1 - \Lambda_2 q_2) d\tau\Big\}. \end{cases} \tag{6}$$

This controller has not been considered in the experiments because it is logically expected not to provide better results than its simplified counterpart. It is more complex but is based on the same idea.

In the appendixes, the application of these design techniques is developed for the Capri (Commande Avancée de PRéhenseur Intelligent) robot, whose model merges both rigid and flexible models, since the first joint is assumed perfectly rigid. Originally, the backstepping method applies to certain classes of triangular systems (see Kanellakopoulos, Kokotovic, and Morse 1991) in which one constructs the controller recursively step by step and then proves stability with a suitable Lyapunov function of a new transformed state vector. Then, convergence and boundedness of the original state follow. On the contrary, for the energy-shaping method, one chooses a positive definite function and then designs the controller such that this function is a Lyapunov function for the closed-loop system. The method in Seibert and Suarez (1990) consists of stabilizing the first subsystem of a cascade system. Then, if the trajectories of the complete system are bounded, asymptotic convergence is assured. From these points of view, the methods are clearly separated. It might be argued that, in fact, they all are different versions of the backstepping method, in which the auxiliary inputs designed at each step can be modified, as well as the new state vector and the final Lyapunov function. (This is in fact illustrated by our two backstepping controllers. In the second version—that is, controller 3—the joint stiffness $K$ no longer appears before $\dot{s}_1 + s_1$ in the right-hand side of the $u$ equation. This modification is expected to decrease significantly the input magnitude when $K$ is large. The experimental results will demonstrate that this is indeed the case.) As noted in Brogliato, Ortega, and Lozano (1995), it does not seem reasonable to adopt such a viewpoint without significantly modifying the original motivations of the respective methods. We thus prefer to keep the differences between them. But we agree this may be a subject of dispute between the respective "inventors." However, the major goal of this work is to illustrate that several controllers, which seem very close to each other either by their design approach or by their final form, can yield very different results when applied to a real process. In Brogliato, Ortega, and Lozano (1995), these controllers have been discussed from several points of view. Most important, it was shown that when the joint stiffness grows unbounded (i.e., the rigid manipulator model is retrieved), the only controller that converges to the rigid control law of Slotine and Li (1988) is controller 4. It can thus be concluded that the extension of the rigid case to the flexible joint case cannot be stated for the other three control laws. We believe that this elegant physical property plays a major role in the closed-loop behavior of the plant. However, it is clear that we have no rigorous proof of this claim. Similarly, the idea that passive schemes are more robust than those that do not possess this property is not, to the best of our knowledge, proved anywhere. Note that the backstepping schemes presented here *do* possess some closed-loop passivity properties, but they are related to transformed coordinates (see Brogliato, Ortega, and Lozano 1995). On the contrary, the passivity-based schemes possess this property in the original generalized coordinates $\tilde{q}$. In this sense, they are closer to the physical system than are the other schemes. This might constitute an intuitive explanation of still more intuitive results.

## 3. The 2-DOF Capri Robot

In this section, we present our experimental device in detail. We shall focus on two points: the mechanical structure and the real-time computer connected to the process. Note that this manipulator has been entirely designed and built in the Laboratoire d'Automatique de Grenoble to provide a convenient way to test motion and force controllers recently developed.

### *3.1. Mechanical Structure*

The Capri robot is a planar mechanism constituted by two links of respective lengths 0.16 m and 0.27 m, connected by two hubs as sketched in Figure 1. The first link is an aluminum AU4G, U-frame to improve stiffness with respect to the forearm, which can be designed as less rigid. The second link has a more peculiar structure because it supports the applied forces. It is designed as a pipe of diameter 0.05 m, and it is equipped with force piezoelectric sensors. The force magnitude, point of application, and orientation can be measured and calculated. The sides of the forearm with Kistler quartz load washers can measure extension and compression forces, and the half-spherical extremity possesses a Kistler 3 components force transducer (only two of them are used) from which it is possible to calculate the magnitude and the orientation of the applied force. In the present paper, these force measurement devices are not needed, since we are concerned with motion control only.

The robot arm is actuated by two DC motors located at the underside of the basement table. They are coupled to the links by reducers (gears and notched belts), each with a ratio of 1:50. The first motor (Infranor MX 10) delivers a continuous torque of 30 N.cm and a peak torque of 220 N.cm for a total weight of 0.85 kg. The second motor (Movinor MR 08) provides a continuous torque of 19 N.cm and a peak torque

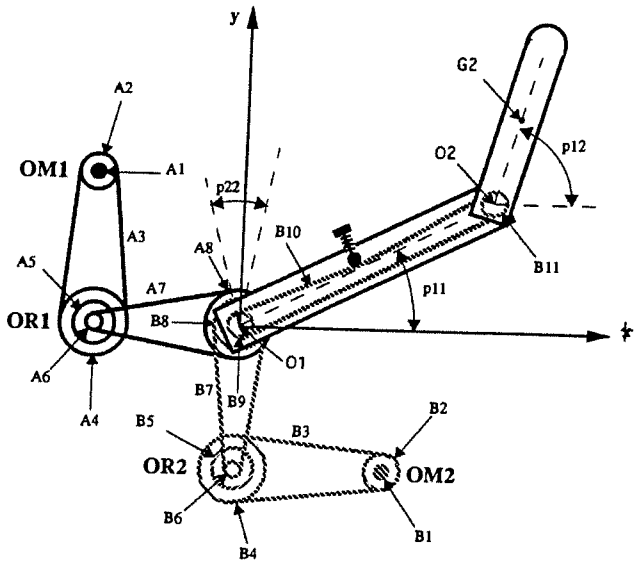


Fig. 1. Capri robot, kinematic scheme.

of 200 N.cm for a weight of 0.65 kg. To increase speed, the drive arrangement is such that the weight is not boarded on the links. Both motors are equipped with a 500 pulses/turn incremental encoder and a DC tachometer making joint position $q_1$ and velocity $\dot{q}_1$ available for feedback. The position $q_2$ is measured by a potentiometer mounted on the last link. In the experiments, the velocity $\dot{q}_2$ has been obtained by differentiating the position signal (a filtering action has been incorporated to decrease the measurement noise effects).

The effective working area of the robot arm is bounded by sensors. An inductive sensor prevents the first arm from doing more than one turn; that is, $q_{11} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. Two microswitches prevent the second arm from overlapping on the first one. They both inhibit the inverters (Infranor MSM 1207) controlling the DC motors.

The values that have been useful for the calculation of the inertial parameters are given in the appendixes.

REMARK 1. The Capri robot has been modeled as a parallel drive rigid-link robot, with the second joint being elastic. It is clear that such a model is only a crude approximation of the real device. Some approximations may be quite justified, like the rigidity of the first joint and of the links. Some others are much more inaccurate.

i. The belt that couples the second actuator and the second joint is modeled as a spring with constant stiffness, which means that only the first mode of its dynamic response is considered.
ii. There are significant clearances in the mechanical transmission (especially at the joints, due to the belts and the pulleys) and a serious amount of dry friction.
iii. The frequency inverters that deliver the current to the motors possess a nonsymmetric dead zone. Therefore, different amounts of current are necessary to start motion in one direction or the other.
iv. As noted above, the value of $\dot{q}_1$ used in the algorithm and obtained by differentiating a potentiometer signal is noisy and has to be filtered.
v. The inertial parameters have been calculated (see Appendix A) by simply measuring and weighing the mechanical elements of the arms. The second joint stiffness has been measured statically off-line. Its value has been found to be 50 Nm/rad, and has been used in the experiments.
vi. Some saturation on the actuators' currents has been imposed by software for obvious safety reasons. Since nothing a priori guarantees stability when the inputs are saturated, the feedback gains have to be chosen so that the control input remains inside these limits.

Some of these approximations stem from the real device and cannot be avoided (points (i), (ii), and (iii)) without modifying the mechanical or electrical structure. The uncertainty in (iv) could perhaps be avoided via the use of observers, but this would drastically modify the control algorithms in structure and complexity, and this is clearly outside the scope of this work. The saturations (vi) are necessary to protect the motors, and have been chosen in accordance with the manufacturer recommendations and our own experience on their natural robustness. The crude identification procedure in (v) has been judged sufficient because the aim of the work was not to make a controller perform as well as possible in view of an application, but rather to compare several controllers. In view of this, the most important fact is that they all be

tested with the same (acceptable) parameters values; that is, if one controller is proven to behave correctly with these sets of parameters, do the others behave as well?

### 3.2. *Real-Time Computer*

A real-time computer is connected to this process. It consists of a set of DSpace boards and a host PC. The PC is an HP Vectra running at 66 MHz with 8 Mo of RAM and a hard disk of 240 Mo. The DSpace system consists of the following:

- a DS 1002 floating-point processor board built around the Texas Instruments TMS/320C30 digital signal processor. This processor allows 32 bits floating point computation at 33 MFlops. A static memory of 128 K words of 32 bits is available on this board. A 2 K words dual-port RAM is used simultaneously by the host PC and the DSP;
- a DS 2002 multichannel ADC board with 2 A/D 16 bits resolution converters (5 $\mu$s conversion time) and a 16-channel multiplexer for each converter;
- a DS 2001 D/A converter board comprising five parallel analog output channels with 12 bits DAC (3 $\mu$s conversion time);
- a DS 3001 incremental encoder board with five parallel input channels. A fourfold pulse multiplication, a digital noise filter, and a 24 bits width counter are used for each channel; and
- a DS 4001 digitak I/O and timer board with 32 digital I/O lines configurable as inputs or outputs in groups of eight lines.

All these boards are attached together by the 32 bits PHS-Bus at a 16 MB/s transfer speed. They are located in a separate rack connected to the host PC by a cable between two adaptation boards.

The PC is used for developments and supervision of the application. Several softwares are available for the DSpace system:

- SED30 and MON30 are used to configure the hardware.
- C30 is the Texas Instruments compiler for the TMS320C30.
- TRACE30W is a graphical real-time software that permits the display of the selected variables of the application.

The application itself is made of two parts: the control algorithm running on the DSP, sampled at 1 ms in our case, and the dialogue interface running on the PC that allows the operator to supervise the execution of the control through the dual port memory.

To guarantee the repeatability of the experiments, there is an initialization procedure that is activated each time the origins have been lost, or at the beginning of the experiment.

## 4. Experimental Results

In this section, we present the experimental results obtained by implementing seven different schemes. To make the study complete, a PD controller of the form $u = -K_p\tilde{q}_2 - K_d\dot{\tilde{q}}_2$ (in the following, $\tilde{q} = q - q_d$ for any function $q(t)$), the Slotine and Li (SLI) scheme, and a controller based on a singular perturbation analysis (SPB), which consists of the Slotine and Li scheme using link variables $q_1$, $\dot{q}_1$ plus a correction term of the form $K_v(\dot{q}_1 - \dot{q}_2)$ (Spong 1989), have been implemented together with the four nonlinear controllers described above (denoted as controllers 1, 2, 3, and 4). This allows us to clearly dissociate the effects of the nonlinearities (the reference trajectories have been chosen fast enough so that Coriolis and centrifugal effects are effective) from the effects of the flexibility (once the rigid controllers are implemented, one can see how the flexible ones improve the closed-loop behavior, if they do). Moreover, it enables us to compare the results obtained from a very simple way of compensating for the flexible effects (SPB scheme) to more complex controllers.

The experiments have been performed as follows. For each controller, three different desired trajectories have been tested. They have been chosen as follows (see Fig. 1 and the appendixes for the definition of the angles $p_{11}$, $p_{12}$, $p_{22}$):

Desired trajectory 1

$$\begin{aligned} p_{12d} &= 0.8\sin(ft) \\ p_{11d} &= -p_{12d} \end{aligned} \tag{7}$$

Desired trajectory 2

$$\begin{aligned} p_{12d} &= 0.4\sin(2ft) \\ p_{11d} &= 0.8\sin(ft) \end{aligned} \tag{8}$$

Desired trajectory 3

$$\begin{aligned} p_{12d} &= \frac{b^5}{(s+b)^5}[g(t)] \\ p_{11d} &= -p_{12d}, \end{aligned} \tag{9}$$

where $f = \omega(1 - \exp(-at))^4$, $a = 14$, $\omega = 9$ rad/s, and $g(t)$ is a square function with magnitude 0.8 rad, period 5 s, and $b = 30$. $s$ in (9) is the Laplace transform variable. The desired trajectories have been chosen to be smooth enough as required by the theoretical developments. However, they are fast enough to render the nonlinearities and flexibility effects significant. Concerning the PD and the Slotine and Li algorithms, one has to replace $p_{12d}$ by $p_{22d}$, since the system is assumed rigid (hence, one uses the motor variables in the control loop). Also, recall that $p_{11} = p_{21}$, since the first joint is assumed to be rigid.

These three desired trajectories, sufficiently different from one another, have been chosen to allow one to make a conclusion about the capability of adaptation of the controllers

**Table 1.**

| *controller* | *err1* (traj.1) | *err2* (traj.1) | *err1* (traj.2) | *err2* (traj.2) |
|---|---|---|---|---|
| PD | 0.346 | 84.5 | 1.4 (1.6) | 360 (1000) |
| SLI | 0.11 | 37.9 | 0.02 (0.034) | 40 (51) |
| ~~SPB~~ | ~~0.23~~ | ~~12~~ | ~~0.3 (0.3)~~ | ~~1300 (1300)~~ |
| ~~Controller 1~~ | ~~0.6~~ | ~~9~~ | ~~0.5 (0.5)~~ | ~~160 (160)~~ |
| Controller 2 | x | x | x | x |
| Controller 3 | 0.34 | 12 | 0.3 | 75 (173) |
| Controller 4 | 0.64 | 9 | 0.224 (0.6) | 70 (150) |

**Table 2.**

| *controller* | *err1* (traj.3) | *err2* (traj.3) |
|---|---|---|
| PD | 0.3 (0.3) | 50 (50) |
| SLI | 0.055 (0.055) | 30 (30) |
| ~~SPB~~ | ~~0.04 (0.04)~~ | ~~12.5 (12.5)~~ |
| ~~Controller 1~~ | ~~0.19 (x)~~ | ~~93 (x)~~ |
| Controller 2 | x | x |
| Controller 3 | 0.135 (0.135) | 30 (30) |
| Controller 4 | 0.19 (0.19) | 15 (15) |

The x indicates that we have not been able to stabilize the system for controller 2. Concerning controller 1 in Table 2, the system was not stable with the gains used for trajectory 1 when trajectory 3 was tested.

to a modification of desired tracking performance. Indeed, we have retained several criteria of performance for the controllers:

- The tracking error value during steady-state regime. More precisely, we have calculated for each law and desired motion the following quantities:

$$err1 = \int_{10}^{20} \tilde{p}_{11}^2(t)dt$$
$$err2 = \int_{10}^{20} \tilde{p}_{12}^2(t)dt \qquad (10)$$

- The shape and the magnitude of the input.
- The ability of the controllers to provide acceptable and constant performance for any of the three desired trajectories, with no modification of the feedback gains.

REMARK 2. The closed-loop behavior during the transient period has not been chosen as a fourth criterion. The reason will be explained below. (6.7)

REMARK 3. From a general point of view, the feedback gains have been tuned as follows. We started with the first link fixed with respect to the base, with only the second link to be controlled. The gains of the PD controller were chosen from the second-order approximation obtained by assuming $k = +\infty$. Since the SLI scheme mainly consists of a PD action plus a nonlinear part, those values served as a basis for the SLI gains. The full-order system is linear of order 4, and we tried to tune the gains essentially by placing the closed-loop poles according to simple criteria (optimal response time, nonoscillatory modes). ~~The gains $k_1$ and $k_2$ for controller 1 have been chosen to provide a reduced damping coefficient equal to one for eq. (46).~~ In all cases, trajectory 1 was used to determine a first set of gains. This provided us with a basis to choose the gains for the complete 2-DOF system. We started with trajectory 1, and we modified the gains in real time (essentially by increasing them) until the performance could no longer be improved. Then trajectories 2 and 3 were tested. Note that the tuning of feedback gains for such nonlinear controllers as those tested here is far from evident. Indeed, even in the linear case, it is in general not possible to place the poles arbitrarily. This is due to the structure of the input, which does not reduce to a pole-placement controller even if the plant is linear. For instance, concerning controller 4 (this is similar to the others), we were not able to find a region in the parameter space within which the gains could be significantly modified and, at the same time, the poles could remain real. Therefore, the parameters were tuned (even for the linear case) more or less intuitively.

The quantities $err1$ and $err2$ are given in Tables 1 and 2. They correspond to the results obtained with trajectories 1, 2, and 3. In each case, the results presented in Tables 1 and 2 represent the average of several experiments. Concerning trajectories 2 and 3, the results outside brackets are those obtained after having retuned the feedback gains. Those between brackets have been obtained using the same gains as for trajectory 1. When they are equal, it means that we have not been able to improve the results. The feedback gains for each controller are given in Tables 3–8. These values allow one to evaluate the modifications that have been done and that correspond to the values of $err1$ and $err2$ reported in Tables 1 and 2. The tracking errors $\tilde{p}_{11}$, $\tilde{p}_{12}$ and the inputs $I_{c1}$, $I_{c2}$ for the first and second link, respectively, are depicted

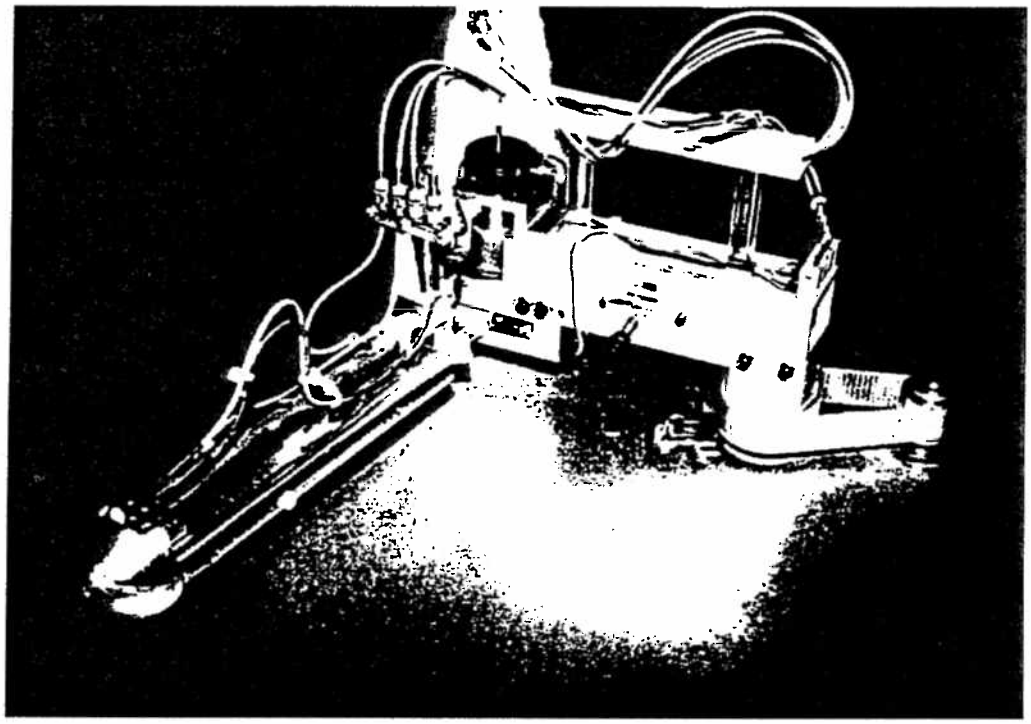

Fig. 2. Capri robot. (Note: Camera-ready artwork unavailable at time of printing.)

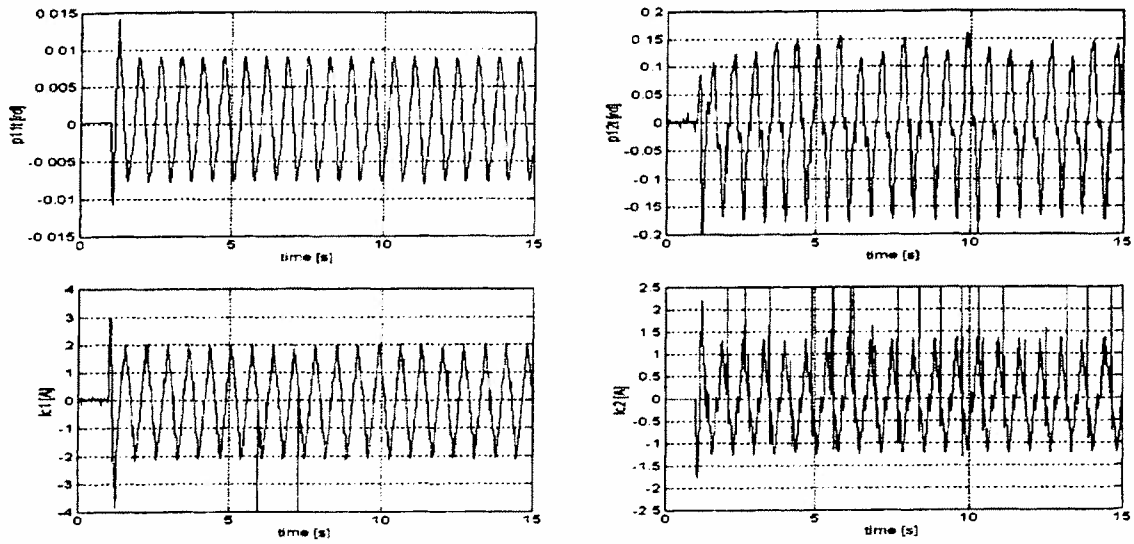


Fig. 3. PD controller, desired trajectory 1.

PD DT1

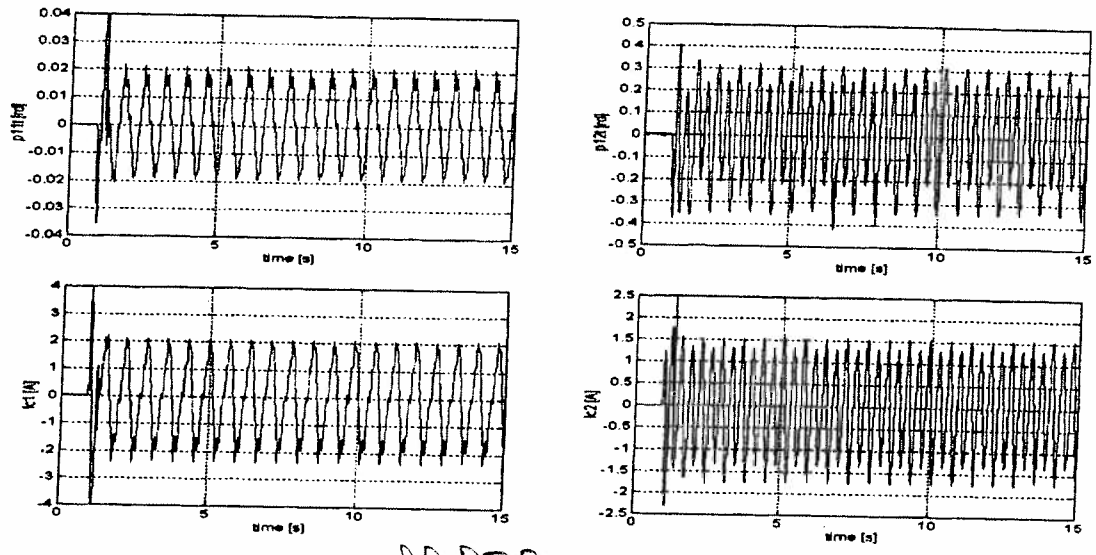


Fig. 4. PD controller, desired trajectory 2.

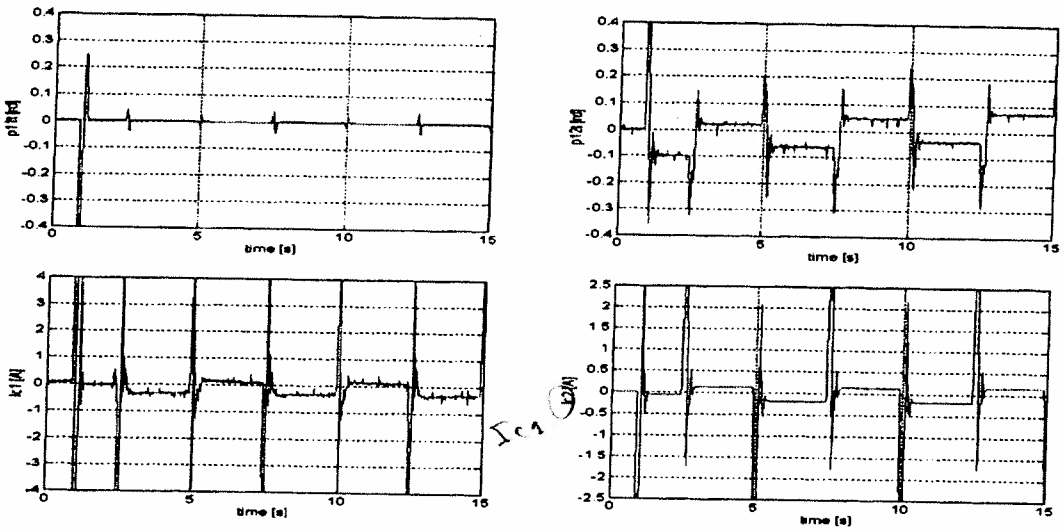


Fig. 5. PD controller, desired trajectory 3.

in Figures 3–20. Notice that the calculated inputs are depicted. The saturations have been chosen as $|I_{c1}| \leq 2A$ and $|I_{c1}| \leq 2A$. Figures 21–32 show a comparison of the closed-loop performance during transient for initial conditions 0, and $\tilde{p}_{22}(0) = \tilde{p}_{12}(0) = 0.4$ rad, $\dot{\tilde{p}}_{12}(0) \neq 0$, $\dot{\tilde{p}}_{22}(0) \neq 0$ (i.e., the second link is initially outside the equilibrium).

Some general comments can be made about Tables 1, 2, and 3–8, and Figures 3–32.

### 4.1. *Adaptation to Desired Motion Changes*

The three desired trajectories are sufficiently different from one another to test the adaptability of the controllers to desired motion changes. The SLI algorithm seems to provide the most invariant performance with respect to the desired motion. This is especially apparent for trajectory 2. In this case, it provides the best $err2$, even after having retuned the gains for controllers 1, 3, 4, and SPB (see the gains values in Tables 3–8). This may be explained by the fact that the SLI input (see $I_{c2}$ in Fig. 7) is much smoother than the other inputs. This in turn may come from the simplicity of this controller, compared to controllers 1, 2, and 3, and from the fact that the SLI input does not use the potentiometer signals $p_{12}$ and $\dot{p}_{12}$. However, despite having a structure quite similar to the SLI, the SPB controller provides a very poor behavior for trajectory 2. We have not been able to find a satisfactory explanation of this strange behavior, although this might be due to some resonance phenomenon (the closed-loop behav-

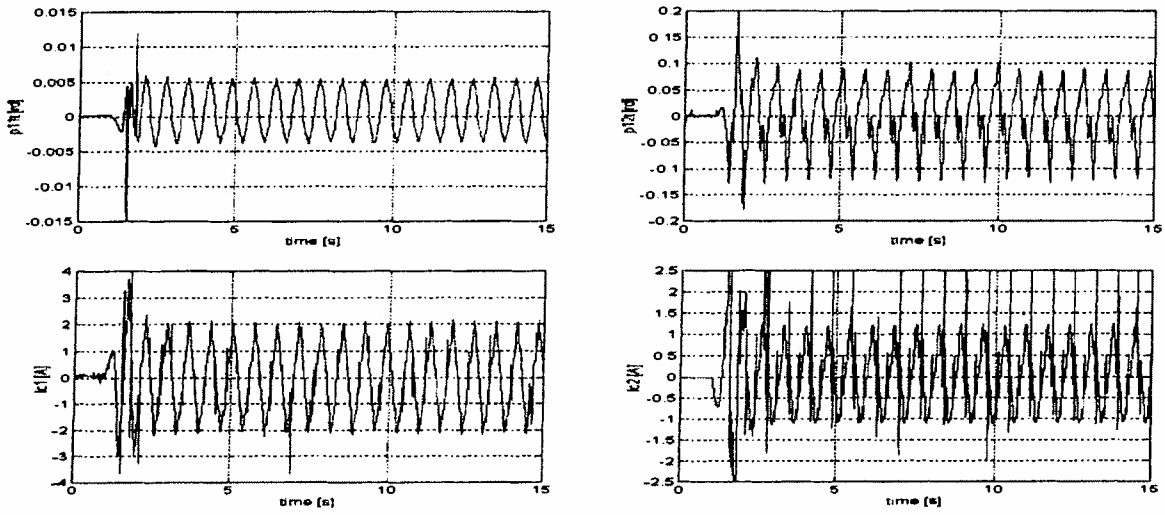


Fig. 6. SLI controller, desired trajectory 1. SLIDT2

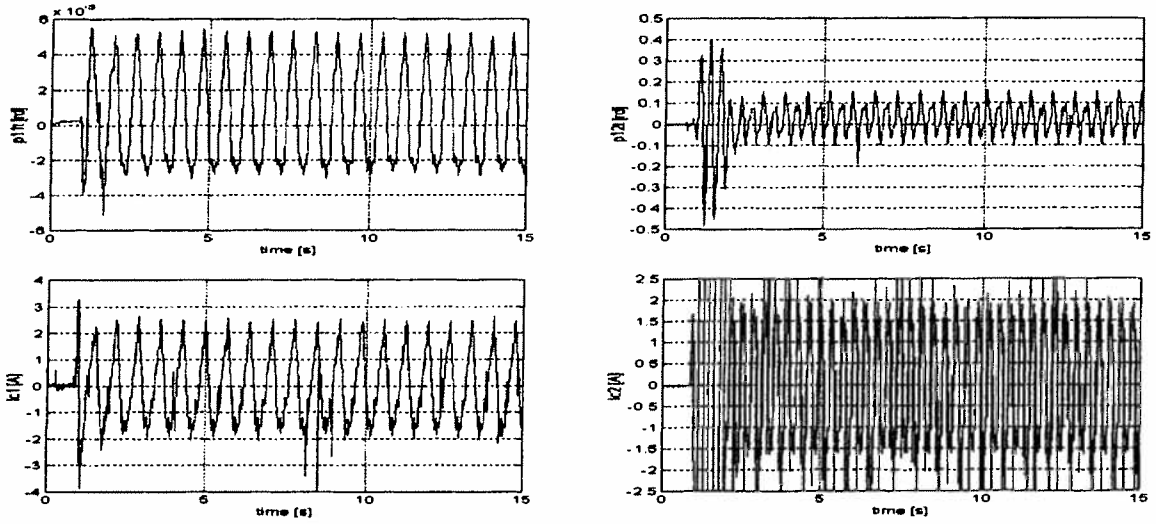


Fig. 7. SLI controller, desired trajectory 2. SLIDT2

ior remains acceptable during a short transient, and then both the input and $\tilde{p}_{12}$ start to oscillate; see Fig. 10). Note finally that controller 1 possesses a much worse adaptation to desired trajectories modifications (see Table 3). This may be explained by more chattering input for trajectory 1 (compare $I_{c2}$ on Fig. 12 and Figs. 15 and 18), which prevents the closed loop to behave correctly without gain modifications. Tables 1 and 2 tend to show that controller 1 behaves significantly worse than controllers 3 and 4 and SLI.

### 4.2. *Backstepping Controllers*

We have not been able to find feedback gains such that controller 2 may stabilize the system. This is due to a too-large input magnitude that hampers stabilization. This confirms the result in Brogliato, Ortega, and Lozano (1995) and shows that an apparently slight modification of the backstepping method has a significant influence on the obtained controller.

### 4.3. *Compensation of Nonlinearities*

Although the PD algorithm provides a stable closed-loop behavior in all cases (but at the price of significant gains modifications), its performance is poor for trajectories 1 and 2. The behavior is much better for trajectory 3, since this is almost a regulation task. The improvements obtained with the SLI scheme show that nonlinearities play an important role in the

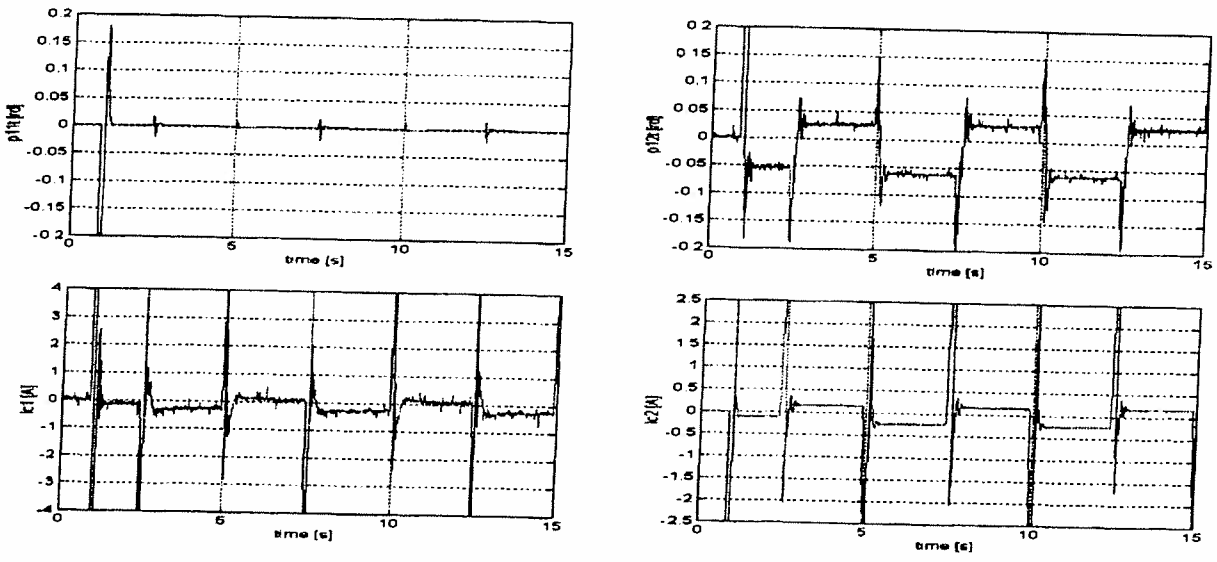


Fig. 8. SLI controller, desired trajectory 3.

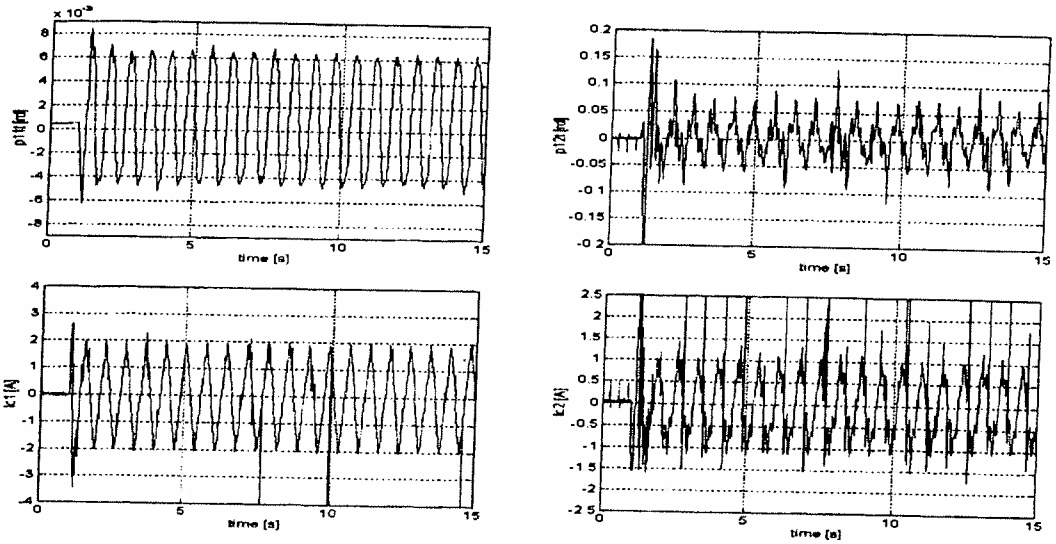


Fig. 9. SPB controller, desired trajectory 1.

system with the desired motions chosen. Finally, these experiments prove that despite a relatively high joint stiffness value (50 Nm/rad), the compensation of the flexibility effects can significantly improve the closed-loop behavior.

### 4.4. *Controller Complexity*

The rather complex structure of controllers 1, 3, and 4 is not an obstacle to their implementation with the available real-time computer described above. In particular, recall that the acceleration $\ddot{q}_1$ and jerk $q_1^{(3)}$ are estimated by inverting the dynamics (this is an inherent fact of backstepping and energy-shaping methods). Such terms have a complicated structure and depend on the physical parameters of the system in a nonlinear way (see Lozano and Brogliato 1992). Some measurements have shown that the sampling period (1 ms) could have been decreased to 0.5 ms.

### 4.5. *Torque Input*

The major problem that prevents certain controllers to behave correctly is the input magnitude and shape. This is clear in the case of controller 2, as noted above. This has also been noted for controllers 1, 3, and 4; their performance for trajectory 2 is less good than that of SLI, mainly because of

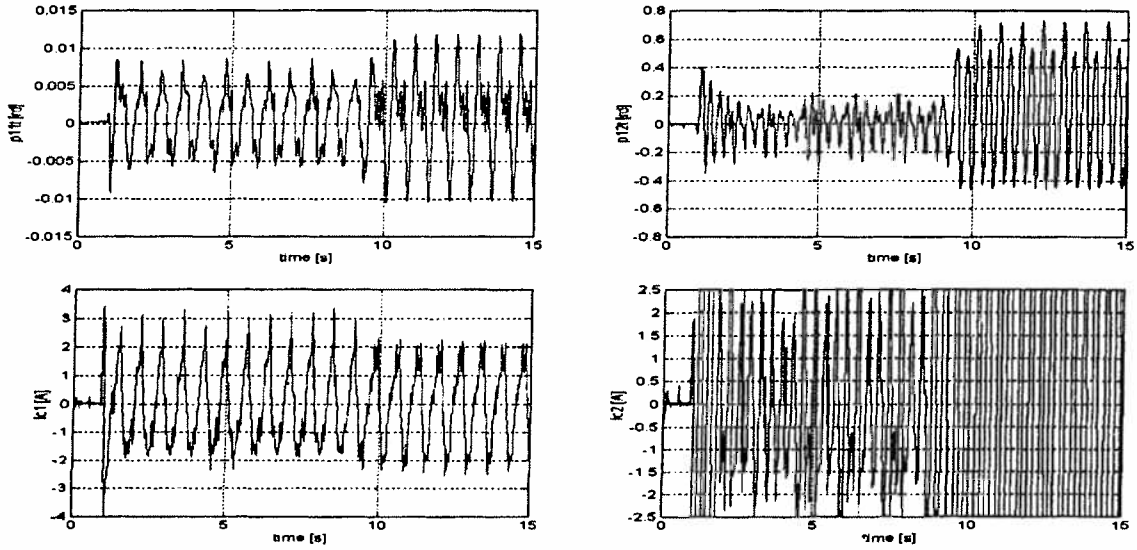


Fig. 10. SPB controller, desired trajectory 2.

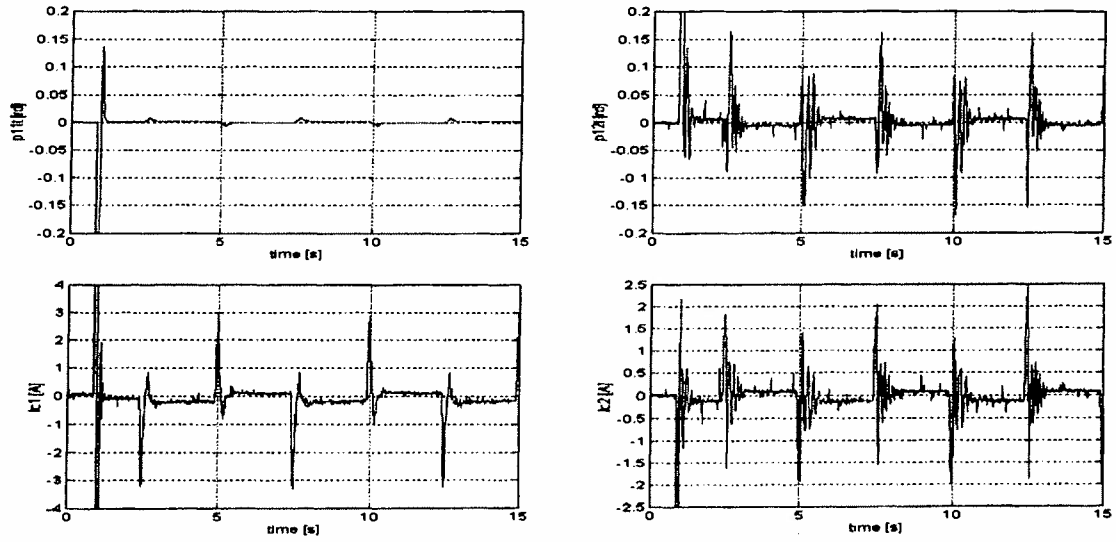


Fig. 11. SPB controller, desired trajectory 3.

chattering in the input, inducing vibrations in the mechanical structure. Chattering is particularly present during the regulation phases in $I_{c2}$ for trajectory 3 and controllers SPB, 1, 3, and 4 (see Figs. 11, 14, 17, and 20), although the SPB input chatters much less than those of controllers 1, 3, and 4 (compare Figs. 9, 10, 11, 2, 26 and Figs. 12–20 and 27–32). On the contrary, the input is very smooth for both the PD and SLI inputs (see Figs. 5 and 8). It can also be seen in Figures 27–28 and 29–32 that controller 1 yields an input that chatters much more than controllers 3 and 4. It may be expected from Figures 29–32 that a better velocity $\dot{p}_{12}$ measurement would bring the shape of $I_{c2}$ in Figures 29–32 closer to that in Figures 23–24 (SLI controller); indeed, they differ only in terms of chatter. More generally, a better velocity measurement would certainly reduce the chattering in all the inputs that feed back $\dot{p}_{12}$.

### 4.6. *Energy Shaping versus Backstepping*

In relation to the comments made in Section 2 on the difference between backstepping and energy-shaping controllers, it is noteworthy that controller 3 (backstepping) and controller 4 (energy shaping) possess quite similar closed-loop behaviors (although $I_{c2}$ chatters slightly less for controller 4; see Figs. 15 and 18 and 17 and 20). The advantage of the energy-shaping method may be that the controller is obtained

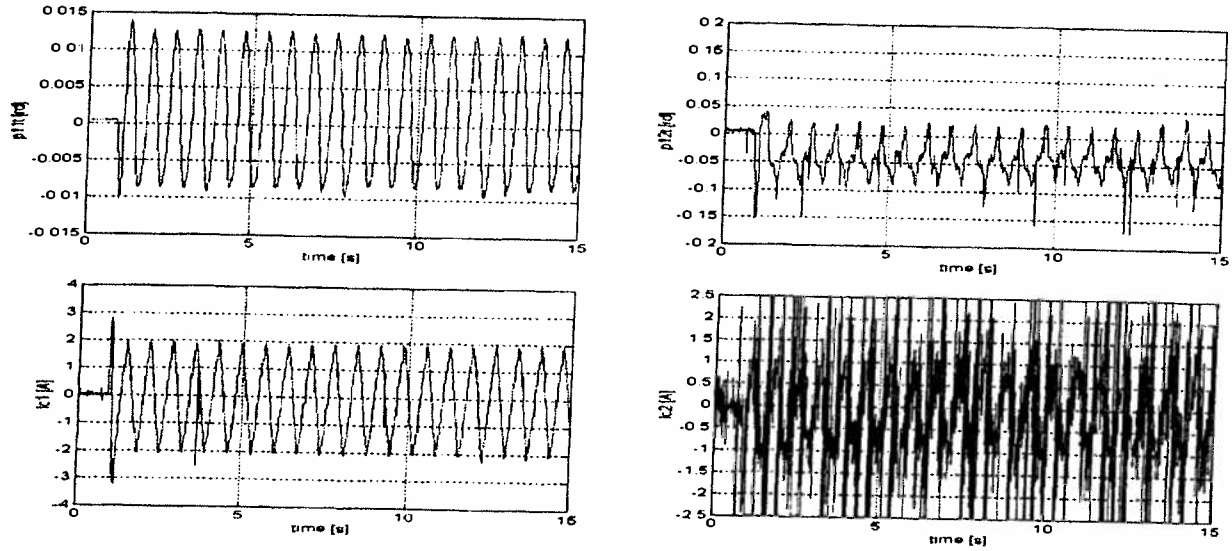


Fig. 12. Controller 1, desired trajectory 1.

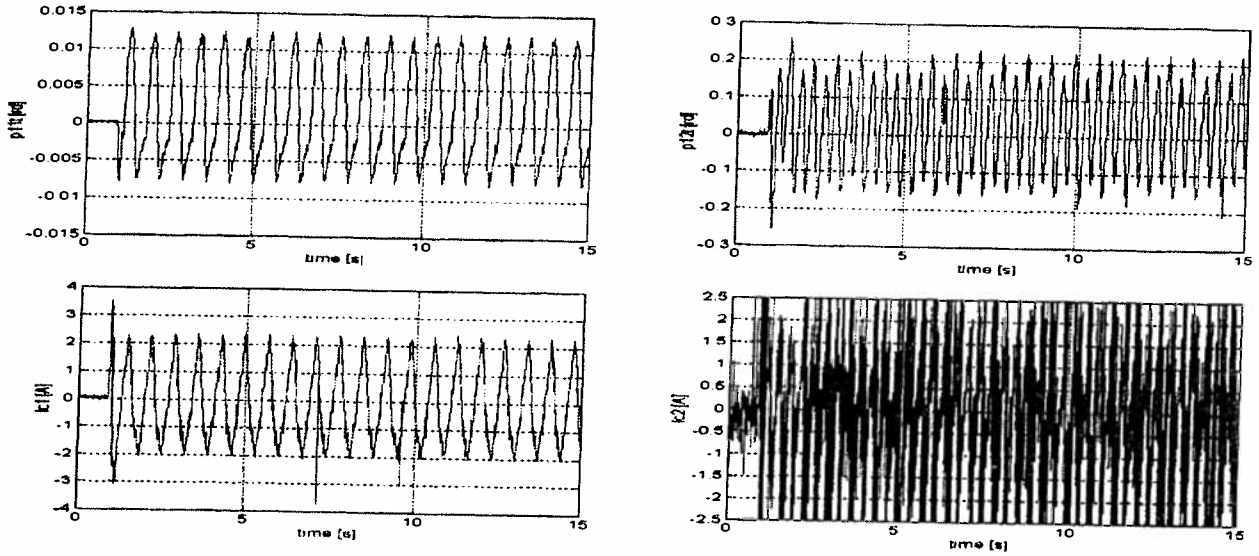


Fig. 13. Controller 1, desired trajectory 2.

in one shot, whereas the backstepping method a priori leads to various algorithms.

### 4.7. *Transient Behavior*

The transient behavior for the tracking error $\tilde{p}_{12}$ can be improved slightly when the flexibilities are taken into account in the controller. This can be seen by comparing Figures 6 and 7 (SLI) with Figures 12 and 13 (controller 1), 15 and 16 (controller 3), and 18 and 19 (controller 4). The tracking error tends to oscillate more for the SLI algorithm than for the other algorithms. Note that those results have been obtained with initial tracking errors $\tilde{p}_{11}$, $\tilde{p}_{12}$, $\tilde{p}_{22}$ close to zero. However, the results in Figures 21–32 prove that the controllers respond quite well to initial state deviation; the transient duration is around 0.5 s for all the controllers. The tracking errors have a similar shape once the transient has vanished, and we have checked that $err2$ has the same value also. The only significant difference is in the initial input $I_{c2}$; the torque is initially much higher for nonzero initial conditions.

In summary, the SLI algorithm proves to be quite robust with respect to unmodeled dynamics (flexibilities) and to desired motion modifications.However, significant improvement in the tracking error can be obtained by taking the joint flexibility into account in the controller design. This also

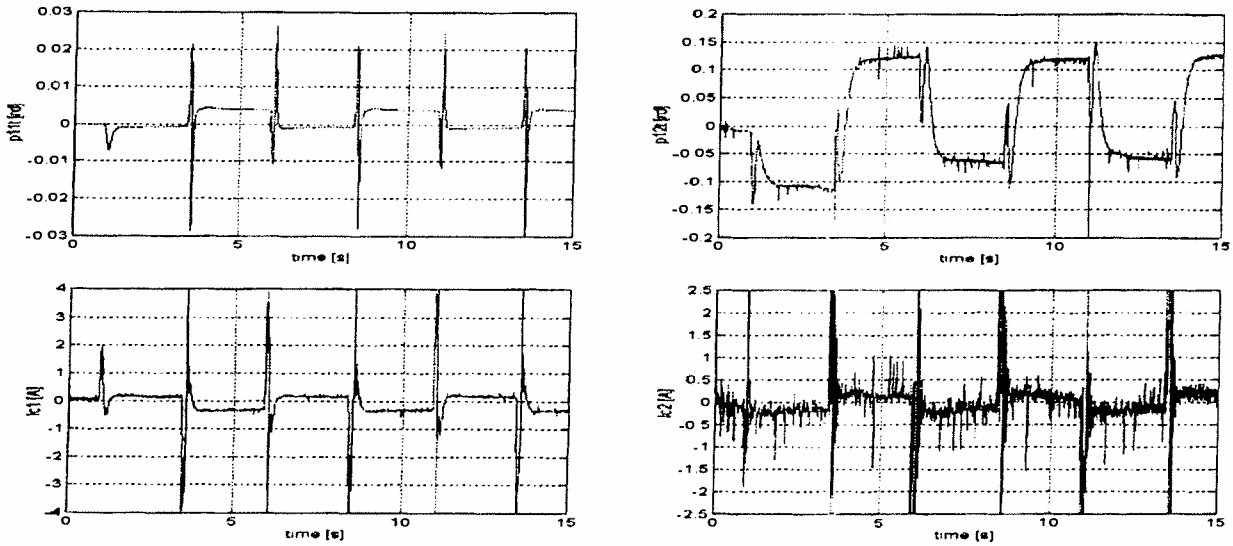


Fig. 14. Controller 1, desired trajectory 3.

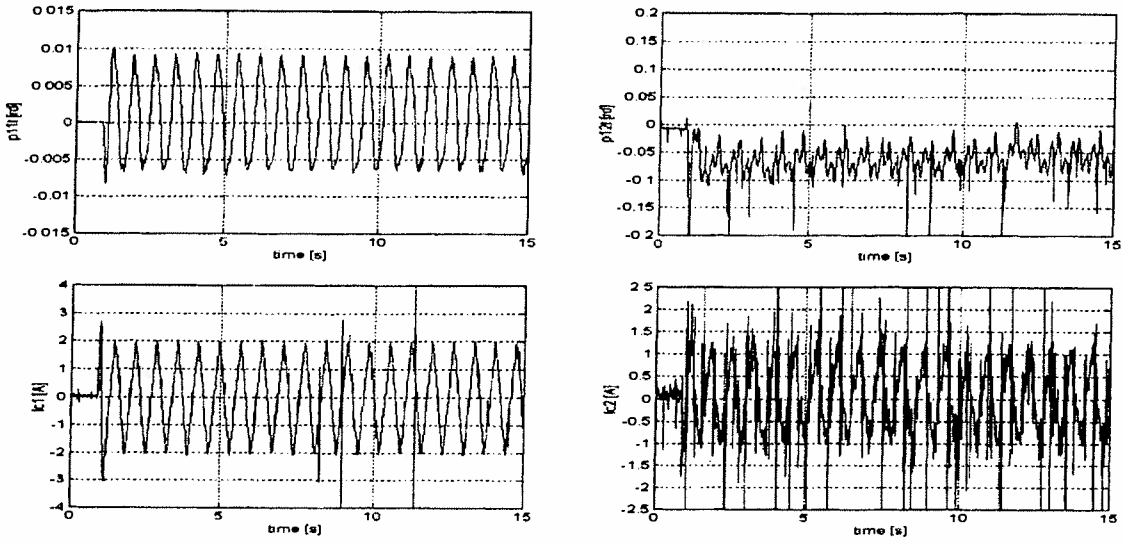


Fig. 15. Controller 3, desired trajectory 1. CT3 DT1

allows one to improve the transient behavior. The major limitation of the nonlinear controllers 1, 2, 3, 4, and SPB is due to the input magnitude and/or chattering. Although chattering may certainly be reduced by better measurement of $p_{12}$ and $\dot{p}_{12}$ (for instance, replacing the potentiometer by an incremental encoder), the size of the input is a more serious problem. Compared to the SLI scheme, controllers 1, 3, and 4 also suffer from a much higher sensitivity with respect to parameter uncertainties. Indeed, as noted above, they contain some acceleration and jerk estimation that rely on a good knowledge of the physical parameters (recall that the SPB controller does not rely on such estimations, which certainly explains its smoother input). As discussed in the introduction, the existing adaptive solutions are not expected to provide an acceptable answer. The usefulness of such controllers might then be called into question. However, it is noteworthy that the joint stiffness of the Capri robot is rather high, and the good closed-loop behavior of the SLI scheme is not surprising. In the future, it will be interesting to make experiments and another comparison with a low flexibility. This will enable us to draw more general conclusions about the respective behaviors and domains of applicability of the tested controllers.

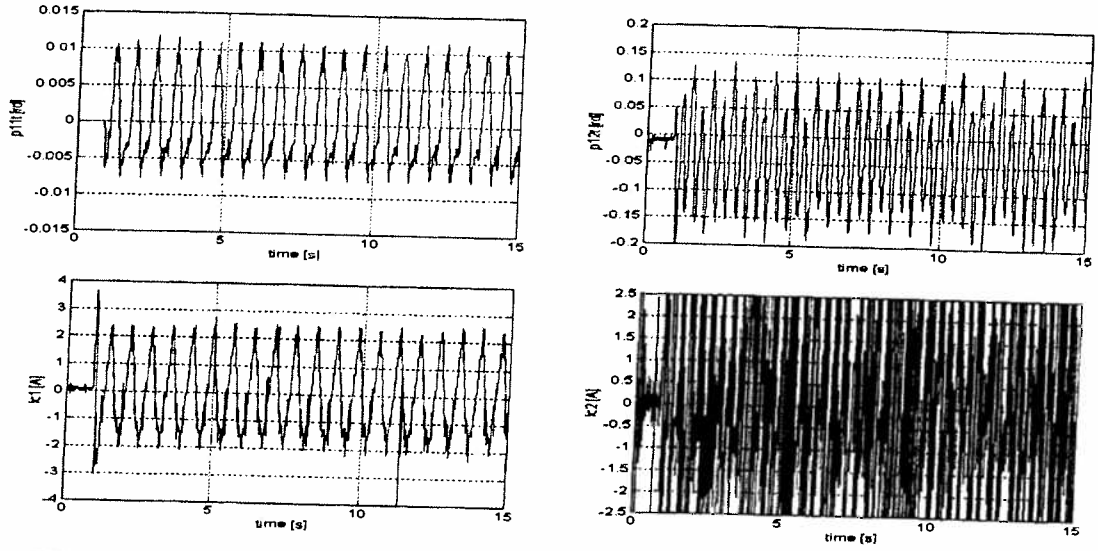


Fig. 16. Controller 3, desired trajectory 2. CT3DT2

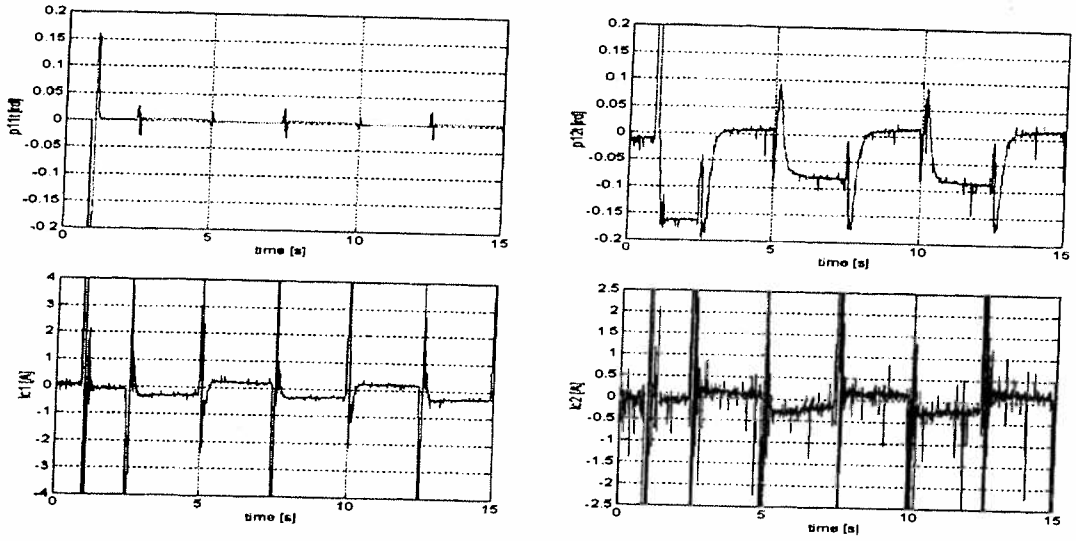


Fig. 17. Controller 3, desired trajectory 3. CT3DT3

REMARK 4. One can notice many peaks in the input $I_{c2}$ for trajectory 1 (see Figs. 3, 6, 9, 12, 15, 18, 26, and 29–32). They are due to saturation of the DC tachometers when the trajectory is at the maximum speed. When the saturation ends, the velocity signal delivered by the tachometers has a short noisy transient that produces such peaks in the input. Note that this has not had significant influence on performance, since those peaks are naturally filtered by the motors (recall that the calculated inputs are depicted).

### 4.8. *Conclusions*

In this paper, we have presented experimental results of nonlinear control of flexible joint manipulators. Seven different schemes were implemented on a 2-DOF planar manipulator and compared: a PD controller, the Slotine and Li scheme, a singular perturbation-based algorithm, and four controllers derived from the backstepping and energy-shaping design methods. This work completes a preceding study (Brogliato, Ortega, and Lozano 1995) that contains a theoretical and numerical comparative study of the four backstepping and energy-shaping controllers. Reliability of the experimental conclusions concerning the backstepping and energy-shaping controllers is enhanced by the fact that they are in accordance with the remarks and comments made in Brogliato, Ortega, and Lozano (1995). This work does not classify several recently proposed control methodologies. It only proves that, when applied to the particular device that we have at our dis-

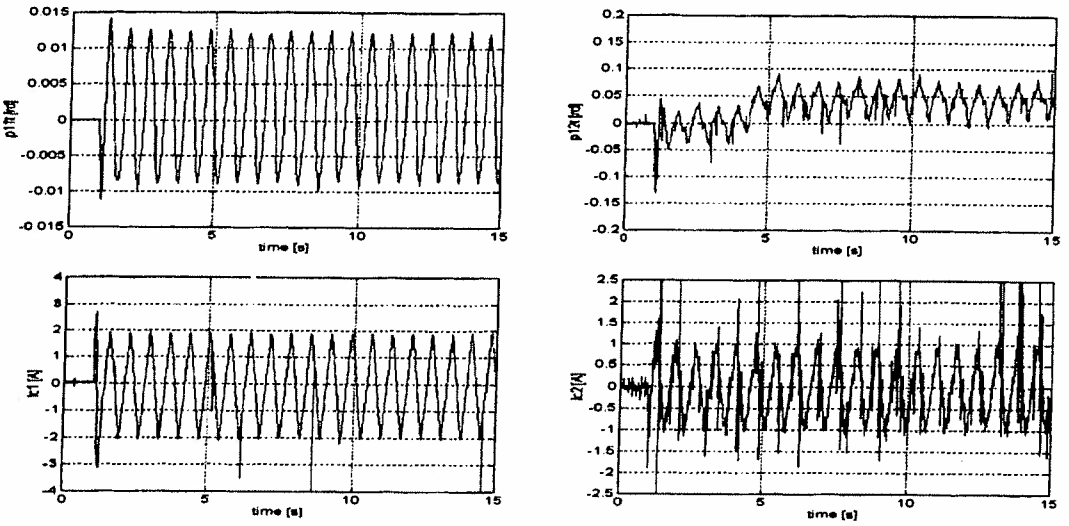


Fig. 18. Controller 4, desired trajectory 1. CT4 DT1

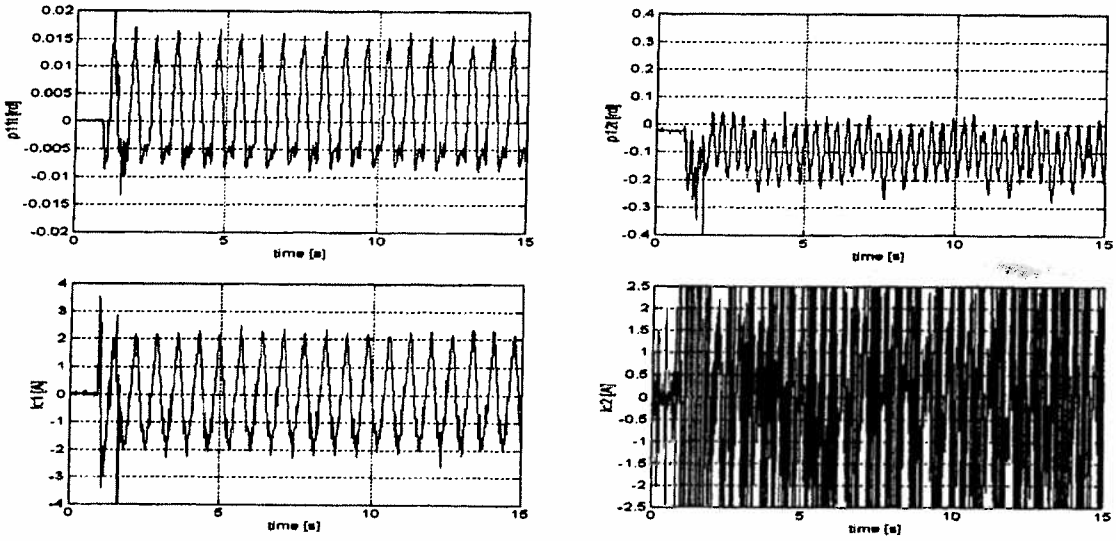


Fig. 19. Controller 4, desired trajectory 2. CT4 DT2

posal in our laboratory, some of them provide much better results. The major conclusions are that although a simple PD controller may work even for high-speed trajectories (but with significant feedback gains modifications to adapt to trajectory changes), it is worth compensating for the smooth nonlinearities and taking the joint flexibilities into account in the control design. In particular, this enables one to obtain acceptable performance with a fixed set of feedback gains. Although the results may appear to be mitigated (there are trajectories in our experiments for which the Slotine and Li controller performs as well, even slightly better, than those that incorporate the joint flexibility in the design), the tracking errors can be significantly decreased in certain cases. It is clear that the rather large value of the joint stiffness plays a crucial role in this results, as well as the crude estimation of the second link velocity. Future experimental work should concern a system with a much more significant flexibility. Note further that the performance gain obtained by incorporating the joint flexibility in the control design may produce inputs with larger magnitude and more chattering. Another conclusion is that the four nonlinear schemes developed and analyzed in Brogliato, Ortega, and Lozano (1995) possess quite different closed-loop behaviors, despite their apparent similarities. Finally, the backstepping and energy-shaping methods may yield controllers with equivalent closed-loop performance. An advantage of the first method is that it pro-

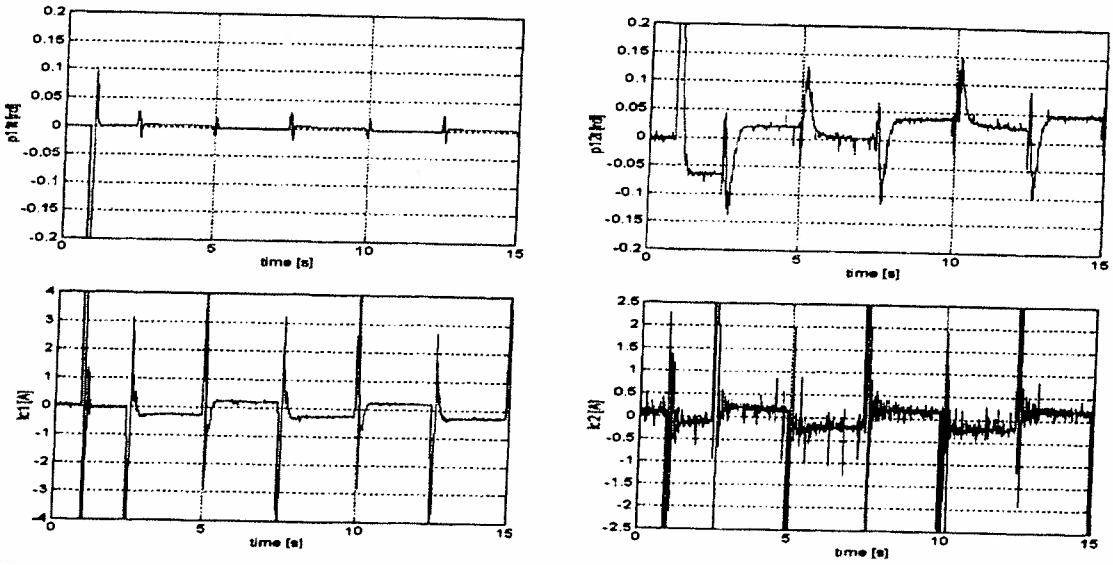


Fig. 20. Controller 4, desired trajectory 3. CT4DT3

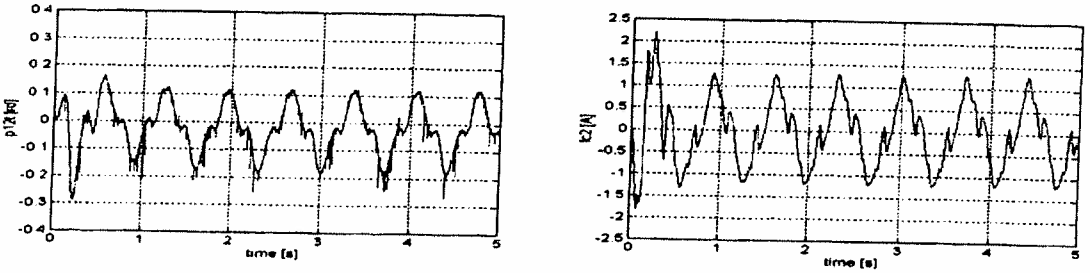


Fig. 21. PD controller, desired trajectory 1, zero initial conditions. PDDT1ZI

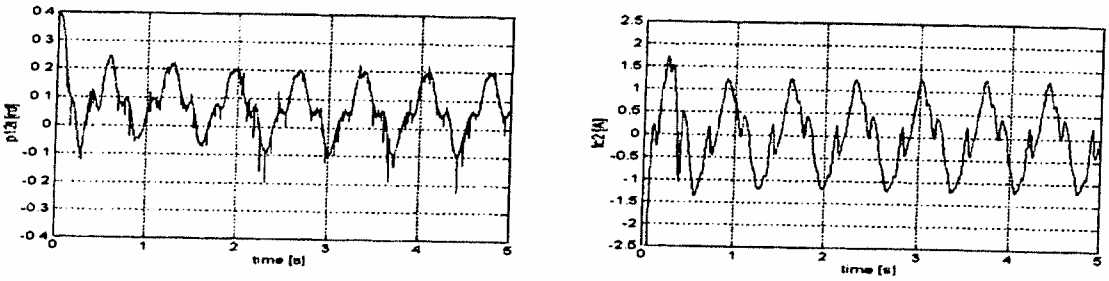


Fig. 22. PD controller, desired trajectory 1, nonzero initial conditions. PDDT1NZI

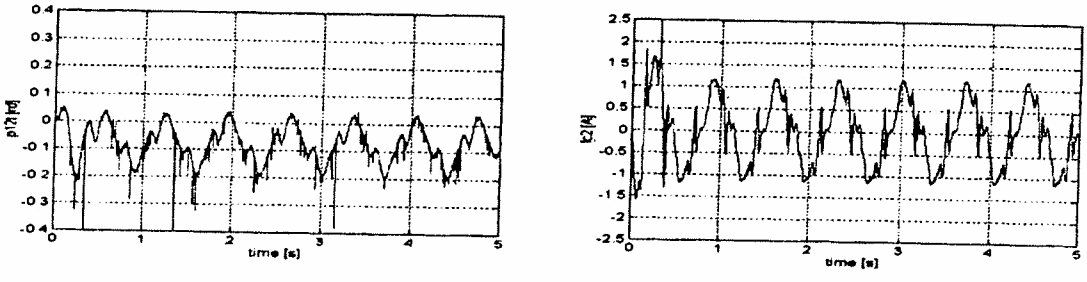


Fig. 23. SLI controller, desired trajectory 1, zero initial conditions.

SLIDT1ZI

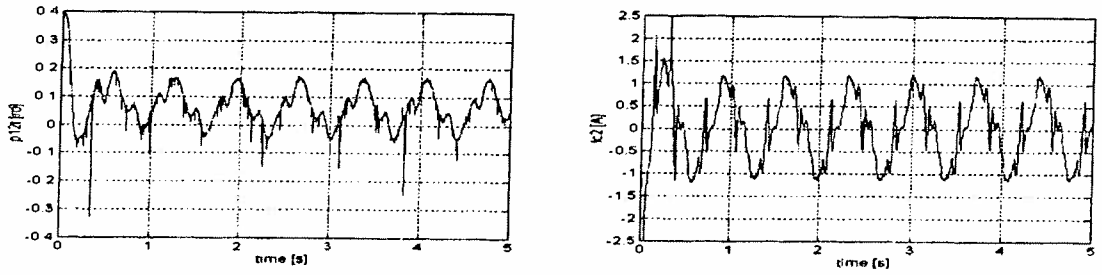


Fig. 24. SLI controller, desired trajectory 1, nonzero initial conditions.

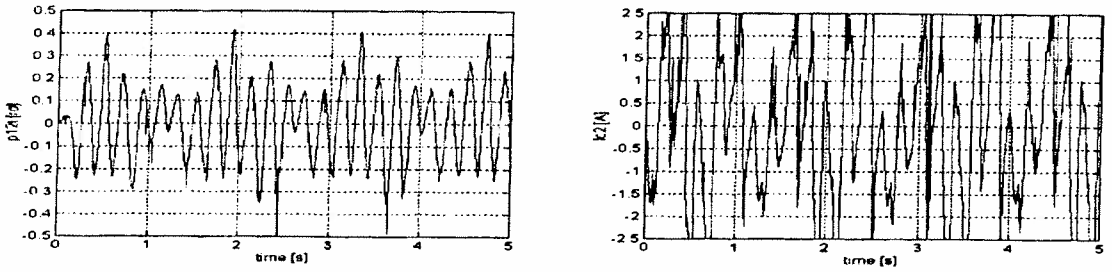


Fig. 25. SPB controller, desired trajectory 1, zero initial conditions.

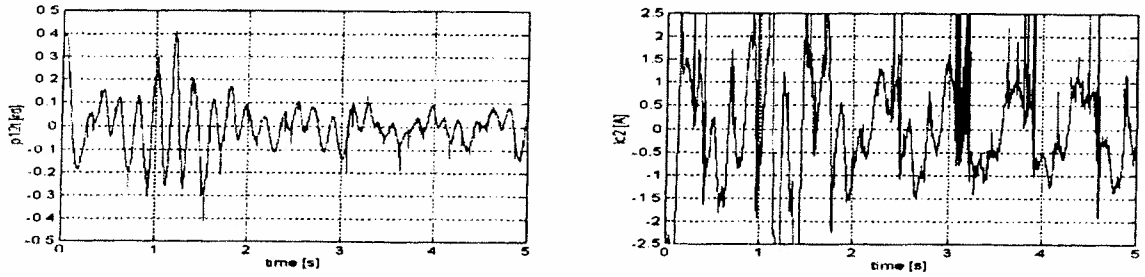


Fig. 26. SPB controller, desired trajectory 1, nonzero initial conditions.

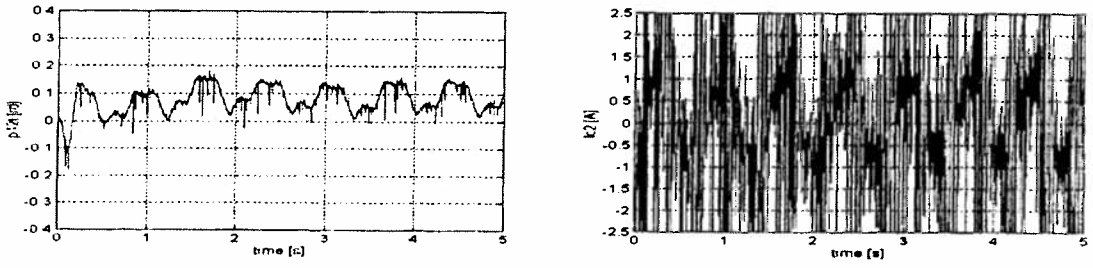


Fig. 27. Controller 1, desired trajectory 1, zero initial conditions.

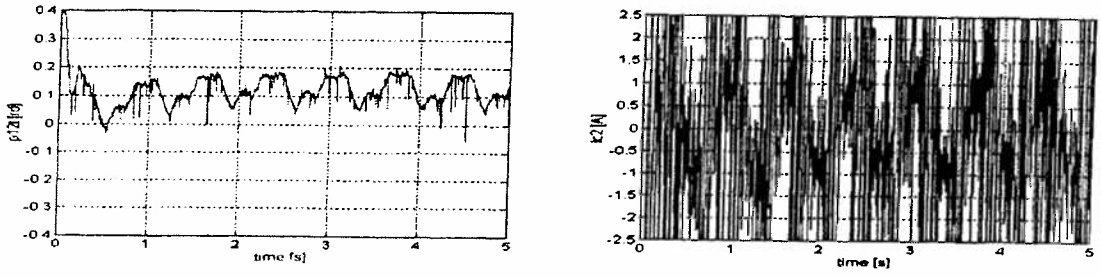

Fig. 28. Controller 1, desired trajectory 1, nonzero initial conditions.

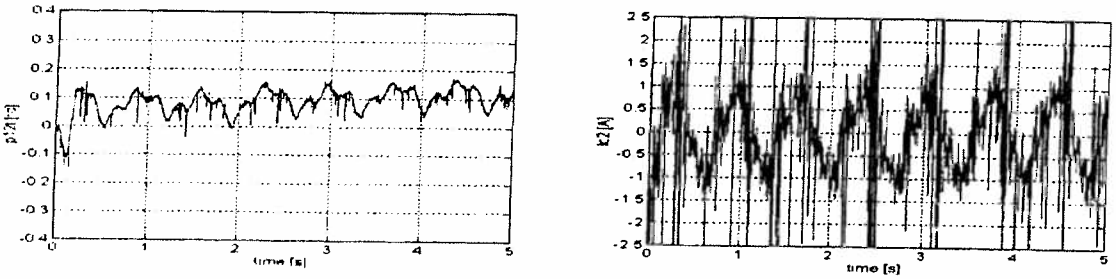

Fig. 29. Controller 3, desired trajectory 1, zero initial conditions. CT3 DT1 Z I

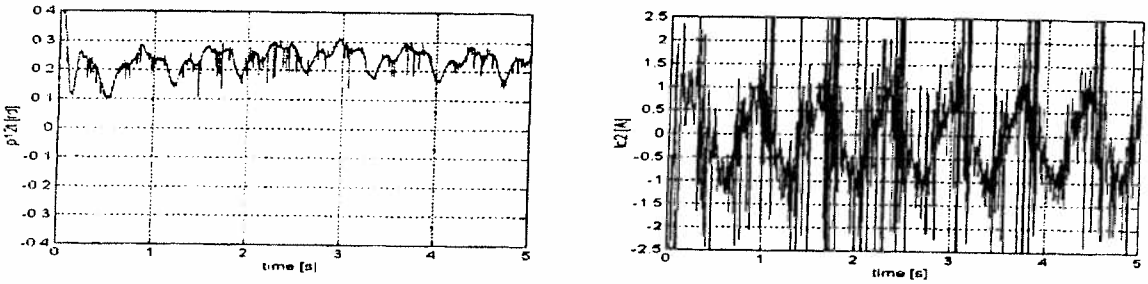

Fig. 30. Controller 3, desired trajectory 1, nonzero initial conditions. CT3DT1 NZI

vides an algorithm in one shot, whereas the backstepping method may lead to high-gain inputs that do not work in practice due to input saturations.

## Appendix A

From the kinematic scheme given in Figure 1, it is easy to see that the $z$-axis of all linkbound coordinate frames has the same direction; that is, the $z$-axis of the base inertial frame (BIF). Furthermore, all the pulleys and the motor shafts rotate around an axis that is parallel to the $z$-axis of the BIF. The momentum of inertia has to be calculated only with respect to the $z$-axis. Let the notation be fixed as follows: $I_M$ denotes the momentum of inertia of the rotors $A_1, B_1$ with respect to the $OM1-z, OM2-z$ axis; $I_{P2}$ denotes the momentum of inertia of the pulleys $A2$, $B2$ with respect to the $OM1-z$, $OM2-z$ axis; $\rho_1$ denotes the gear ratio between the pulleys $A2$, $B2$ and the pulleys$A4$, $B4$, respectively; $I_{P4}$ denotes the momentum of inertia of the pulleys $A4$, $B4$ with respect to the $OR1-z, OR2-z$ axis; $\rho_2$ denotes the gear ratio of the gears $A5$, $B5$; $I_R$ denotes the input inertia gear about the $OR1-z$ axis; $I_{P6}$ denotes the momentum of inertia of the pulleys $A6$, $B6$ with respect to the $OR1-z, OR2-z$ axis; $\rho_3$ denotes the gear ratio between the pulleys $A6$, $B6$ and the pulleys $A8$, $B8$, respectively; $I_{B9}$ denotes the momentum of inertia of the pulleys $B8$, $B9$ about the $O1-z$ axis; $I_1$ denotes the momentum of inertia of the first link including the $A8$ pulley with respect to the $O1-z$ axis; $I_2$ denotes the momentum of inertia of the second link including the $B11$ pulley about the $G2-z$ axis, where $G2$ denotes the center of the mass of the second link; $m_2$ denotes the mass of the second link including the mass of the $B11$ pulley; $l_1$ denotes the length of the first link; $a_2$ denotes the distance from $O2$ to the center of the mass of the second link; and $k$ denotes the elastic constant of the $B10$ cable. The gear ratio between the pulleys $B9$ and $B11$ is equal to one. The parameters of

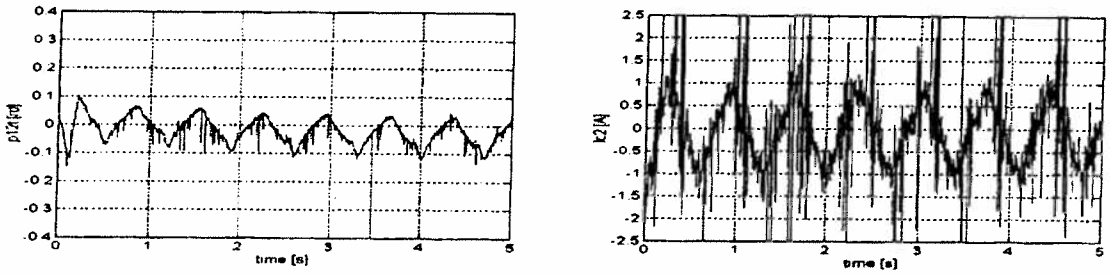


Fig. 31. Controller 4, desired trajectory 1, zero initial conditions.

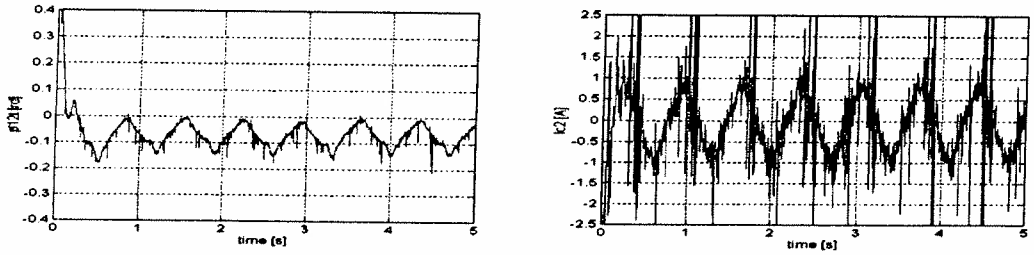


Fig. 32. Controller 4, desired trajectory 1, nonzero initial conditions.

the Capri robot are calculated to be $I_M = 21 \ .10^{-6}$ Kg.m$^2$, $I_{P2} = 10^{-6}$ Kg.m$^2$, $I_{P4} = 1.77 \ 10^{-6}$ Kg.m$^2$, $I_M = 7 \ .10^{-6}$ Kg.m$^2$, $I_{P6} = 1.36 \ 10^{-6}$ Kg.m$^2$, $I_1 = 3.443 \ 10^{-3}$ Kg.m$^2$, $I_2 = 4.366 \ 10^{-3}$ Kg.m$^2$, $I_{B9} = 2 \ .10^{-6}$ Kg.m$^2$, $\rho_1 = 1/2.5$, $\rho_2 = 1/10$, $\rho_3 = 1/2$, $m_2 = 0.732$ Kg, $l_1 = 0.16$ m, $a_2 = 0.15$ m.

The dynamic model of the Capri robot is derived from Lagrange's equations, choosing the generalized coordinates $p_{11}$, $p_{12}$, and $p_{22}$ as depicted in Figure 1. Let the vectors $p_1$, $p_2 \in \mathbf{R}^2$ be defined as follows:

$$p_1 := [p_{11}, p_{12}]^T$$
$$p_2 := [p_{11}, p_{22}]^T,$$

where $p_{21} = p_{11}$, since the first joint is assumed to be rigid. The total kinetic energy is given by

$$T = \frac{1}{2}\dot{p}_1^T M(p_1)\dot{p}_1 + \frac{1}{2}\dot{p}_2^T J \dot{p}_2, \tag{11}$$

where

$$M(p_1) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \quad J = \begin{bmatrix} J_{11} & 0 \\ 0 & J_{22} \end{bmatrix}$$

$$M_{11} = m_2 l_1^2 + I_1$$
$$M_{12} = m_2 l_1 a_2 cos(p_{12} - p_{11})$$
$$M_{21} = M_{12}$$
$$M_{22} = m_2 a_2^2 + I_2$$
$$J_{11} = \frac{I_M + I_{P2}}{(\rho_1\rho_2\rho_3)^2} + \frac{I_{P4} + I_R}{(\rho_2\rho_3)^2} + \frac{I_{P6}}{(\rho_3)^2}$$
$$J_{22} = J_{11} + I_{B9}.$$

Since the generalized coordinate $p_{22}$ is taken with respect to the same frame as the generalized coordinate $p_{12}$, the total potential energy coincides with the elastic potential energy; that is,

$$U = \frac{1}{2}k(p_{12} - p_{22})^2 + U_0, \tag{12}$$

where $k \in \mathbf{R}$, $k > 0$ is the elastic constant of the cable $B10$ as depicted in Figure 1.

The system's Lagrangian is given by

$$L = T - U. \tag{13}$$

The Lagrange equations are given by

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{p}_{11}}\right) - \frac{\partial L}{\partial p_{11}} = \tau_1$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{p}_{12}}\right) - \frac{\partial L}{\partial p_{12}} = 0$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{p}_{22}}\right) - \frac{\partial L}{\partial p_{22}} = \tau_2,$$

where $\tau_1$, $\tau_2 \in \mathbf{R}$ are the torques of the two motors, respectively. The dynamic equation of the Capri robot takes the following form:

$$D(p_1)\ddot{p}_1 + C(p_1, \dot{p}_1)\dot{p}_1 + K(p_1 - p_2) = [\tau_1, 0]^T \tag{14}$$

$$J_{22}\ddot{p}_{22} - k(p_{12} - p_{22}) = \tau_2, \tag{15}$$

where

$$D(p_1) = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \quad C(p_1, \dot{p}_1) = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

**Table 3. Feedback Gains.**

| *PD controller* | traj.1 | traj.2 | traj.3 |
|---|---|---|---|
| $k_{p1}$ | 1500 | 650 | 1500 |
| $k_{p2}$ | 250 | 10 | 250 |
| $k_{d1}$ | 30 | 4 | 30 |
| $k_{d2}$ | 5 | 3.5 | 5 |

**Table 4.**

| *SLI controller* | traj.1 | traj.2 | traj.3 |
|---|---|---|---|
| $\lambda^1$ | 50 | 50 | 50 |
| $\lambda^2$ | 50 | 65 | 50 |
| $k_{d11}$ | 42 | 24 | 42 |
| $k_{d22}$ | 6 | 20 | 6 |

**Table 5.**

| *SPB controller* | traj.1 | traj.2 | traj.3 |
|---|---|---|---|
| $\lambda^1$ | 48 | 48 | 48 |
| $\lambda^2$ | 40 | 40 | 40 |
| $k_v$ | 10 | 10 | 10 |
| $k_{d11}$ | 24 | 24 | 24 |
| $k_{d22}$ | 4.4 | 4.4 | 4.4 |

**Table 6.**

| *controller 1* | traj.1 | traj.2 | traj.3 |
|---|---|---|---|
| $\lambda^1$ | 25 | 25 | 25 |
| $\lambda^2$ | 30 | 30 | 25 |
| $k_1$ | 53 | 53 | 34 |
| $k_2$ | 3.5 | 3.5 | 2.8 |
| $k_{d11}$ | 24 | 24 | 24 |
| $k_{d22}$ | 14 | 14 | 10 |

**Table 7.**

| *controller 3* | traj.1 | traj.2 | traj.3 |
|---|---|---|---|
| $\lambda^1$ | 44 | 44 | 44 |
| $\lambda^2$ | 52 | 52 | 52 |
| $\lambda_2$ | 18 | 35 | 18 |
| $\lambda_3$ | 18 | 35 | 18 |
| $k_{d11}$ | 20 | 20 | 20 |
| $k_{d22}$ | 5.5 | 5.5 | 5.5 |

**Table 8.**

| *controller 4* | traj.1 | traj.2 | traj.3 |
|---|---|---|---|
| $\lambda^1$ | 48 | 48 | 48 |
| $\lambda^2$ | 14.6 | 40 | 14.6 |
| $k_{d11}$ | 12 | 12 | 12 |
| $k_{d22}$ | 1.2 | 1.4 | 1.2 |
| $k_{d33}$ | 6.6 | 8 | 6.6 |

$$K = \begin{bmatrix} 0 & 0 \\ 0 & k \end{bmatrix} \tag{16}$$

$$\begin{aligned} d_{11} &= m_2 l_1^2 + I_1 \\ d_{12} &= m_2 l_1 a_2 \cos(p_{12} - p_{11}) \\ d_{21} &= d_{12} \\ d_{22} &= m_2 a_2^2 + I_2 \end{aligned} \tag{17}$$

$$\begin{aligned} c_{11} &= 0 \\ c_{12} &= -m_2 l_1 a_2 \sin(p_{12} - p_{11})\dot{p}_{12} \\ c_{21} &= -m_2 l_1 a_2 \sin(p_{12} - p_{11})\dot{p}_{11} \\ c_{22} &= 0. \end{aligned} \tag{18}$$

## Appendix B

This appendix is devoted to explicitly calculating the control laws for the Capri robot following the methods presented in Brogliato, Ortega, and Lozano (1995). In the following, $B_1 = \begin{bmatrix} k_{d11} & 0 \\ 0 & k_{d22} \end{bmatrix}$, $B_2 = k_{d33}$, $K_1 = k_1$, $K_2 = k_2$, see (2) and (5) for the notations.

### *Controllers 2 and 3*

Let the dynamical equations of the Capri robot described by (14) and (15) be considered. Choosing the following feedback laws:

$$\tau_1 = d_{11}\ddot{q}_{11r} + d_{12}\ddot{q}_{12r} + c_{11}\dot{q}_{11r} + c_{12}\dot{q}_{12r} - k_{d11}s_{11} \tag{19}$$

$$\tau_2 = J_{22}v - k(p_{12} - p_{22}), \tag{20}$$

eqs. (14) and (15) reduce to the so-called cascade form:

$$d_{11}\dot{s}_{11} + d_{12}\dot{s}_{12} + c_{11}s_{11} + c_{12}s_{12} - k_{d11}s_{11} = 0 \tag{21}$$

$$d_{21}\ddot{p}_{11} + d_{22}\ddot{p}_{12} + c_{21}\dot{p}_{11} + c_{22}\dot{p}_{12} + kp_{12} = kp_{22} \tag{22}$$

$$\ddot{p}_{22} = v, \tag{23}$$

where $\dot{q}_{11r} = \dot{p}_{11d} - \lambda^1\tilde{p}_{11}$, $\dot{q}_{12r} = \dot{p}_{12d} - \lambda^2\tilde{p}_{12}$, $s_{11} = \dot{\tilde{p}}_{11} + \lambda^1\tilde{p}_{11}$, $s_{12} = \dot{\tilde{p}}_{12} + \lambda^2\tilde{p}_{12}$, $\tilde{p}_{11} = p_{11} - p_{11d}$, $\tilde{p}_{12} = p_{12} - p_{12d}$, $\lambda^1 \in \mathbf{R}$, $\lambda^1 > 0$, $\lambda^2 \in \mathbf{R}$, $\lambda^2 > 0$, $k_{d11} \in \mathbf{R}$, $k_{d11} > 0$, and $v \in \mathbf{R}$ is the new input.

STEP 1

Let the signal $p_{22d}$ be chosen as follows:

$$p_{22d} = p_{12} + k^{-1}(c_{21}\dot{q}_{11r} + c_{22}\dot{q}_{12r} + d_{21}\ddot{q}_{11r} + d_{22}\ddot{q}_{12r} - k_{d22}s_{12}), \tag{24}$$

where $k_{d22} \in \mathbf{R}$, $k_{d22} > 0$. Consider that $p_{22}$ is the control input in eq. (22) and that the feedback law $p_{22}$ is equal to $p_{22d}$, where $p_{22d}$ is defined in (24). Equations (21)–(23) can be rewritten in the following form:

$$D(p_1)\dot{s}_1 + C(p_1, \dot{p}_1)s_1 + B_1 s_1 - K\tilde{p}_2 = 0 \tag{25}$$

$$\ddot{p}_{22} = v, \tag{26}$$

where

$$s_1 = [s_{11}, s_{12}]^T \quad \tilde{p}_{22} = p_{22} - p_{22d}.$$

For $\tilde{p}_{22} = 0$ (i.e., $p_{22} = p_{22d}$), the origin of the system described by eq. (25) is globally asymptotically stable. In fact, let the following Lyapunov function candidate be considered:

$$V_1 = \frac{1}{2} s_1^T D(p_1) s_1. \tag{27}$$

Taking into account that the matrix $\dot{D} - 2C$ is skew symmetric, the derivative of $V_1$ along the solutions of (25) is

$$\dot{V}_1 = -\frac{1}{2} s_1^T B_1 s_1. \tag{28}$$

Now, since $p_{22}$ is not the input, there is an error that is taken into account by the following integrator:

$$\dot{\tilde{p}}_{22} = \dot{p}_{22} - \dot{p}_{22d}, \tag{29}$$

where the signal $p_{22d}$ is computable from position and velocity measurements only.

STEP 2

Consider that $\dot{p}_{22}$ is the control input in eq. (29). Let a Lyapunov function candidate of the system described by eqs. (25) and (29) be defined as follows:

$$V_2 = V_1 + \frac{1}{2} \tilde{p}_{22}. \tag{30}$$

The derivative of $V_2$ along the solutions of (25) and (29) takes the following form:

$$\dot{V}_2 = -\frac{1}{2} s_1^T B_1 s_1 + K s_{12} \tilde{p}_{22} + \tilde{p}_{22} \dot{\tilde{p}}_{22}. \tag{31}$$

To have $\dot{V}_2$ negative definite, $\dot{p}_{22}$ takes the following form:

$$\dot{p}_{22} = -\lambda_2 \tilde{p}_{22} - k s_{12} + \dot{p}_{22d}, \tag{32}$$

where $\lambda_2 \in \mathbf{R}$, $\lambda_2 > 0$. Since $\dot{p}_{22}$ is not the input, the error $\tilde{e}_2 = \dot{p}_{22} - e_{2d}$, where $e_{2d} = -\lambda_2 \tilde{p}_{22} - k s_{12} + \dot{e}_{22d}$ is introduced. Adding an integrator, the overall error equations can be written as follows:

$$D\dot{s}_1 + Cs_1 + B_1 s_1 = K\tilde{p}_2 \tag{33}$$

$$\dot{\tilde{p}}_{22} = \tilde{e}_2 - \lambda_2 \tilde{p}_{22} - k s_{12} \tag{34}$$

$$\dot{\tilde{e}}_2 = -\dot{e}_{2d} + v, \tag{35}$$

where $\dot{e}_{2d}$ is computable from position and velocity measurements only.

STEP 3

Consider the following Lyapunov function candidate in order to study the origin of the error dynamics described by (33)–(35):

$$V_3 = V_2 + \frac{1}{2} \tilde{e}_2^2. \tag{36}$$

The derivative of $V_3$ along the solutions of (33)-(35) takes the following form:

$$\dot{V}_3 = -s_1^T B_1 s_1 - \lambda_2 \tilde{p}_{22}^2 + \tilde{e}_2 [\tilde{p}_{22} - \dot{e}_{2d} + v]. \tag{37}$$

To have the function $\dot{V}_3$ negative definite, the input $v$ is chosen as follows:

$$v = -\lambda_3 \tilde{e}_2 + \dot{e}_{2d} - \tilde{p}_{22}. \tag{38}$$

As is well known (Kanellakopoulos, Kokotovic, and Morse 1991; Brogliato, Ortega, and Lozano 1995), the global output tracking is solved by the following control laws:

$$\tau_1 = d_{11} \ddot{q}_{11r} + d_{12} \ddot{q}_{12r} + c_{11} \dot{q}_{11r} + c_{12} \dot{q}_{12r} - k_{d11} s_{11} \tag{39}$$

$$\tau_2 = -k(p_{12} - p_{22}) + J_{22}[\ddot{p}_{22d} - (1 + \lambda_2 \lambda_3)\tilde{p}_{22} - k(\lambda_3 s_{12} + \dot{s}_{12}) - \dot{\tilde{p}}_{22}(\lambda_2 + \lambda_3)]. \tag{40}$$

Following the same procedure as above, let us consider the following Lyapunov function candidates at Steps 1, 2, and 3, respectively:

$$V_{1b} = V_1$$

$$V_{2b} = V_{1b} + \frac{1}{2} k \tilde{p}_{22}^2$$

$$V_{3b} = V_{2b} + \frac{1}{2} k \tilde{e}_{2b}^2, \tag{41}$$

where $\tilde{e}_{2b} = \dot{p}_{22} - e_{2db}$ and $e_{2db} = -\lambda_2 \tilde{p}_{22} - s_{12} + \dot{p}_{22d}$, $\lambda_2 \in \mathbf{R}$, $\lambda_2 > 0$. The control laws that solve the global tracking problem take the following form:

$$\tau_1 = d_{11} \ddot{q}_{11r} + d_{12} \ddot{q}_{12r} + c_{11} \dot{q}_{11r} + c_{12} \dot{q}_{12r} - k_{d11} s_{11} \tag{42}$$

$$\tau_2 = -k(p_{12} - p_{22}) + J_{22}[\ddot{p}_{22d} - (\lambda_2 + \lambda_3)\dot{\tilde{p}}_{22} - (1 + \lambda_2 \lambda_3)\tilde{p}_{22} - \lambda_3 s_{12} - \dot{s}_{12}]. \tag{43}$$

*Controller 1*

Let the dynamical equations of the Capri robot described by (14) and (15) be considered. Let the signal $p_{22d}$, interpreted as a desired input to the link dynamics, be chosen as in (24), and the control $\tau_1$ as in (39). The control torque $\tau_2$ is chosen as follows:

$$\tau_2 = J_{22} \ddot{p}_{22d} - k_1 \tilde{p}_{22} - k_2 \dot{\tilde{p}}_{22} - k(p_{12} - p_{22}), \tag{44}$$

where $k_1$, $k_1 \in \mathbf{R}$, $k_2$, $k_2 > 0$. Equations (14) and (15) can be rewritten in the following form:

$$D\dot{s}_1 + Cs_1 + B_1 s_1 + K\tilde{p}_2 = 0 \tag{45}$$

$$J_{22} \ddot{\tilde{p}}_{22} + k_2 \dot{\tilde{p}}_{22} + k_1 \tilde{p}_{22} = 0. \tag{46}$$

Taking into account the results in Seibert and Suarez (1990) and Brogliato, Ortega, and Lozano (1995), it is possible to show that the origin of the system described by eqs. (45) and (46) is globally asymptotically stable; that is, the global output problem is solved by (24), (39), and (44).

*Controller 4*

To find a solution to the global output tracking, it is convenient to rewrite (14) and (15) in the following form:

$$\bar{D}\dot{s} + \left(\bar{C} + \bar{B}\right) s + \bar{K}\tilde{p} = \psi, \tag{47}$$

where

$$\bar{D} = \begin{bmatrix} D & 0 \\ 0 & J_{22} \end{bmatrix} \quad \bar{C} = \begin{bmatrix} C & 0 \\ 0 & 0 \end{bmatrix}$$

$$\bar{K} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & k & -k \\ 0 & -k & k \end{bmatrix} \bar{B} = \text{diag}\,[B_1, B_2]$$

$$= \begin{bmatrix} k_{d11} & 0 & 0 \\ 0 & k_{d22} & \\ 0 & 0 & k_{d33} \end{bmatrix}$$

$s = [s_1, s_{22}]^T$, $s_{22} = \dot{\tilde{p}}_{22} + \lambda^2 \tilde{p}_{22}$, $k_{d11}, k_{d22}, k_{d33} \in \mathbf{R}$, $k_{d11}, k_{d22}, k_{d33} > 0$, $\tilde{p} = [\tilde{p}_{11}, \tilde{p}_2]^T$ and

$$\psi = \begin{bmatrix} \tau_1 \\ 0 \\ \tau_2 \end{bmatrix} - \begin{bmatrix} d_{11}\ddot{q}_{11r} + d_{12}\ddot{q}_{12r} \\ +c_{11}\dot{q}_{11r} + c_{12}\dot{q}_{12r} \\ -k(p_{12d} - p_{22d}) + d_{21}\ddot{q}_{11r} \\ +d_{22}\ddot{q}_{12r} + c_{21}\dot{q}_{11r} + c_{22}\dot{q}_{12r} \\ J_{22}\ddot{q}_{22r} - k(p_{12d} - p_{22d}) \end{bmatrix} + \bar{B}s. \tag{48}$$

In Brogliato, Ortega, and Lozano (1995), it is shown that the solution of the equation $\psi \equiv 0$ defines a nonlinear state feedback globally tracking controller.

## Acknowledgments

The authors would like to thank A. Ibach and J. Pèrard for their help.

## End Note

A PD controller provides $L_\infty$ stability in case of tracking for sufficiently high feedback gains (Kawamura, Miyazaki, and Arimoto 1988). Our experimental results are in accordance with the fact that the stabilizing value of the gains may depend on the desired trajectory.

## References

Brogliato, B., and Lozano, R. 1996. Correction to "Adaptive Control of Robot Manipulators with Flexible Joints." *IEEE Trans. on Automatic Control* 41(6):920–921.

Brogliato B., Ortega R., and Lozano R. 1995. Global tracking controllers for flexible joint manipulators: A comparative study. *Automatica* 31(7):941–956.

Dawson, D. M., Caroll, J. J., and Schneider, M. 1994. Integrator backstepping control of a brush DC motor turning a robotic load. *IEEE Trans. on Control Systems Technology* 2(3):233–244.

Ghorbel, F., Fitzmorris, A., and Spong, M. W. 1990. Robustness of adaptive control of robot manipulators: Theory and experiments. *Advanced Robot Control* (Lecture Notes in Control and Information Sciences). London: Springer-Verlag.

Ghorbel, F., Hung, J. Y., and Spong, M. W. 1989 (December). Adaptive control of flexible joint robots. *IEEE Control System Magazine* 9(7):9–13.

Ghorbel, F., and Spong, M. W. 1992 (Nice, France, May). Adaptive composite control of robots with flexible joints. *IEEE Int. Conf. on Robotics and Automation.*

Kanellakopoulos, I., Kokotovic, P., and Morse, A. S. 1991. Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Trans. on Automatic Control* 36:1241–1253.

Kawamura, S., Miyazaki, F., and Arimoto, S. 1988. Is a local linear PD feedback control law effective for trajectory tracking of robot motion? *IEEE Int. Conf. on Robotics and Automation*, pp. 1335–1340.

Khorasani, K. 1992. Adaptive control of flexible-joint robots. *IEEE Trans. on Robotics and Automation* 8(2):250–267.

Lozano, R., and Brogliato, B. 1992. Adaptive control of robot manipulators with flexible joints. *IEEE Trans. on Automatic Control* 37(2):174–181.

Nicosia, S., and Tomei, P. 1992. A method to design adaptive controllers for flexible joint robots. *J. Robotic Systems* 10(6):835–846.

Ortega, R., and Espinosa, G. 1993. Torque regulation of induction motors. *Automatica* 29(3):621–633.

Seibert, P., and Suarez, R. 1990. Global stabilization of cascade systems. *Systems and Control Letters* 14(4):347–352.

Slotine, J. J., and Li, W. 1988. Adaptive manipulator control: A case study. *IEEE Trans. on Automatic Control* 33(11):995–1003.

Spong, M. W. 1987. Modeling and control of elastic joint robots. *ASME J. Dyn. Sys. Meas. Control* 109:310–319.

Spong, M. W. 1989. Adaptive control of flexible joint manipulators. *Systems and Control Letters* 13:15–21.

Spong, M. W. 1990. Control of flexible-joint robots: A survey. *New Trends and Applications of Distributed Parameter Control Systems* (Lecture Notes in Pure and Applied Mathematics), edited by G. Chen, E. B Lee, W. Littman, and L. Markus. New York: Marcel Dekker.

Spong, M. W. 1995. Adaptive control of flexible joint manipulators: Comments on two papers. *Automatica* 31(4):585–590.

Takegaki, M., and Arimoto, S. 1981. A new feedback method for dynamic control of manipulators. *ASME J. Dyn. Sys. Meas. Control* 102:119–125.

Yoshikawa, T. 1990. *Foundations of Robotics. Analysis and Control.* Cambridge, MA: MIT Press.