Nonlinear and Markovian $\mathcal{H}_\infty$ Controls of Underactuated Manipulators

Adriano Almeida Gonçalves Siqueira and Marco Henrique Terra, *Member, IEEE*

***Abstract*—This paper develops procedures based on $\mathcal{H}_\infty$ techniques to achieve position control underactuated manipulators. First, in a deterministic approach, we compare two nonlinear $\mathcal{H}_\infty$ control techniques in order to verify the differences in structure and robustness of controllers based on quasi-linear parameter varying (LPV) representation and game theory. In this case, the use of brakes is necessary when a fault occurs. Secondly, we develop a procedure based on Markov theory, where an $\mathcal{H}_\infty$ Markovian controller is used to control the position of manipulators subject to abrupt changes in configuration. In this case, the brakes are not necessary to guarantee the stability of the system. The theory developed is tested in an actual manipulator setup.**

***Index Terms*—Fault tolerant control, linear parameter varying (LPV) representation, Markovian control, nonlinear $\mathcal{H}_\infty$ control, underactuated manipulator.**

I. Introduction

PARAMETRIC uncertainties and exogenous disturbances increase the difficulty of reference tracking control for robotic manipulators. Additionally, actuator failures can suddenly occur during the manipulator motion. If the robot is working in hazardous or unstructured environments, where any nonplanned motion can be critical, position control after the failure is of utmost importance. Among the possible actuator failure modes, the free torque one, where the torque supplied to the motors of one or more joints vanishes suddenly, can destabilize the system with the possibility of damaging the manipulator components. When a free torque fault occurs the fully actuated manipulator becomes an underactuated one. This kind of mechanical system, with less actuators than degrees of freedom, has been studied by several researchers [1]–[4], [6], [13], [20]. A position control strategy for underactuated manipulator was first proposed in [1]. According to that strategy, first, all the passive joints (those with faulty actuators) are controlled to the desired final position and locked in place. Then, with the passive joints fixed, the active ones (those with working actuators) are controlled. In [6], three possibilities of selecting the joints to be controlled in each phase are proposed. In the previous references, the control strategies were based on feedback linearization approach.

In this paper, we develop three $\mathcal{H}_\infty$ control strategies for underactuated manipulators. First, we compare two nonlinear $\mathcal{H}_\infty$ control techniques. Nonlinear $\mathcal{H}_\infty$ control consists in guaranteeing that the $\mathcal{L}_2$ gain between the disturbance and the output is bounded by an attenuation level γ. The recent development of the linear parameter varying (LPV) technique provides a systematic way to design controllers that schedule the varying parameters of the system and satisfy the $\mathcal{L}_2$ gain condition [22]. The nonlinear dynamics can be represented as an LPV system with the parameters as function of the state, namely, quasi-LPV representation. Based on the observation that this representation is not unique, in [14] it was proposed a scheme that incorporates the freedom to select the LPV representation into the optimization problem for $\mathcal{L}_2$ gain synthesis. A bilinear matrix inequality (BMI), which is not convex in the design variables, is needed to solve this state feedback problem, and there is no guarantee that the global optimum can be found. However, analogous to the $D-K$ iteration for μ synthesis [23], an iterative procedure can be used to avoid convexity problems. On the other hand, in [10] game theory was used to solve the $\mathcal{H}_\infty$ control problem for a fully actuated manipulator, using the state tracking error equation, proposed in [17]. An explicit global solution for this problem, formulated as a minimax game, was developed. The solution proposed in [10] was applied in a fully actuated experimental manipulator with high inertia in [19]. An adaptive nonlinear $\mathcal{H}_\infty$ control algorithm was proposed in [9] considering that unknown parameters can be learned by a classical adaptive update law. In [8], an adaptive neural network tracking control with a guaranteed $\mathcal{H}_\infty$ performance was developed; it does not require the knowledge of either the mathematical model or the parameterization of the manipulator. Similar results were obtained in [15] for position/force control of constrained manipulators.

Based on the scenario described above, in this paper, we adopt and compare two methodologies to control our experimental manipulator UArm II in the fully actuated and underactuated configurations: the $\mathcal{H}_\infty$ control via a quasi-LPV representation of the system [14] and the technique proposed in [10]. Since the $\mathcal{H}_\infty$ control problem via game theory in [10] applies only to fully actuated manipulators, we propose its extension to the underactuated case. This comparison study is motivated by the observation that these two classes of nonlinear $\mathcal{H}_\infty$ controllers are different in nature: the first technique, via quasi-LPV representation, provides a gain, after the solution of several coupled Riccati inequalities, that varies with time; the second technique provides a gain, via an analytical solution based on game theory, that is constant, similar to the results obtained with feedback linearization procedures. We investigate the robustness of both controllers.

Manuscript received October 15, 2002; revised July 21, 2003. Manuscript received in final form February 6, 2004. Recommended by Associate Editor K. Kozlowski. This work was supported by Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), Brazil, under Grant 00/00388-9.

The authors are with the Electrical Engineering Department,University of São Paulo at São Carlos, Cep. 13566-590, São Carlos, Brazil (e-mail: siqueira@sel.eesc.usp.br; terra@sel.eesc.usp.br).

Digital Object Identifier 10.1109/TCST.2004.833626

Even though the previous $\mathcal{H}_\infty$ controllers are designed to attenuate uncertainties and disturbances, the stability of the manipulator control, when the configuration is changed after a fault occurrence, is not guaranteed. To use these controllers in a fault tolerant robot system, it is necessary to use brakes after the fault detection, restarting the movement from zero velocity.

The third controller is designed to avoid the necessity of utilizing breaks when a fault occurs. We develop, via Markov theory, a procedure to incorporate abrupt changes in the manipulator configuration, and Markovian controllers are used to guarantee stability. The Markov theory is based on stochastic linear systems subject to abrupt variations, namely, Markovian jump linear systems (MJLSs) [16]. The nonlinear system is linearized around operation points, and a Markovian model is developed regarding the changes at the operation points and the probability of a fault. The control strategy is developed based on the fact that, with deterministic controllers, there do not exist guarantees of stability when the system is subject to abrupt changes. The fault tolerant control system for manipulators developed in this paper utilizes the Markovian $\mathcal{H}_\infty$ control of [11].

This paper is organized as follows. In Sections II and III, the quasi-LPV representations of the fully actuated and of the underactuated manipulator are presented; the $\mathcal{H}_\infty$ control via a quasi-LPV representation is presented in Section IV; in Section V, it is presented the solution of the $\mathcal{H}_\infty$ control problem for a fully actuated manipulator proposed in [10] and its extension to the underactuated case; results using the two techniques, obtained with the experimental manipulator UArm II, are presented in Section VI; finally, the fault tolerant system developed via Markov theory is presented in Section VII.

We denote by $\mathcal{C}^1(\Re^n, \Re^m)$ the set of continuously differentiable functions that map $\Re^n$ to $\Re^m$, and $\mathcal{S}$ as the set of symmetric matrices. The Euclidean norm of a vector is denoted by $\|\cdot\|$, i.e., $\|z\|^2 = z^T z$ for $z \in \Re^k$. A matrix C is said to be skew-symmetric when $C = -C^T$. The notation $\mathcal{L}_2$ will be used for bounded energy signals, i.e., $\mathcal{L}_2(0,T) = \left\{ \mathrm{w} : \int_0^T \|\mathrm{w(t)}\|^2 \,\mathrm{dt} < \infty \right\}$.

II. Quasi-LPV Representation of a Manipulator

The dynamic equations of a robot manipulator can be found with Lagrange theory as

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) \tag{1}$$

where $q \in \Re^n$ is the joint position vector, $M(q) \in \Re^{n\times n}$ is the symmetric positive definite inertia matrix, $C(q,\dot{q}) \in \Re^{n\times n}$ is the Coriolis and centripetal matrix, $F(\dot{q}) \in \Re^n$ is the frictional torque vector, $G(q) \in \Re^n$ is the gravitational torque vector, and $\tau \in \Re^n$ is the torque vector. Parametric uncertainties can be introduced dividing the parameter matrices $M(q)$, $C(q,\dot{q})$, $F(\dot{q})$, and $G(q)$ into a nominal and a perturbed part

$$\begin{aligned} M(q) &= M_0(q) + \Delta M(q) \\ C(q,\dot{q}) &= C_0(q,\dot{q}) + \Delta C(q,\dot{q}) \\ F(\dot{q}) &= F_0(\dot{q}) + \Delta F(\dot{q}) \\ G(q) &= G_0(q) + \Delta G(q) \end{aligned}$$

where $M_0(q)$, $C_0(q,\dot{q})$, $F_0(\dot{q})$, and $G_0(q)$ are the nominal matrices and $\Delta M(q)$, $\Delta C(q,\dot{q})$, $\Delta F(\dot{q})$, and $\Delta G(q)$ are the parametric uncertainties. According to the robotics literature, the matrix $C_0(q,\dot{q}) - (1/2)\dot{M}_0(q,\dot{q})$ is skew-symmetric.

A finite energy exogenous disturbance, τ_d, can also be introduced. After these considerations (1) becomes

$$\tau + \delta(q,\dot{q},\ddot{q}) = M_0(q)\ddot{q} + C_0(q,\dot{q})\dot{q} + F_0(\dot{q}) + G_0(q) \tag{2}$$

with

$$\delta(q,\dot{q},\ddot{q}) = -(\Delta M(q)\ddot{q} + \Delta C(q,\dot{q})\dot{q} + \Delta F(\dot{q}) + \Delta G(q) - \tau_d).$$

The state tracking error is defined as

$$\tilde{x} = \begin{bmatrix} \dot{q} - \dot{q}^d \\ q - q^d \end{bmatrix} = \begin{bmatrix} \dot{\tilde{q}} \\ \tilde{q} \end{bmatrix} \tag{3}$$

where q^d and $\dot{q}^d \in \Re^n$ are the desired reference trajectory and the corresponding velocity, respectively. The variables q^d, $\dot{q}^d$, and $\ddot{q}^d$, the desired acceleration, are assumed to be within the physical and kinematics limits of the manipulator.

The dynamic equation for the state tracking error is given from (2) and (3) as

$$\dot{\tilde{x}} = A(q,\dot{q})\tilde{x} + Bu + Bw \tag{4}$$

with

$$\begin{aligned} A(q,\dot{q}) &= \begin{bmatrix} -M_0^{-1}(q)C_0(q,\dot{q}) & 0 \\ I_n & 0 \end{bmatrix} \\ B &= \begin{bmatrix} I_n \\ 0 \end{bmatrix} \\ w &= M_0^{-1}(q)\delta(q,\dot{q},\ddot{q}) \\ u &= M_0^{-1}(q)(\tau - M_0(q)\ddot{q}^d - C_0(q,\dot{q})\dot{q}^d \\ &\quad - F_0(\dot{q}) - G_0(q)). \end{aligned}$$

The applied torque is given by

$$\tau = M_0(q)(\ddot{q}^d + u) + C_0(q,\dot{q})\dot{q}^d + F_0(\dot{q}) + G_0(q).$$

Although the matrix $M_0(q)$ explicitly depends on the joint positions, we can consider it as function of the position error [17]

$$M_0(q) = M_0(\tilde{q} + q^d) = M_0(\tilde{x}, t).$$

The same can be observed for $C_0(q,\dot{q})$. Hence, (4) is actually a quasi-LPV representation for the robot manipulator, i.e., with $A(\tilde{x},t)$.

III. Underactuated Manipulator

Underactuated robot manipulators are mechanical systems with less actuators than degrees of freedom. For this reason, the position control of the passive joints is performed based on the dynamic coupling between them and the active joints [1]. Here, we consider that the passive joints are equipped with brakes. The strategy consists in controlling all the passive joints to reach their desired position, applying torques in the active ones, and

then turning on the brakes. After that, all the active joints are controlled as if the manipulator were fully actuated.

Consider a manipulator with n joints, of which n_p are passive and n_a are active joints. It is known [1] that, using breaks, no more than n_a joints of the manipulator can be controlled at any instant. The n_a joints to be controlled are grouped in the vector $q_c \in \Re^{n_a}$. The remaining joints are grouped in the vector $q_r \in \Re^{n-n_a}$. There are three possibilities of forming the vector q_c [6]:

1) q_c contains only passive joints: when $n_p \geq n_a$ and all other passive joints, if any, are kept locked;
2) q_c contains passive and active joints: all other passive joints, if any, are kept locked;
3) q_c contains only active joints.

With these possibilities in mind we can define the control strategy. First, choose the vector q_c as the possibilities 1 or 2 (according to n_p), until all passive joints have reached the desired position. Second, choose q_c as the possibility 3, and control the active joints to the desired position.

The dynamic (2) can now be partitioned as

$$\begin{bmatrix} \tau_a \\ 0 \end{bmatrix} + \begin{bmatrix} \delta_a(q,\dot{q},\ddot{q}) \\ \delta_u(q,\dot{q},\ddot{q}) \end{bmatrix} = \begin{bmatrix} M_{ar}(q) & M_{ac}(q) \\ M_{ur}(q) & M_{uc}(q) \end{bmatrix} \begin{bmatrix} \ddot{q}_r \\ \ddot{q}_c \end{bmatrix} + \begin{bmatrix} C_{ar}(q,\dot{q}) & C_{ac}(q,\dot{q}) \\ C_{ur}(q,\dot{q}) & C_{uc}(q,\dot{q}) \end{bmatrix} \begin{bmatrix} \dot{q}_r \\ \dot{q}_c \end{bmatrix} + \begin{bmatrix} F_a(\dot{q}) \\ F_u(\dot{q}) \end{bmatrix} + \begin{bmatrix} G_a(q) \\ G_u(q) \end{bmatrix} \quad (5)$$

where the indexes a and u represent the active and free (unlocked) passive joints, respectively. Factoring out $\ddot{q}_r$ in the second line of (5) and substituting the result back on its first line, we obtain

$$\tau_a + \overline{\delta}(q,\dot{q},\ddot{q}) = \overline{M}_0(q)\ddot{q}_c + \overline{C}_0(q,\dot{q})\dot{q}_c + \overline{D}_0(q,\dot{q})\dot{q}_r + \overline{F}_0(q,\dot{q}) + \overline{G}_0(q)$$

with

$$\begin{aligned} \overline{M}_0(q) &= M_{ac}(q) - M_{ar}(q)M_{ur}^{-1}(q)M_{uc}(q) \\ \overline{C}_0(q,\dot{q}) &= C_{ac}(q,\dot{q}) - M_{ar}(q)M_{ur}^{-1}(q)C_{uc}(q,\dot{q}) \\ \overline{D}_0(q,\dot{q}) &= C_{ar}(q,\dot{q}) - M_{ar}(q)M_{ur}^{-1}(q)C_{ur}(q,\dot{q}) \\ \overline{F}_0(q,\dot{q}) &= F_a(\dot{q}) - M_{ar}(q)M_{ur}^{-1}(q)F_u(\dot{q}) \\ \overline{G}_0(q) &= G_a(q) - M_{ar}(q)M_{ur}^{-1}(q)G_u(q) \\ \overline{\delta}(q,\dot{q},\ddot{q}) &= \delta_a(q,\dot{q},\ddot{q}) - M_{ar}(q)M_{ur}^{-1}(q)\delta_u(q,\dot{q},\ddot{q}). \end{aligned}$$

The state tracking error is defined as

$$\tilde{x}_c = \begin{bmatrix} \dot{q}_c - \dot{q}_c^d \\ q_c - q_c^d \end{bmatrix} = \begin{bmatrix} \dot{\tilde{q}}_c \\ \tilde{q}_c \end{bmatrix} \quad (6)$$

where q_c^d and $\dot{q}_c^d \in \Re_a^n$ are the desired reference trajectory and the corresponding velocity for the controlled joints, respectively. There is no reference trajectory for the remaining joints.

Hence, a quasi-LPV representation of the underactuated manipulator can be defined as follows:

$$\dot{\tilde{x}}_c = \overline{A}(q,\dot{q})\tilde{x}_c + \overline{B}\overline{u} + \overline{B}\overline{w} \quad (7)$$

with

$$\overline{A}(q,\dot{q}) = \begin{bmatrix} -\overline{M}_0^{-1}(q)\overline{C}_0(q,\dot{q}) & 0 \\ I_{n_a} & 0 \end{bmatrix}$$

$$\begin{aligned} \overline{B} &= \begin{bmatrix} I_{n_a} \\ 0 \end{bmatrix} \\ \overline{w} &= \overline{M}_0^{-1}(q)\overline{\delta}(q,\dot{q},\ddot{q}) \\ \overline{u} &= \overline{M}_0^{-1}(q)(\tau_a - \overline{M}_0(q)\ddot{q}_c^d - \overline{C}_0(q,\dot{q})\dot{q}_c^d \\ &\quad - \overline{D}_0(q,\dot{q})\dot{q}_r - \overline{F}_0(\dot{q}) - \overline{G}_0(q)). \end{aligned}$$

From (7), the applied torque can be given by

$$\tau_a = \overline{M}_0(q)\ddot{q}_c^d + \overline{C}_0(q,\dot{q})\dot{q}_c^d + \overline{D}_0(q,\dot{q})\dot{q}_r + \overline{F}_0(\dot{q}) + \overline{G}_0(q) + \overline{M}_0(q)\overline{u}. \quad (8)$$

Again, although the matrix $\overline{M}_0(q)$ explicitly depends on the joint positions, we can consider it as function of the position error $\overline{M}_0(q) = \overline{M}_0(q_c, q_r) = \overline{M}_0(\tilde{q}_c + q_c^d(t), q_r) = \overline{M}_0(\tilde{x}_c, q_r, t)$. The same can be considered for $\overline{C}_0(q,\dot{q})$. In this case, $\overline{C}_0(q,\dot{q})$ is also function of $\dot{q}_r$. Hence, (7) is actually a quasi-LPV representation for the robot manipulator, i.e., with $\overline{A}(\tilde{x}_c, q_r, \dot{q}_r, t)$.

IV. Nonlinear $\mathcal{H}_\infty$ Control via Quasi-LPV Representation

Consider a nonlinear system with input signal $w \in \Re^p$ and output variables $z \in \Re^q$

$$\begin{aligned} \dot{x} &= f(x) + g(x)w \\ z &= h(x) + k(x)w \end{aligned} \quad (9)$$

where $f(0) = 0$ and $h(0) = 0$, and $x \in \Re^n$ are state variables. Assume that $f(\cdot)$, $g(\cdot)$, and $h(\cdot)$ are continuously differentiable functions. System (9) has an $\mathcal{L}_2$ gain $\leq \gamma$ if

$$\int_0^T \|z(t)\|^2 \, dt \leq \gamma^2 \int_0^T \|w(t)\|^2 \, dt \quad (10)$$

for all $T \geq 0$ and all $w \in \mathcal{L}_2(0, \mathrm{T})$ with the system starting from $x(0) = 0$. The nonlinear dynamics, (9), is parameterized in a state-dependent representation

$$\dot{x} = A(x)x + g(x)w$$

where $A(x) = A_0(x) + N(x)$ with $A_0(x)$ satisfying $f(x) = A_0(x)x$ and $N(x)x = 0$. There exists an infinite number of possible matrix representations $A(x)$ for a given $f(x)$. To keep the number of varying parameters in the state matrices to a minimum, the dependence of the state matrices $A(x)$ and $g(x)$ on the state variables will be changed to $\rho(x) \in \mathcal{C}^1(\Re^n, \Re^m)$ with $m < n$, i.e., $A(x)$ and $g(x)$ become $A(\rho(x))$ and $g(\rho(x))$, respectively. For instance, $\rho(x)$ can simply represent part of the state variables. Assume that the underlying parameter $\rho(x)$ varies in the allowable set

$$F_P^\nu = \left\{ \rho \in \mathcal{C}^1(\Re^n, \Re^m) : \rho(x) \in P \right\}$$

where $P \subset \Re^m$ is a compact set and $\underline{\nu}_i(\rho) \leq \dot{\rho}_i \leq \overline{\nu}_i(\rho)$, $i = 1, \ldots, m$. Note that the bounds on the rate variation of the parameter are allowed to vary with the parameter value.

The following quasi-LPV representation of the nonlinear system will be used for the synthesis problem:

$$\begin{bmatrix} \dot{x} \\ z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} A(\rho(x)) & B_1(\rho(x)) & B_2(\rho(x)) \\ C_1(\rho(x)) & D_{11}(\rho(x)) & D_{12}(\rho(x)) \\ C_2(\rho(x)) & D_{21}(\rho(x)) & D_{22}(\rho(x)) \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix} \tag{11}$$

where $u \in \Re^s$ is the control input. For the sake of simplicity, the following standard assumptions are considered in the above model: the terms $D_{11}(\rho(x))$, $D_{12}(\rho(x))$, and $D_{21}(\rho(x))$ are assumed to be zero and the term $D_{22}(\rho(x))$ is scaled to the identity matrix [14]. We want to find a continuous function $F(\rho(x))$ such that the closed-loop system has an $\mathcal{L}_2$ gain less than or equal to γ under a state feedback law $u = F(\rho(x))x$.

As stated above, we have an infinite number of choices when constructing the quasi-LPV model. In the following theorem, proposed in [14], this degree of freedom is used to shape the control system performance.

Theorem IV.1: If there exist continuously differentiable matrix functions $X(\rho(x)) > 0$ and $N(\rho(x))$ with $N(\rho(x))x = 0$ that satisfy

$$\begin{bmatrix} E(\rho(x)) & X(\rho(x))C_1^T(\rho(x)) & B_1(\rho(x)) \\ C_1(\rho(x))X(\rho(x)) & -I & 0 \\ B_1^T(\rho(x)) & 0 & -\gamma^2 I \end{bmatrix} < 0 \tag{12}$$

where

$$\begin{aligned} E(\rho(x)) = &-\sum_{i=1}^{m} \underline{\nu}_i(\rho) \frac{\partial X(\rho(x))}{\partial \rho_i} - B_2(\rho(x))B_2^T(\rho(x)) \\ &+ (\widehat{A}(\rho(x)) + N(\rho(x)))X(\rho(x)) \\ &+ X(\rho(x))(\widehat{A}(\rho(x)) + N(\rho(x)))^T \end{aligned}$$

and $\widehat{A}(\rho(x)) = A(\rho(x)) - B_2(\rho(x))C_2(\rho(x))$ for all $\rho(x) \in P$, then the closed-loop system (11) has $\mathcal{L}_2$ gain $\leq \gamma$ under the state feedback law

$$u = -(B_2(\rho(x))X^{-1}(\rho(x)) + C_2(\rho(x)))x. \tag{13}$$

The notation $\sum_{i=1}^{m} \underline{\nu}_i(\rho)$ represents that every combination of $\overline{\nu}_i(\rho)$ and $\underline{\nu}_i(\rho)$ should be included in the inequality. Hence, (12) actually represents 2^m inequalities.

The constraint (12) is not affine in $X(\rho(x))$ and $N(\rho(x))$. For any value of $\rho(x)$, (12) is a bilinear matrix inequality (BMI) in $X(\rho(x))$ and $N(\rho(x))$. Hence, minimizing γ over both $X(\rho(x))$ and $N(\rho(x))$ under the BMI is not a convex optimization problem. However, when we fix $X(\rho(x))$ or $N(\rho(x))$ and try to minimize γ over the not fixed variable, it is a convex problem. Similar to the $D-K$ iteration for μ synthesis, an $X-N$ iteration is proposed in [14] as a practical scheme to achieve the best closed-loop performance.

1) Calculate the best achievable γ level for the current quasi-LPV model (Theorem IV.1 considering $N(\rho(x)) = 0$). If the performance is satisfactory, stop. If not, go to the next step.
2) Calculate the best γ over all $N(\rho(x))$ with $X(\rho(x))$ fixed from the previous step (Theorem IV.1). If the new γ is satisfactory or unchanged, stop. If not, go to the next step.
3) Replace $A(\rho(x))$ with $A(\rho(x)) + N(\rho(x))$. Go to step 1.

Note that this strategy is not guaranteed to converge to the global optimum because it is not convex in both $X(\rho(x))$ and $N(\rho(x))$. The BMI (12) is reduced to a linear matrix inequality (LMI) in $X(\rho(x))$ in step 1 and in $N(\rho(x))$ in step 2. One of the difficulties when applying Theorem IV.1 is to determine the bound of the variation rate of $\rho(x)$ since it is no longer known *a priori* to us. A practical approach is to use the best estimate of the operating range of the selected varying parameters, and an approximation on the parameter variation rates. These assumptions should be verified by simulations after the controller is designed.

A practical scheme using basis functions for $X(\rho(x))$ and $N(\rho(x))$ and gridding the parameter set P was developed in [14] to solve constraint (12). First select a set of $\mathcal{C}^1$ functions $\{f_i(\rho)\}_{i=1}^{M}$ as a basis for $X(\rho(x))$

$$X(\rho(x)) = \sum_{i=1}^{M} f_i(\rho(x))X_i$$

where $X_i \in \mathcal{S}^{n\times n}$ is the coefficient matrix for $f_i(\rho(x))$. For the state-dependent matrix $N(\rho(x))$, the basis functions should be selected so that the condition $N(\rho(x))x = 0$ is satisfied. For example, if the state vector contains four elements, i.e., $x = [x_1\ x_2\ x_3\ x_4]^T$, the matrix $N(\rho(x))$ can be expressed in terms of $\mathcal{C}^1$ functions $\{g_i(\rho)\}_{i=1}^{Q}$ as

$$N(\rho(x)) = \sum_{i=1}^{Q} (g_i(\rho(x))N_i)\begin{bmatrix} -x_4 & -x_3 & x_2 & x_1 \end{bmatrix}$$

where $N_i \in \Re^n$.

If the matrices $X(\rho(x))$ and $N(\rho(x))$, as given above, are substituted in (12), the constraint becomes a BMI in terms of the matrix variables $\{X_i\}_{i=1}^{M}$ and $\{N_i\}_{i=1}^{Q}$ when the parameter ρ is fixed. To solve this infinite dimensional optimization problem, we can grid the parameter set P in L points $\{\rho_k\}_{k=1}^{L}$ in each dimension. Since (12) consists of 2^m constraints, a total of $(2^m+1)L^m$ matrix inequalities in terms of the matrices X_i and N_i have to be solved in steps 1 and 2, respectively.

V. Nonlinear $\mathcal{H}_\infty$ Control via Game Theory

In this section, we utilize game theory to solve the $\mathcal{H}_\infty$ control problem of a robot manipulator derived from (4) after the state transformation given by [10], [17]

$$\tilde{z} = \begin{bmatrix} \tilde{z}_1 \\ \tilde{z}_2 \end{bmatrix} = T_0\tilde{x} = \begin{bmatrix} T_{11} & T_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{\tilde{q}} \\ \tilde{q} \end{bmatrix} \tag{14}$$

where $T_{11}, T_{12} \in \Re^{n\times n}$ are constant matrices to be determined.

The dynamic equation of the state tracking error becomes

$$\dot{\tilde{x}} = A_T(\tilde{x},t)\tilde{x} + B_T(\tilde{x},t)u + B_T(\tilde{x},t)w \tag{15}$$

with

$$\begin{aligned} A_T(\tilde{x},t) &= T_0^{-1}\begin{bmatrix} -M_0^{-1}(q)C_0(q,\dot{q}) & 0 \\ T_{11}^{-1} & -T_{11}^{-1}T_{12} \end{bmatrix} T_0 \\ B_T(\tilde{x},t) &= T_0^{-1}\begin{bmatrix} M_0^{-1}(q) \\ 0 \end{bmatrix} \\ w &= M_0(q)T_{11}M_0^{-1}(q)\delta(q,\dot{q},\ddot{q}) \end{aligned}$$

$$u = M_0(q)T_1\dot{\tilde{x}} + C_0(q,\dot{q})T_1\tilde{x}.$$

The relationship between the applied torques and the control input is given by

$$\tau = M_0(q)\ddot{q} + C_0(q,\dot{q})\dot{q} + F_0(\dot{q}) + G_0(q) \tag{16}$$

with

$$\ddot{q} = \ddot{q}^d - T_{11}^{-1}T_{12}\dot{\tilde{x}} - T_{11}^{-1}M_0^{-1}(q)\left(C_0(q,\dot{q})B^TT_0\tilde{x} - u\right). \tag{17}$$

The $\mathcal{H}_\infty$ control problem for a manipulator seeks the attenuation of the effects of the disturbance w in the system by a state feedback control strategy of the form $u = F(\tilde{x})\tilde{x}$. With this intention, and subject to the tracking error dynamics, the following performance criterion, including a desired disturbance attenuation level γ, is proposed in [10]:

$$\min_{u(\cdot)\in\mathcal{L}_2}\max_{0\neq w(\cdot)\in\mathcal{L}_2}\frac{\int_0^\infty\left(\frac{1}{2}\tilde{x}^T(t)Q\tilde{x}(t)+\frac{1}{2}u^T(t)Ru(t)\right)dt}{\int_0^\infty\left(\frac{1}{2}w^T(t)w(t)\right)dt}\leq\gamma^2 \tag{18}$$

where Q and R are positive definite symmetric weighting matrices and $\tilde{x}(0) = 0$.

This performance criterion is actually the $\mathcal{L}_2$ gain (10) represented as a *minimax* problem with weighting matrices introduced in the output and in the control input terms.

According to game theory, the solution of this *minimax* problem is found if there exists a continuously differentiable Lyapunov function $V(\tilde{x},t)$ that satisfies the following Bellman–Isaacs equation (for further details, see [5]):

$$-\frac{\partial V(\tilde{x},t)}{\partial t} = \min_{u(\cdot)}\max_{w(\cdot)}\left\{L(\tilde{x},u,w) + \left(\frac{\partial V(\tilde{x},t)}{\partial\tilde{x}}\right)^T\tilde{x}\right\}$$

with terminal condition $V(\tilde{x}(\infty),\infty) = 0$ and

$$L(\tilde{x},u,w) = \frac{1}{2}\tilde{x}^T(t)Q\tilde{x}(t) + \frac{1}{2}u^T(t)Ru(t) - \frac{1}{2}\gamma^2w^T(t)w(t).$$

Selecting the Lyapunov function of the form

$$V(\tilde{x},t) = \frac{1}{2}\tilde{x}^TP(\tilde{x},t)\tilde{x} \tag{19}$$

where $P(\tilde{x},t)$ is a positive definite symmetric matrix for all $\tilde{x}$ and t, the Bellman–Isaacs equation is then changed to the following Riccati equation:

$$\dot{P}(\tilde{x},t) + P(\tilde{x},t)A_T(\tilde{x},t) + A_T(\tilde{x},t)P(\tilde{x},t) + P(\tilde{x},t)B_T(\tilde{x},t)\left(R^{-1} - \frac{1}{\gamma^2}I\right)B_T(\tilde{x},t)P(\tilde{x},t) + Q = 0.$$

With an appropriate choice of the matrix $P(\tilde{x},t)$ and by use of the skew-symmetric matrix $C_0(q,\dot{q}) - (1/2)\dot{M}_0(q,\dot{q})$, the Riccati equation can be simplified to an algebraic matrix equation. The matrix $P(\tilde{x},t)$ chosen by [10], [17] is

$$P(\tilde{x},t) = T_0^T\begin{bmatrix}M_0(\tilde{x},t) & 0\\ 0 & K\end{bmatrix}T_0 \tag{20}$$

where K is a positive definite symmetric constant matrix. The simplified algebraic equation is given as

$$\begin{bmatrix}0 & K\\ K & 0\end{bmatrix} - T_0^TB\left(R^{-1} - \frac{1}{\gamma^2}I\right)B^TT_0 + Q = 0. \tag{21}$$

The optimal control is given as

$$u^* = -R^{-1}B^TT_0\tilde{x}. \tag{22}$$

One can note that the resulting control input u is actually a static gain; this can be a disadvantage in comparison with the nonlinear gain of the quasi-LPV control input, (13). The terminal condition is satisfied for this choice of $P(\tilde{x},t)$ [10]. Then, to solve the $\mathcal{H}_\infty$ problem for a manipulator, we should find matrices K and T_0, which solve (21) that are defined as follows:

$$T_0 = \begin{bmatrix}R_1^TQ_1 & R_1^TQ_2\\ 0 & I\end{bmatrix} \tag{23}$$

and

$$K = \frac{1}{2}\left(Q_1^TQ_2 + Q_2^TQ_1\right) - \frac{1}{2}\left(Q_{21}^T + Q_{12}\right)$$

with the conditions $K > 0$ and $R < \gamma^2I$. Matrix R_1 is the result of the Cholesky factorization

$$R_1^TR_1 = \left(R^{-1} - \frac{1}{\gamma^2}I\right)^{-1} \tag{24}$$

and the positive definite symmetric matrix Q is factored as

$$Q = \begin{bmatrix}Q_1^TQ_1 & Q_{12}\\ Q_{12}^T & Q_2^TQ_2\end{bmatrix}. \tag{25}$$

Some remarks can be made about the selection of the weighting matrices Q_1, Q_2, and R and the attenuation level γ.

- There exists a compromise between the parameter γ and the weighting matrix R: first, R is selected and then γ is adjusted according to the constraint $R < \gamma^2I$.
- If γ increases ($\gamma\to\infty$), the resulting controller approximates the $\mathcal{H}_2$ controller, which does not guarantee disturbance attenuation.
- In order to satisfy the conditions $K > 0$ and $R < \gamma^2I$, and to obtain feasible solution to the Cholesky factorization, it is easier to select the weighting matrices Q_1, Q_2, and R as diagonal matrices, and $Q_{12} = 0$. Let $Q_1 = q_1I_n$, $Q_2 = q_2I_n$, and $R = rI_n$. With these choices and considering (16), (17), and (22), the applied torque becomes

$$\tau = M_0(q)\left(\ddot{q}^d - \frac{q_2}{q_1}\dot{\tilde{x}}_2\right) + C_0(q,\dot{q})\left(\dot{q}^d - \frac{q_2}{q_1}\tilde{x}_2\right) + F_0(\dot{q}) + G_0(q) - \frac{1}{r}\left[I_n\ \ \frac{q_2}{q_1}I_n\right]\tilde{x}. \tag{26}$$

- If off-diagonal elements are added in matrix Q_1, even when Q_1 remains positive definite, matrix T_{11}^{-1} may contain negative elements. In this case, positive feedback may appear in the control law, (16), (17), and (22).

In practice, these weighting matrices were selected heuristically, testing each new controller in the manipulator.

A. Underactuated Manipulators

In this section, we will extend the result of the last section to underactuated manipulators. For this, a new partition of (2) is used

$$\begin{bmatrix}\tau_r\\ \tau_c\end{bmatrix}+\begin{bmatrix}\delta_r(q,\dot q,\ddot q)\\ \delta_c(q,\dot q,\ddot q)\end{bmatrix}=\begin{bmatrix}M_{rr}(q) & M_{rc}(q)\\ M_{cr}(q) & M_{cc}(q)\end{bmatrix}\begin{bmatrix}\ddot q_r\\ \ddot q_c\end{bmatrix}$$
$$+\begin{bmatrix}C_{rr}(q,\dot q) & C_{rc}(q,\dot q)\\ C_{cr}(q,\dot q) & C_{cc}(q,\dot q)\end{bmatrix}\begin{bmatrix}\dot q_r\\ \dot q_c\end{bmatrix}+\begin{bmatrix}F_r(\dot q)\\ F_c(\dot q)\end{bmatrix}+\begin{bmatrix}G_r(q)\\ G_c(q)\end{bmatrix} \quad (27)$$

where τ_c are the controlled joint torques and τ_r are the remaining joint torques. For simplicity, the index 0 representing the nominal system is not shown.

Consider the following state transformation:

$$\tilde z=\begin{bmatrix}\tilde z_1\\ \tilde z_2\end{bmatrix}=T_0\tilde x_c=\begin{bmatrix}T_{11} & T_{12}\\ 0 & I\end{bmatrix}\begin{bmatrix}\dot{\tilde q}_c\\ \tilde q_c\end{bmatrix} \quad (28)$$

where $T_{11},T_{12}\in\Re^{n\times n}$ are constant matrices to be determined. Then the control input can be selected as

$$\overline u=M_{cc}(q)T_1\dot{\tilde x}_c+C_{cc}(q,\dot q)T_1\tilde x_c. \quad (29)$$

Considering the second line of (27) and the state tracking error (3), a state transformation similar to (28) can be used to generate the following dynamic equation of an underactuated manipulator:

$$\dot{\tilde x}_c=\overline A_T(\tilde x_c,t)\tilde x_c+\overline B_T(\tilde x_c,t)\overline u+\overline B_T(\tilde x_c,t)\overline w \quad (30)$$

with

$$\overline A_T(\tilde x_c,t)=T_0^{-1}\begin{bmatrix}-M_{cc}^{-1}(q)C_{cc}(q,\dot q) & 0\\ T_{11}^{-1} & -T_{11}^{-1}T_{12}\end{bmatrix}T_0$$
$$\overline B_T(\tilde x_c,t)=T_0^{-1}\begin{bmatrix}M_{cc}^{-1}(q)\\ 0\end{bmatrix}$$
$$\overline w=M_{cc}(q)T_{11}M_{cc}^{-1}(q)\delta_c(q,\dot q,\ddot q).$$

Note that we are considering $M_{cc}(q)$ in terms of controlled positions, and in consequence the matrix $A_T(\tilde x_c,t)$ is given in terms of controlled positions. This is a realistic assumption, as demonstrated in the experimental results in the next section.

From (29), the control acceleration is given by

$$\ddot q_c=\ddot q_c^d-T_{11}^{-1}T_{12}\dot{\tilde x}_c-T_{11}^{-1}M_{cc}^{-1}(q)\left(C_{cc}(q,\dot q)B^TT_0\tilde x_c-\overline u\right). \quad (31)$$

Equation (31) gives the necessary acceleration to make the controlled joints follow the desired reference trajectory. The torques in the actives joints can be calculated using this acceleration. For this, (2) is partitioned as [6]

$$\begin{bmatrix}\tau_a\\ 0\end{bmatrix}+\begin{bmatrix}\delta_a(q,\dot q,\ddot q)\\ \delta_u(q,\dot q,\ddot q)\end{bmatrix}=\begin{bmatrix}M_{ar}(q) & M_{ac}(q)\\ M_{ur}(q) & M_{uc}(q)\end{bmatrix}\begin{bmatrix}\ddot q_r\\ \ddot q_c\end{bmatrix}+\begin{bmatrix}b_a(q,\dot q)\\ b_u(q,\dot q)\end{bmatrix} \quad (32)$$

where $b(q,\dot q)=C(q,\dot q)+F(\dot q)+G(q)$. Factoring out the vector $\ddot q_r$ in the second line of (32) and substituting in the first one, we obtain

$$\tau_a=\left(M_{ac}(q)-M_{ar}(q)M_{ur}^{-1}(q)M_{uc}(q)\right)\ddot q_c+b_a(q,\dot q)$$
$$-\delta_a(q,\dot q,\ddot q)-M_{ar}(q)M_{ur}^{-1}(q)(b_u(q,\dot q)-\delta_u(q,\dot q,\ddot q)). \quad (33)$$

Fig. 1. Underactuated Arm II.

The optimal control law for the underactuated case is given as follows:

$$\overline u^*=-R^{-1}\bar B^TT_0\tilde x_c \quad (34)$$

where the matrix $\bar B$ is given in (7). The matrix $P_c(\tilde x_c,t)$ chosen as

$$P_c(\tilde x_c,t)=T_0^T\begin{bmatrix}M_{cc}(\tilde x_c,t) & 0\\ 0 & K_c\end{bmatrix}T_0$$

where K_c is a positive definite symmetric constant matrix which solves the analogous $\mathcal{H}_\infty$ control problem for the underactuated case. Note that the skew-symmetry of the matrix $C_{cc}(q,\dot q)-(1/2)\dot M_{cc}(q,\dot q)$ is still guaranteed.

VI. Experimental Results

To validate the proposed $\mathcal{H}_\infty$ control solutions, we apply the techniques presented in the previous sections to our experimental underactuated manipulator Underactuated Arm II (UArmII), designed and built by H. Ben Brown, Jr., Pittsburgh, PA (Fig. 1). This three-link manipulator has special-purpose joints containing each an actuator (a dc motor) and a pneumatic brake, so that they can act as active or passive joints. The manipulator configuration can be changed enabling or not the dc motor of each joint. All possible configurations, according to the active (A) and passive (P) joints location in the arm, are accepted: AAA, AAP, APA, PAA, APP, PAP, and PPA. For example, the configuration AAP means that joints 1 and 2 are active and joint 3 is passive.

The manipulator's kinematic and dynamic nominal parameters, which are used to calculate the nominal matrices $M_0(q)$, $C_0(q,\dot q)$ (see Appendix), and $F_0(\dot q)$ are shown in Table I, where F_i is the coefficient of a linear velocity-dependent frictional term $F_0(\dot q)$. The motor dynamic model is assumed to be linear and given by $V=\alpha\tau$, where the vector V contains the applied voltages and $\alpha=4.4\ \mathrm{V/N.m}$.

All the control strategies are developed and implemented in the MatLab programming environment. The interface between the computer and the motor amplifiers is made by an

TABLE I
ROBOT PARAMETERS

i	m_i	I_i	L_i	lc_i	F_i
	(kg)	(kg.m^2)	(m)	(m)	(kg.m^2/s)
1	0.850	0.0153	0.203	0.096	0.28
2	0.850	0.0100	0.203	0.096	0.18
3	0.625	0.0100	0.203	0.077	0.10

input–output Servo-To-Go board. This board contains analog outputs to set the amplifiers, digital inputs with quadrature to read the encoders and digital outputs to set the solenoid valves of the brakes. The board virtual device driver is accessed by dynamically linked libraries (dlls) created in the MatLab workspace from C++ files.

A. Fully Actuated Configuration

The experiment was performed with an initial position $q(0) = [0^\circ\ 0^\circ\ 0^\circ]^T$ and desired final position $q(T) = [-20^\circ\ 30^\circ\ -30^\circ]^T$, where the vector $T = [3.4\ 4.0\ 4.0]$ seconds contains the trajectory duration time for each joint. The reference trajectory, q^d, is a fifth degree polynomial. An external disturbance τ_d, starting at 2.0 s, is introduced in the active joints in the form

$$\tau_d = \begin{bmatrix} 3e^{-2t}\sin(4\pi t) \\ -2e^{-2t}\sin(4\pi t) \\ 2e^{-2t}\sin(4\pi t) \end{bmatrix}.$$

The maximum disturbance peak is approximately 40% of the torque value at $t = 2.0$ s.

1) Quasi-LPV Results: To apply the algorithm described in Section IV, the manipulator has to be represented as in (11). The parameters $\rho(\tilde{x})$ chosen are the state representing the position errors of joints 2 and 3, i.e., $\tilde{q}_2$ and $\tilde{q}_3$, respectively

$$\rho(\tilde{x}) = [\,\tilde{q}_2 \quad \tilde{q}_3\,].$$

This choice is based on the fact that the inertia matrix, $M(q)$, and the Coriolis matrix, $C(q,\dot{q})$, are functions of the positions of joints 2 and 3, and, as said before, are functions of the position errors. The system outputs, z_1 and z_2, are the position and velocity errors and the control variable, u, respectively. Hence, the system can be described by (11) with

$$\begin{aligned} A(\rho(x)) &= A(\rho(\tilde{x})) \\ B_1(\rho(x)) &= B \\ B_2(\rho(x)) &= B \\ C_1(\rho(x)) &= I_6 \\ C_2(\rho(x)) &= 0 \end{aligned}$$

where the matrices $A(\rho(\tilde{x}))$ and B are defined in (4).

The compact set P is defined as $\rho(\tilde{x}) \in [-30,30]^\circ \times [-30,30]^\circ$. The parameter variation rate is bounded by $|\dot{\rho}| \le 50^\circ/\text{s}$. The basis functions selected are

$$\begin{aligned} f_1(\rho(\tilde{x})) &= g_1(\rho(\tilde{x})) = 1 \\ f_2(\rho(\tilde{x})) &= g_2(\rho(\tilde{x})) = \cos(\tilde{q}_2) \\ f_3(\rho(\tilde{x})) &= g_3(\rho(\tilde{x})) = \cos(\tilde{q}_3). \end{aligned}$$

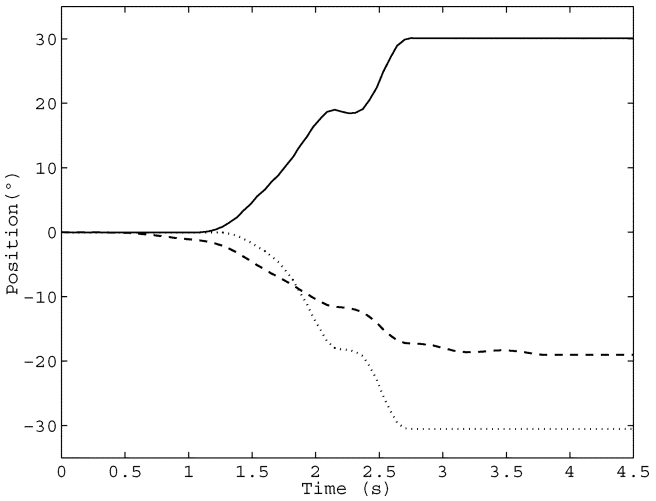

Fig. 2. Joint position, quasi-LPV control, and fully actuated configuration.

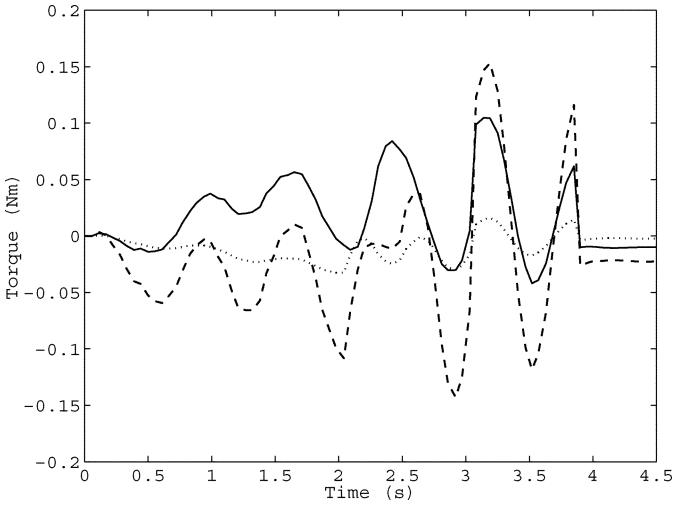

Fig. 3. Applied torque, quasi-LPV control, and fully actuated configuration.

The matrices $X(\rho(\tilde{x}))$ and $N(\rho(\tilde{x}))$, when represented in this basis are given by

$$X(\rho(\tilde{x})) = \sum_{i=1}^{3} f_i(\rho(\tilde{x}))X_i$$

and

$$N(\rho(\tilde{x})) = \sum_{i=1}^{3} g_i(\rho(\tilde{x}))N_i\,[\,-\tilde{q}^T \quad \dot{\tilde{q}}^T\,]$$

noting that the matrix $N(\rho(\tilde{x}))$ satisfies the condition $N(\rho(\tilde{x}))\tilde{x} = 0$. The parameter space was divided in $L = 5$ which means that 125 LMIs have to be solved for the X_i and N_i variables. The best attenuation level found, after two iterations $X - N$, was $\gamma = 1.2$. In the experimental results, joint positions and applied torques are shown in Figs. 2 and 3, respectively. In all graphics presented in this section and in the next sections, the dashed line represents joint 1, the solid line indicates joint 2, and the dot line represents joint 3.

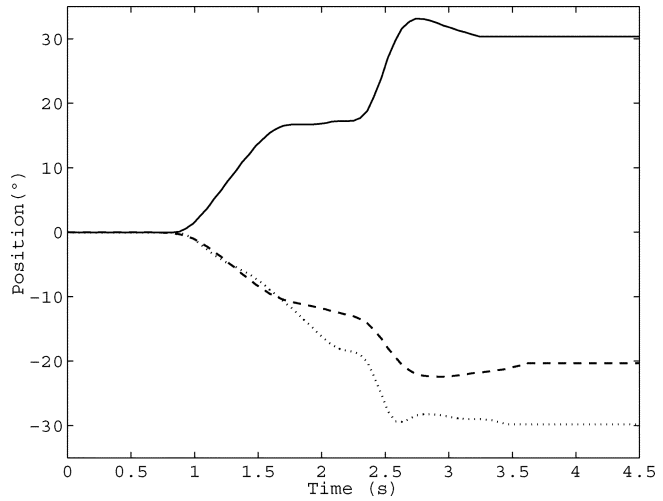

Fig. 4. Joint position, game theory control, and fully actuated configuration.

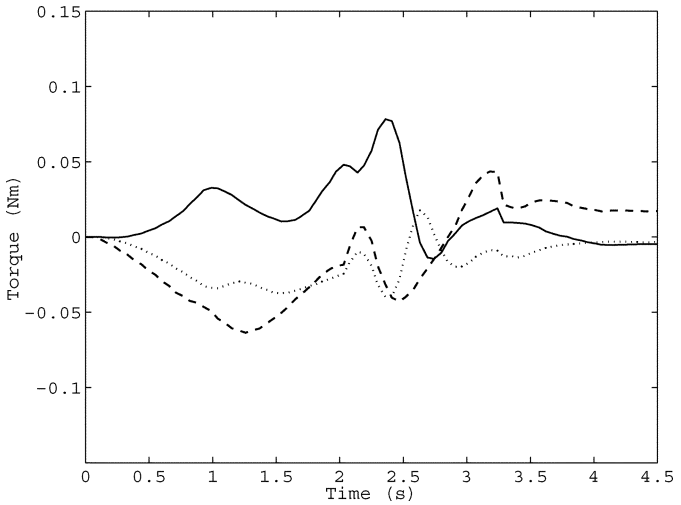

Fig. 5. Applied torque, game theory control, and fully actuated configuration.

2) Game Theory Results: For the controller designed via game theory, described in Section V, the attenuation level found was $\gamma = 1.9$. The weighting matrices used were

$$Q_1 = I_3,\ Q_2 = 2I_3,\ Q_{12} = 0,\ \text{and}\ R = 3.5I_3.$$

The experimental results are shown in Figs. 4 and 5. If we compare both methodologies (Figs. 2 and 4), we can observe that the second one takes more time to avoid the disturbances that occur at $t = 2.0$ s, before reaching the desired position. We can observe also that the torques of the quasi-LPV controller, Fig. 3, oscillate with more intensity to avoid disturbances than the torques of the controller based on game theory, Fig. 5.

B. Underactuated Configuration

The underactuated configuration used to validate the proposed methodology was the APA configuration. For this configuration, two control phases are necessary to control all joints to the set-point. Since the configuration APA has $n_a = 2$, two joints can be controlled in each phase. In the first phase, named APA$_u$, the vector of controlled joints, q_c, is selected as $q_c = [\ q_2\ q_3\ ]^T$, i.e., one passive joint (joint 2) and one active joint (joint 3) are selected (possibility 2 described in Section III). In the second phase, named APA$_l$, the active joints are selected to form the vector of controlled joints, $q_c = [\ q_1\ q_3\ ]^T$ (possibility 3). In this phase, joint 2 is kept locked, since it has already reached the set-point.

For the experiment, the initial position and the desired final position were, respectively, $q(0) = [0^\circ\ 0^\circ\ 0^\circ]^T$ and $q(T^1, T^2) = [20^\circ\ 20^\circ\ 20^\circ]^T$, where $T^1 = [1.0\ 1.0]$ seconds and $T^2 = [5.0\ 5.0]$ seconds are the trajectory duration time for the phases 1 and 2, respectively. An external disturbance τ_d, starting at 0.3 s, is introduced in the active joints 1 and 3 in the form

$$\tau_d = \begin{bmatrix} 0.5e^{-4t}\sin(4\pi t) \\ -0.05e^{-6t}\sin(4\pi t) \end{bmatrix}.$$

The maximum disturbance peak is approximately 30% of the torque value at $t = 0.3$ s.

1) Quasi-LPV Results: To apply the algorithm described in Section IV, the manipulator should be represented by (11). For the first control phase, the parameters $\rho(\tilde{x}_c)$ selected are the state representing the position errors of the joints 2 and 3, i.e., $\tilde{q}_2$ and $\tilde{q}_3$, respectively

$$\rho(\tilde{x}_c) = [\,\tilde{q}_2 \quad \tilde{q}_3\,]^T.$$

The system outputs, z_1 and z_2, are position and velocity errors of the controlled joints and the control variable, u, respectively. Hence, the system can be described by (11) with

$$\begin{aligned} A(\rho(x)) &= \overline{A}(\rho(\tilde{x}_c)) \\ B_1(\rho(x)) &= \overline{B} \\ B_2(\rho(x)) &= \overline{B} \\ C_1(\rho(x)) &= I_4 \\ C_2(\rho(x)) &= 0 \end{aligned}$$

where the matrices $\overline{A}(\rho(\tilde{x}_c))$ and $\overline{B}$ are defined in (7). The compact set P, parameter variation rates, and the basis functions are the same used in the AAA configuration. The parameter space was divided in $L = 5$. The best attenuation level found, after two iterations $X - N$ was $\gamma = 1.35$.

For the second control phase, the parameters selected that are part of the state vector were

$$\rho(\tilde{x}_c) = [\,\tilde{q}_3 \quad \dot{\tilde{q}}_3\,]^T$$

where $\tilde{q}_3$ and $\dot{\tilde{q}}_3$ are the position and velocity errors of joint 3. The position and velocity errors of the controlled joints and the control variable u are the outputs of the system.

The compact set P is defined as $\rho(\tilde{x}_c) \in [-30,\ 30]^\circ \times [-50,\ 50]^\circ/\text{s}$. The parameter variation rate is bounded by $|\dot{\rho}| \leq [50^\circ/\text{s}\ 30^\circ/\text{s}^2]$. The basis functions selected are

$$\begin{aligned} f_1(\rho(\tilde{x_c})) &= g_1(\rho(\tilde{x}_c)) = 1 \\ f_2(\rho(\tilde{x_c})) &= g_2(\rho(\tilde{x}_c)) = \cos(\tilde{q}_3) \\ f_3(\rho(\tilde{x_c})) &= g_3(\rho(\tilde{x}_c)) = \cos(\dot{\tilde{q}}_3). \end{aligned}$$

The parameter space was divided in $L = 5$. The best level of attenuation found after 2 iterations $X - N$ was $\gamma = 1.80$.

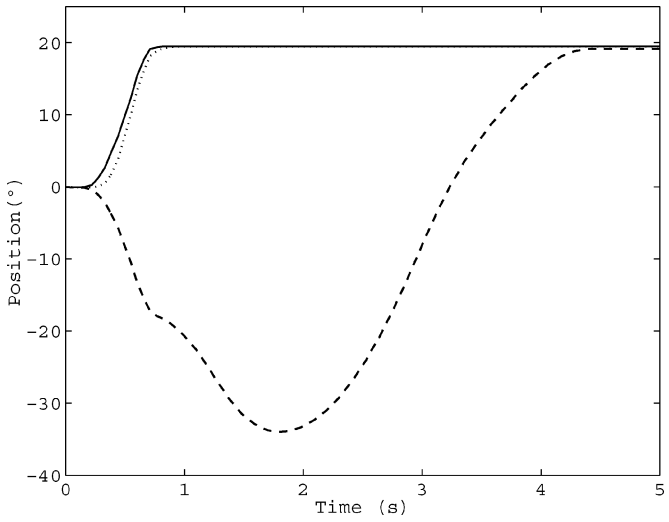

Fig. 6. Joint position, quasi-LPV control, and underactuated configuration.

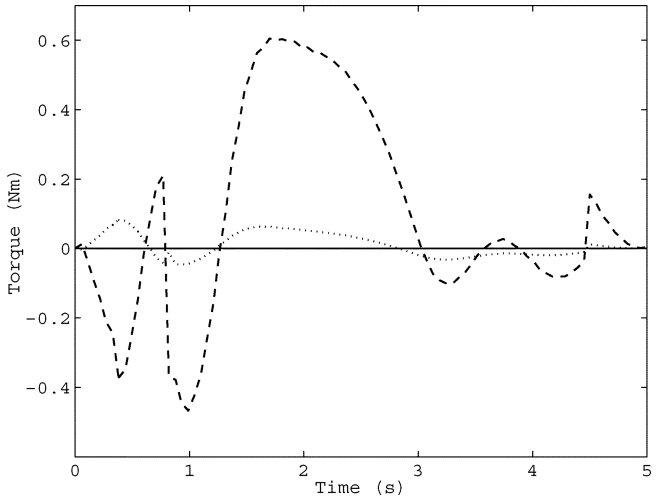

Fig. 7. Applied torque, quasi-LPV control, and underactuated configuration.

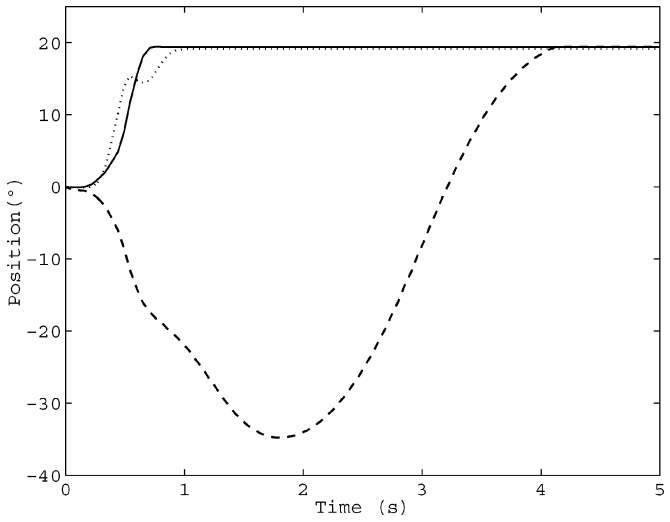

Fig. 8. Joint position, game theory control, and underactuated configuration.

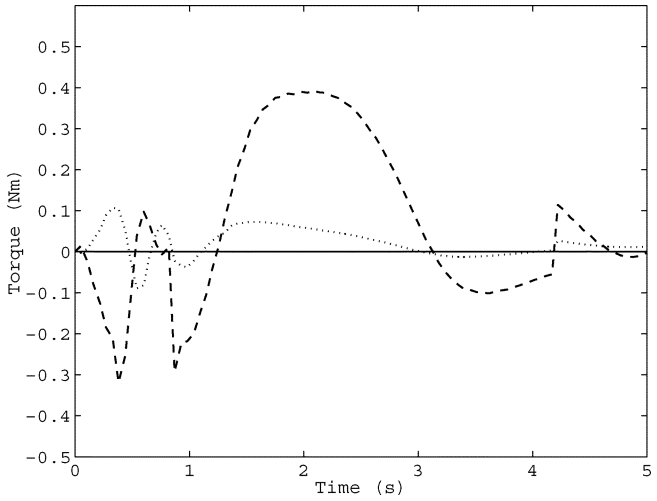

Fig. 9. Applied torque, game theory control, and underactuated configuration.

Experimental results of joint positions and applied torques are shown in Figs. 6 and 7, respectively.

2) Game Theory Results: For the first control phase, the best attenuation level found was $\gamma = 1.9$. The weighting matrices used were $Q_1 = I_2$, $Q_2 = 4I_2$, $Q_{12} = 0$, and $R = 3.5I_2$. For the second control phase, the best attenuation level found was $\gamma = 1.9$. The weighting matrices used in this phase are the same used in the first one. The experimental results are shown in Figs. 8 and 9, respectively.

Note that matrix $M_{cc}(q)$ in the first control phase, i.e., $M_{cc}(q) = [M_{22}(q) \;\; M_{23}(q); \; M_{32}(q) \;\; M_{33}(q)]$, is a function of joint positions 2 and 3 based on (30), where $M_{cc}(q)$ is considered in terms of controlled positions. In the second control phase joint 2 is locked and $M_{cc}(q) = [M_{11}(q) \;\; M_{13}(q); \; M_{31}(q) \;\; M_{33}(q)]$ is a function only of q_3 which is a controlled joint in this phase. Hence, matrix $M_{cc}(q)$ is a function only of the controlled joints during both control phases of this configuration (APA).

Comparing Figs. 6–7 and Figs. 8–9, we can verify that the quasi-LPV technique has presented better robustness compared to the second technique, based on game theory. This comparative result first can be verified looking for the γ of the both control strategies in Table II. In addition, the positions of joints 2 and 3 for the period of time between 0 and 1.5 s show that the technique based on game theory presented an oscillation in the third joint in the presence of the disturbances. This control law is similar in nature to the control laws based on feedback linearization, because of its static gain; certainly this is a limitation for effective disturbance rejection. It is important to note that a disadvantage of the LPV controller is the time spent with the selection of the basis functions to yield robust performance, which can be long.

TABLE II
γ FOR BOTH METHODOLOGIES

Controller	First phase	Second phase
Quasi-LPV	1.35	1.8
Game theory	1.9	1.9

C. Fault Occurrence

Consider now that the underactuated configuration APA is the consequence of a free torque fault in the second joint. With

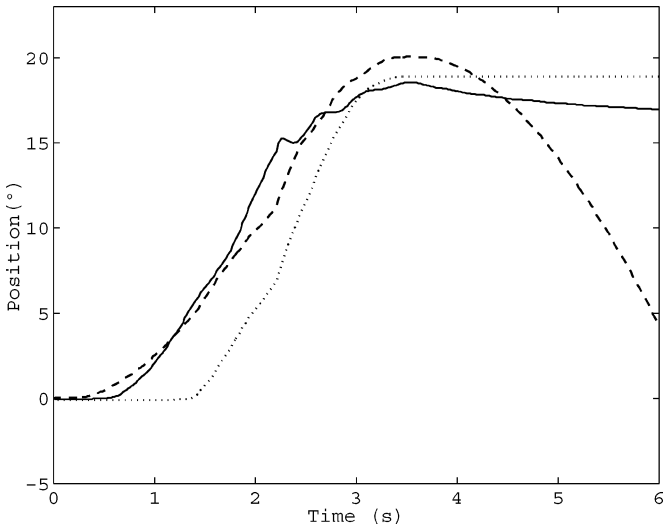

Fig. 10. Joint position, quasi-LPV control, and fault occurrence.

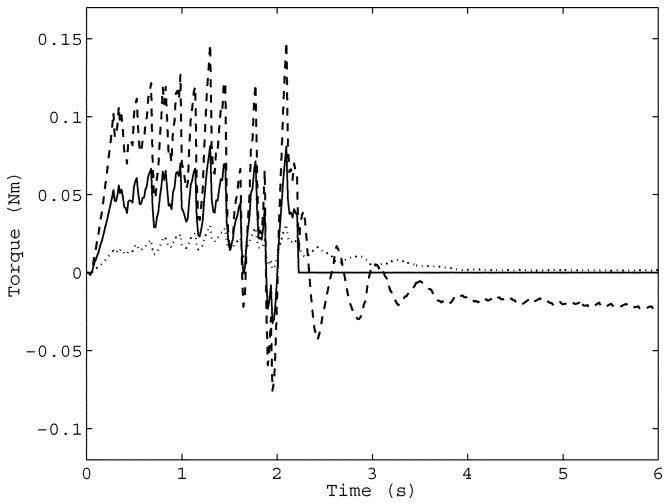

Fig. 11. Applied torque, quasi-LPV control, and fault occurrence.

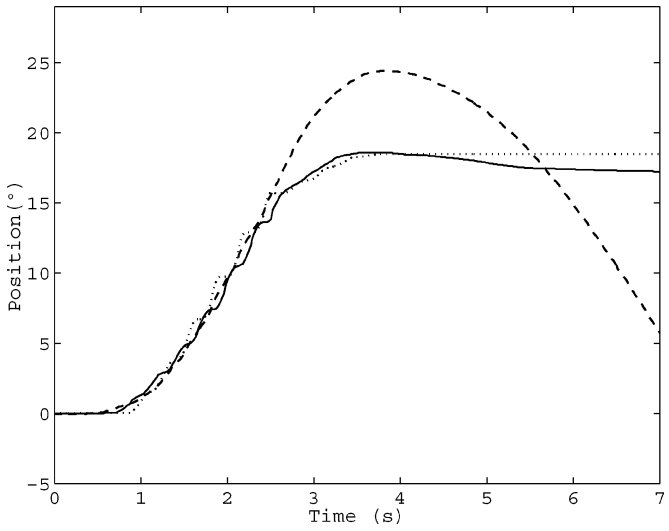

Fig. 12. Joint position, game theory control, and fault occurrence.

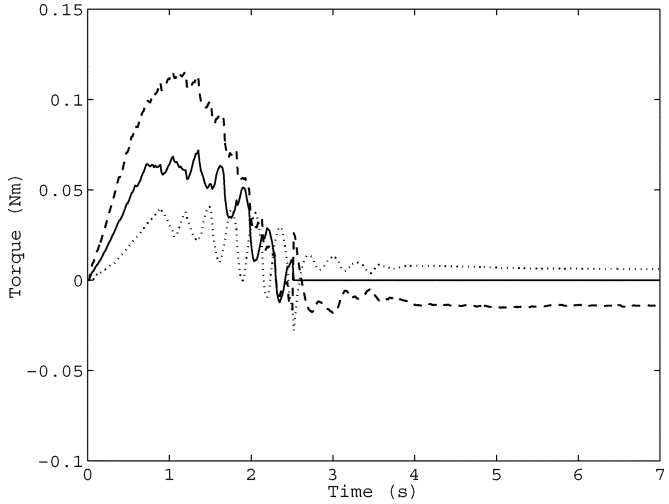

Fig. 13. Applied torque, game theory control, and fault occurrence.

the controllers designed in the previous sections, there is no guarantee that the joints will reach the set-point taking into account the sudden change in the configuration from AAA to APA. To verify this behavior, an experiment was implemented considering initially the manipulator in the fully-actuated configuration AAA. The initial position was $q(0) = [0^\circ\ 0^\circ\ 0^\circ]^T$ and the desired final position $q(T) = [20^\circ\ 20^\circ\ 20^\circ]^T$, where $T = [4.0\ 4.0\ 4.0]$ seconds. When the joint positions reached approximately 15° for all joints, at $t = 2.5$ s, an artificial free torque fault was introduced in the second joint. Here, we assume that a fault detection system such as those presented in [21] indicate the fault as soon as it occurs. Hence, the controller is changed from the fully actuated configuration to the underactuated one maintaining the manipulator's movement (without using the brakes during the control reconfiguration). As can be seen in Figs. 10 and 11, the quasi-LPV controllers were not able to react immediately to the fault occurrence. The same behavior is observed when the controllers based on game theory are used, Figs. 12 and 13.

An alternative procedure in the presence of a fault is to use brakes during the reconfiguration phase. In this case, all joints are locked for t_f seconds between the fault detection and the beginning of the APA configuration control phase. The time t_f is chosen according to the manipulator's dynamics and the time response of the brakes. One disadvantage of this procedure is the necessity of turning on the brakes even though the joints may be moving with high velocities, which can damage some components. This procedure, using brakes, was implemented in the UArm II. For the UArm II, when the brakes are turned on, there are some oscillations in the joint positions which take at least 1 s to subside. Hence, we choose $t_f = 1.0$ s. These oscillations can be seen in the results of this experiment using the quasi-LPV controllers (Figs. 14 and 15) and the game theory ones (Figs. 16 and 17). Note another disadvantage of this procedure: it is necessary a high torque to restart the motion after the reconfiguration phase. Hence, our next objective is to design a controller that eliminates the necessity of locking the joints between the fully actuated and underactuated phases.

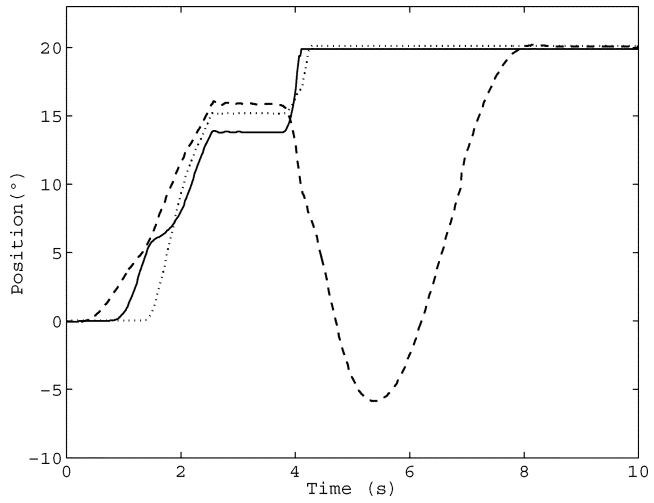

Fig. 14. Joint position, quasi-LPV control, and reconfiguration with brakes.

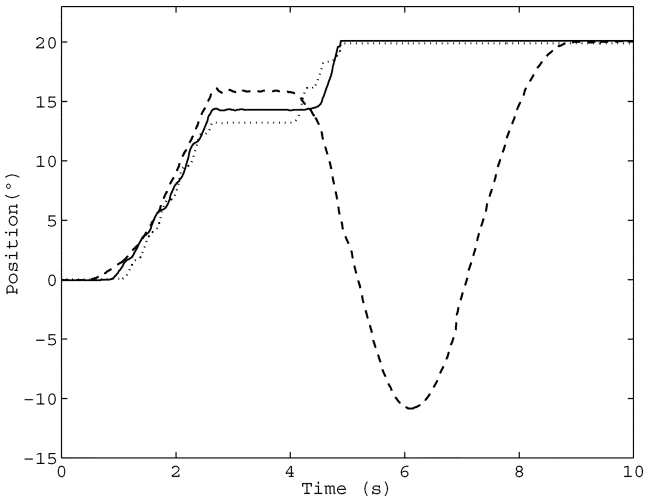

Fig. 16. Joint position, game theory control, and reconfiguration with brakes.

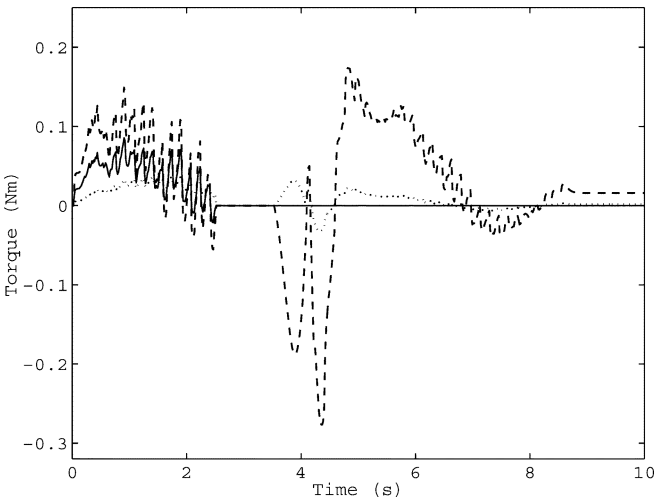

Fig. 15. Applied torque, quasi-LPV control, and reconfiguration with brakes.

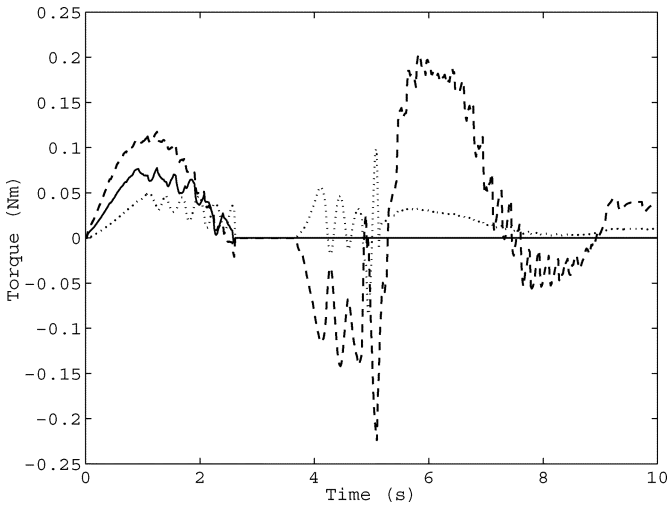

Fig. 17. Applied torque, game theory control, reconfiguration with brakes.

VII. Markovian Control

A. Markovian Model

In this section, Markov theory is used to solve the problem of free torque fault occurrence described above. The dynamic model of an underactuated manipulator, (33), can be represented as

$$\tau_a = \overline{M}(q)\ddot{q}_c + \overline{b}(q,\dot{q}) + \overline{\delta}(q,\dot{q},\ddot{q}) \tag{35}$$

with

$$\begin{aligned}\overline{M}(q) &= M_{ac}(q) - M_{ar}(q)M_{ur}^{-1}(q)M_{uc}(q)\\ \overline{b}(q,\dot{q}) &= b_a(q,\dot{q}) - M_{ar}(q)M_{ur}^{-1}(q)b_u(q,\dot{q})\\ \overline{\delta}(q,\dot{q},\ddot{q}) &= -\delta_a(q,\dot{q},\ddot{q}) + M_{ar}(q)M_{ur}^{-1}(q)\delta_u(q,\dot{q},\ddot{q}).\end{aligned}$$

The fully actuated manipulator (2) can be represented by (35), with $\overline{M}(q) = M(q)$, $\overline{b}(q,\dot{q}) = b(q,\dot{q})$, and $\overline{\delta}(q,\dot{q},\ddot{q}) = -\delta(q,\dot{q},\ddot{q})$. The linearization of the fully actuated or underactuated manipulator, represented by (35), around an operation point with position q_0 and velocity $\dot{q}_0$, gives a system in the form

$$\begin{cases}\dot{x} &= \bar{A}(q,\dot{q})x + \bar{B}(q)\tau + \bar{B}(q)\bar{\delta}(q,\dot{q},\ddot{q})\\ z &= \bar{C}x + \bar{D}\tau,\end{cases}$$

where

$$\bar{A}(q,\dot{q}) = \left[\begin{matrix}0 & I\\ -\frac{\partial}{\partial q}(\overline{M}^{-1}(q)\overline{b}(q,\dot{q})) & -\overline{M}^{-1}(q)\frac{\partial}{\partial \dot{q}}(\overline{b}(q,\dot{q}))\end{matrix}\right]\Bigg|_{(q_0,\dot{q}_0)}$$

$$\bar{B}(q) = \left[\begin{matrix}0\\ \overline{M}^{-1}(q)\end{matrix}\right]\Bigg|_{(q_0)} \quad \bar{C} = \begin{bmatrix}I & 0\\ 0 & 0\end{bmatrix}$$

$$\bar{D} = \begin{bmatrix}0\\ I\end{bmatrix} \quad x = \begin{bmatrix}q^d - q\\ \dot{q}^d - \dot{q}\end{bmatrix}.$$

A preliminary PD controller with gains K_P and K_D can be introduced in the applied torques, τ, and the system can be rewritten as follows:

$$\begin{cases}\dot{x} &= \tilde{A}(q,\dot{q})x + \tilde{B}(q)u + \tilde{B}(q)\bar{\delta}(q,\dot{q},\ddot{q})\\ z &= \tilde{C}x + \tilde{D}u\end{cases}$$

TABLE III
LINEARIZATION

Operation Points			Linearization Points					
AAA	APA_u	APA_l	q_1	q_2	q_3	$\dot{q}_1$	$\dot{q}_2$	$\dot{q}_3$
1	9	17	5	5	5	0	0	0
2	10	18	15	5	5	0	0	0
3	11	19	5	15	5	0	0	0
4	12	20	15	15	5	0	0	0
5	13	21	5	5	15	0	0	0
6	14	22	15	5	15	0	0	0
7	15	23	5	15	15	0	0	0
8	16	24	15	15	15	0	0	0

where (see equation at the bottom of the page).

The Markov theory considered here is based on discrete time systems. Hence, the system in continuous time has to be defined in discrete form

$$\begin{cases} x(k+1) &= A(q,\dot{q})x(k) + B(q)u(k) + B(q)w(k) \\ z(k) &= Cx(k) + Du(k). \end{cases} \tag{36}$$

where $w(k)$ is the discrete disturbance $\bar{\delta}(q,\dot{q},\ddot{q})$. The Markovian model developed in this section describes both the changes at the operation points of the plant (36), and the probability of a fault occurrence. For simplicity and without loss of generality we will consider that faults occur only in the second joint. Consequently, the configuration after the fault is APA. Remember that, for this configuration, two phases of control are necessary to control all joints to the set-point, APA_u with $q_c = [\, q_2 \; q_3 \,]^T$ and APA_l with $q_c = [\, q_1 \; q_3 \,]^T$.

The workspace of each joint was divided in two areas, delimited in steps of 10°. For each area a point was defined, 5° for the first area and 15° for the second one. All the possible combinations of positioning of the three joints, q_1, q_2, q_3, in these two points were used to map the workspace of the manipulator. Then eight linearization points, with the velocities set to zero, were found. These points are shown in the right side of Table III. The number of linearization points depends on how high is the variation rate of the value of $A(q,\dot{q})$ and $B(q)$ with respect to q. For our manipulator UArm II, this rate is low. Hence, the number of points chosen here is actually very conservative. However, this choice does not add any difficulty to design the controllers.

For each linearization point we have three set of matrices $A(q,\dot{q})$, $B(q)$, C, and D. The first set defines normal operation AAA, where all joints are active; the second set, APA_u, defines a fault with the second joint unlocked; and the third set, APA_l, defines a fault with the second joint locked. The dimensions of the matrices $A(q,\dot{q})$, $B(q)$, C, and D for the configurations APA_u and APA_l are smaller than for the configuration AAA. However, in Markovian theory, all the systems of the model must have the same dimension. To solve this problem, lines and columns of zeros are introduced in the matrices.

According to our Markovian model we have 24 operation points, and according to Markovian theory we must group them in a transition probability matrix $\mathcal{P}$ of dimension 24×24. The matrix $\mathcal{P}$ was partitioned in nine submatrices of dimension 8×8, shown in (37). The operation points and the matrix $\mathcal{P}$ were selected empirically. We can independently model the normal operation, the faulty operation, and the probability of a fault occurrence adjusting the submatrices of $\mathcal{P}$.

We now explain how the matrix $\mathcal{P}$ works. The element p_{ij} of the matrix $\mathcal{P}$ represents the probability of the system, being at the operation point i, to go to the point j at the next discrete time k. This implies in $\sum_j p_{ij} = 1$ for any line i of $\mathcal{P}$.

We will refer to $\mathcal{P}$ in the partitioned form (37). The submatrix $\mathcal{P}_{\mathrm{AAA}}$ groups the relationship between operation points of normal operation (three active joints) in the set AAA, and the diagonal submatrix $\mathcal{P}_f$ groups the probabilities of a fault occurrence when the system is in normal operation AAA. We choose the same probabilities for all operation points. The null submatrix $\mathcal{P}_0$ in the first line of $\mathcal{P}$ represents that, when a fault occurs, the system goes from the set AAA to the set APA_u but not to the set APA_l.

After the occurrence of the fault the system is in the set APA_u, and we consider the second line of $\mathcal{P}$, where $\mathcal{P}_{\mathrm{APA_u}}$ groups the relationships between the operation points in the set APA_u, $\mathcal{P}_0$ represents that the defective joint cannot be repaired, and the matrix $\mathcal{P}_s$ represents the probability of the system to go to the set APA_l. After that, the system can be in the set APA_u or APA_l, according to $\mathcal{P}_s$. In the third line of $\mathcal{P}$, $\mathcal{P}_{\mathrm{APA_l}}$ groups the relationships between the operation points in the set APA_l, $\mathcal{P}_s$ represents the probability for the system to return to the set APA_u, and $\mathcal{P}_0$ represents that the defective joint cannot be repaired.

$$\mathcal{P} = \left[\begin{array}{c|c|c} \mathcal{P}_{\mathrm{AAA}} & \mathcal{P}_f & \mathcal{P}_0 \\ \hline \mathcal{P}_0 & \mathcal{P}_{\mathrm{APA_u}} & \mathcal{P}_s \\ \hline \mathcal{P}_0 & \mathcal{P}_s & \mathcal{P}_{\mathrm{APA_l}} \end{array}\right] \tag{37}$$

$$\tilde{A}(q,\dot{q}) = \left[\begin{array}{cc} 0 & I \\ -\frac{\partial}{\partial q}(\overline{M}^{-1}(q)\bar{b}(q,\dot{q})) & +\overline{M}^{-1}(q)K_P - \overline{M}^{-1}(q)[\frac{\partial}{\partial \dot{q}}(\bar{b}(q,\dot{q})) - K_D] \end{array}\right]\Bigg|_{(q_0,\dot{q}_0)}$$

$$\tilde{B}(q) = \left[\begin{array}{c} 0 \\ \overline{M}^{-1}(q) \end{array}\right]\Bigg|_{(q_0)} \qquad \tilde{C} = \begin{bmatrix} I & 0 \\ K_P & K_D \end{bmatrix}$$

$$\tilde{D} = \begin{bmatrix} 0 \\ I \end{bmatrix} \qquad x = \begin{bmatrix} q^d - q \\ \dot{q}^d - \dot{q} \end{bmatrix}$$

$$\tau = u - [\, K_P \quad K_D \,]\, x.$$

$$\mathcal{P}_{\mathrm{AAA}} = \begin{bmatrix} 0.89 & 0.10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.10 & 0.79 & 0.10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.10 & 0.79 & 0.10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.10 & 0.79 & 0.10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.10 & 0.79 & 0.10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.10 & 0.79 & 0.10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.10 & 0.79 & 0.10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.10 & 0.89 \end{bmatrix}$$

$$\mathcal{P}_f = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.01 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix}$$

$$\mathcal{P}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathcal{P}_{\mathrm{APA_u}} = \begin{bmatrix} 0.78 & 0.20 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.10 & 0.78 & 0.10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.10 & 0.78 & 0.10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.10 & 0.78 & 0.10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.10 & 0.78 & 0.10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.10 & 0.78 & 0.10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.10 & 0.78 & 0.10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.20 & 0.78 \end{bmatrix}$$

$$\mathcal{P}_{\mathrm{APA_l}} = \mathcal{P}_{\mathrm{APA_u}} \quad \mathcal{P}_s = 2\mathcal{P}_f.$$

The system starts at an operation point in the set AAA, and it operates according to $\mathcal{P}_{\mathrm{AAA}}$ and has a probability given by $\mathcal{P}_f$ of a fault occur. If a fault occurs, the system keeps the linearization point and goes to the corresponding operation point in the set APA$_u$. Later the system switches between the sets APA$_u$ and APA$_l$ according to $\mathcal{P}_s$ until all joints reach the set point. The Markovian model is shown in Fig. 18.

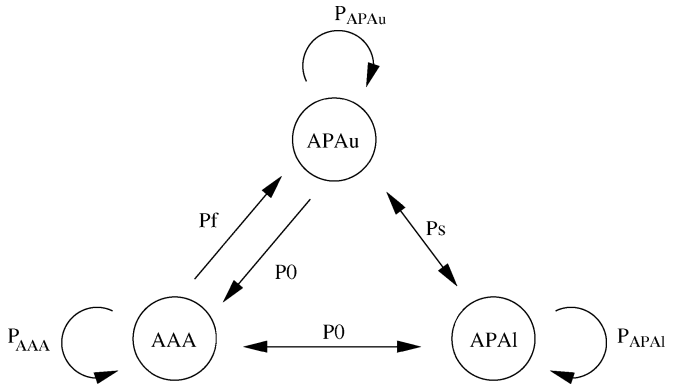

Fig. 18. Markovian model of a manipulator with free joint fault.

B. $\mathcal{H}_\infty$ Markovian Control

In this section we consider the $\mathcal{H}_\infty$ Markovian control in discrete time form given in [11], for the general system given by the linear system

$$\begin{cases} x(k+1) = A_{\Theta(k)}(q,\dot{q})x(k) + B_{\Theta(k)}(q)u(k) + W_{\Theta(k)}w(k) \\ z(k) = C_{\Theta(k)}x(k) + D_{\Theta(k)}u(k) \\ x(0) = x_0 \\ \Theta(0) = \Theta_0 \end{cases} \tag{38}$$

subjected to Markovian jumps where $A_{\Theta(k)} = (A_1, \ldots, A_N) \in \mathbb{H}^n$, $B_{\Theta(k)} = (B_1, \ldots, B_N) \in \mathbb{H}^{m,n}$, $C_{\Theta(k)} = (C_1, \ldots, C_N) \in \mathbb{H}^{n,s}$, $D_{\Theta(k)} = (D_1, \ldots, D_N) \in \mathbb{H}^{m,s}$ with $D_i^* D_i > 0$ for all i, $W_{\Theta(k)} = (W_1, \ldots, W_N) \in \mathbb{H}^{r,n}$ [these matrices are given by the system (36)], and $w = (w(0), w(1), \ldots) \in l_2^r$. $\Theta(k)$ is a Markov chain with values in $\{1, \ldots, N\}$, and Θ_0 is used to specify the initial condition of the Markov chain. The control input $u(k)$ is given by

$$u(k) = F_{\Theta(k)}x(k) \tag{39}$$

where $F_{\Theta(k)}$ will be defined in the following. Throughout this section the following notation is used: $\mathbb{R}^n$ and $\mathbb{C}^n$ stand for the n-dimensional real and complex Euclidean spaces, respectively; $\mathbb{B}(\mathbb{C}^m, \mathbb{C}^n)$ denotes the normed linear space of all $n \times m$ complex matrices and $\mathbb{B}(\mathbb{R}^m, \mathbb{R}^n)$ denotes the normed linear space of all $n \times m$ real matrices; for simplicity, written as $\mathbb{B}(\mathbb{C}^n)$ and $\mathbb{B}(\mathbb{R}^n)$ whenever $n = m$; we set * for conjugate transpose, and $\mathbb{H}^{m,n}$ denotes the linear space made up of all sequences of complex matrices $V = (V_1, \ldots, V_N)$, with $V_i \in \mathbb{B}(\mathbb{C}^m, \mathbb{C}^n)$ for $i = 1, \ldots, N$; for simplicity $\mathbb{H}^{n,n} = \mathbb{H}^n$; l_2^r is the Hilbert set of random variables of second-order $w = \big(w(0), w(1), \ldots\big)$ with $w(k) \in \mathbb{R}^r$ where

$$\|w\|_2^2 = \sum_{k=0}^{\infty} \|w(k)\|_2^2 < \infty$$

and $\|w(k)\|_2^2 = E(\|w(k)\|^2)$. The operator $\mathcal{E}$ is defined as $\mathcal{E}(\cdot) = \big(\mathcal{E}_1(\cdot), \ldots, \mathcal{E}_N(\cdot)\big) \in \mathbb{B}(\mathbb{R}^n)$, and for $V = (V_1, \ldots, V_N) \in \mathbb{H}^n$

$$\mathcal{E}_i(V) = \sum_{j=1}^{N} p_{ij} V_j$$

where $\mathcal{P} = [p_{ij}]$. For simplicity and without loss of generality [11], we set $D_i^* D_i = I$ and $C_i^* D_i = 0$.

Taking $\mathcal{Z}(\Theta_0, w) = (z(0), z(1), \ldots)$ as the set of all outputs of the system (38), the following norm is defined:

$$\|\mathcal{Z}(\Theta_0, \cdot)\| = \sup_{w \in l_2^r} \frac{\|\mathcal{Z}(\Theta_0, w)\|_2}{\|w\|_2}. \tag{40}$$

The $\mathcal{H}_\infty$ control problem consists in obtaining a controller that stabilizes the linear system (38) and ensures that the norm from the additive input disturbance to the output (40) is less than a specified attenuation value γ, with $Q_i = C_i^* C_i$, (C, A) detectable in the quadratic mean, and $\gamma > 0$ fixed. There exists $F = (F_1, \ldots, F_N) \in \mathbb{H}^{n,m}$ that stabilizes (A, B) in the

quadratic mean, and $||\mathcal{Z}(\Theta_0,\cdot)|| < \gamma$ for all Θ_0, if there exist $X = (X_1,\ldots,X_N) \in \mathbb{H}^{n+}$ satisfying the conditions

i) $I - (1/\gamma^2)W_i^*\mathcal{E}_i(X)W_i > 0$

ii)

$$\begin{aligned} X_i =& Q_i + A_i^*\mathcal{E}_i(X)A_i - A_i^*\mathcal{E}_i(X)\begin{bmatrix}B_i & \frac{1}{\gamma}W_i\end{bmatrix}\left(\begin{bmatrix}I & 0\\ 0 & -I\end{bmatrix}\right. \\ &\left. + \begin{bmatrix}B_i^*\\ \frac{1}{\gamma}W_i^*\end{bmatrix}\mathcal{E}_i(X)\begin{bmatrix}B_i & \frac{1}{\gamma}W_i\end{bmatrix}\right)^{-1} \\ &\begin{bmatrix}B_i^*\\ \frac{1}{\gamma}W_i^*\end{bmatrix}\mathcal{E}_i(X)A_i \\ =& Q_i + F_i^*F_i - G_i^*G_i + \left(A_i + B_iF_i + \frac{1}{\gamma}W_iG_i\right)^*\mathcal{E}_i(X) \\ &\times\left(A_i + B_iF_i + \frac{1}{\gamma}W_iG_i\right) \end{aligned}$$

where

$$\begin{aligned} F_i =& -\left(I + B_i^*\mathcal{E}_i(X)B_i + \frac{1}{\gamma^2}B_i^*\mathcal{E}_i(X)W_i\left(I\right.\right. \\ &\left.\left. - \frac{1}{\gamma^2}W_i^*\mathcal{E}_i(X)W_i\right)^{-1}W_i^*\mathcal{E}_i(X)B_i\right)^{-1}B_i^*\left(I\right. \\ &\left. + \frac{1}{\gamma^2}\mathcal{E}_i(X)W_i\left(I - \frac{1}{\gamma^2}W_i^*\mathcal{E}_i(X)W_i\right)^{-1}W_i^*\right)\mathcal{E}_i(X)A_i = \\ &-(I + B_i^*\mathcal{E}_i(X)B_i)^{-1}B_i^*\mathcal{E}_i(X)\left(A_i + \frac{1}{\gamma}W_iG_i\right) \\ G_i =& \left(I - \frac{1}{\gamma^2}W_i^*\mathcal{E}_i(X)W_i + \frac{1}{\gamma^2}W_i^*\mathcal{E}_i(X)B_i(I\right. \\ &\left. + B_i^*\mathcal{E}_i(X)B_i)^{-1}B_i^*\mathcal{E}_i(X)W_i\right)^{-1} \\ &\times\left(\frac{1}{\gamma^2}W_i^*(I - \mathcal{E}_i(X)B_i(I + B_i^*\mathcal{E}_i(X)B_i)^{-1}B_i^*)\mathcal{E}_i(X)A_i\right) \\ =& \left(I - \frac{1}{\gamma^2}W_i^*\mathcal{E}_i(X)W_i\right)^{-1}\frac{1}{\gamma}W_i^*\mathcal{E}_i(X)(A_i + B_iF_i). \end{aligned}$$

iii) $r_\sigma(\mathcal{L}) < 1$, where $\mathcal{L}(\cdot) = (\mathcal{L}_1(\cdot),\ldots,\mathcal{L}_N(\cdot))$ is defined by

$$\begin{aligned} \mathcal{L}_i(\cdot) = \left(A_i + B_iF_i + \frac{1}{\gamma}W_iG_i\right)^*\mathcal{E}_i(\cdot) \\ \times\left(A_i + B_iF_i + \frac{1}{\gamma}W_iG_i\right) \end{aligned}$$

for $i = 1,\ldots,N$.

According to [11], *i*) to *iii*) are necessary and sufficient conditions for the existence of a solution to the $\mathcal{H}_\infty$ control problem.

C. Markovian Results

In this section, we present experimental results obtained from implementation of the $\mathcal{H}_\infty$ Markovian controller, calculated according to the previous section and using the Markovian model described in Section VII-A, in the underactuated manipulator UArm II. The controller was designed using the discrete time Markovian jump linear systems (DTMJLS) toolbox, developed by Ricardo P. Marques, see [18] for details.

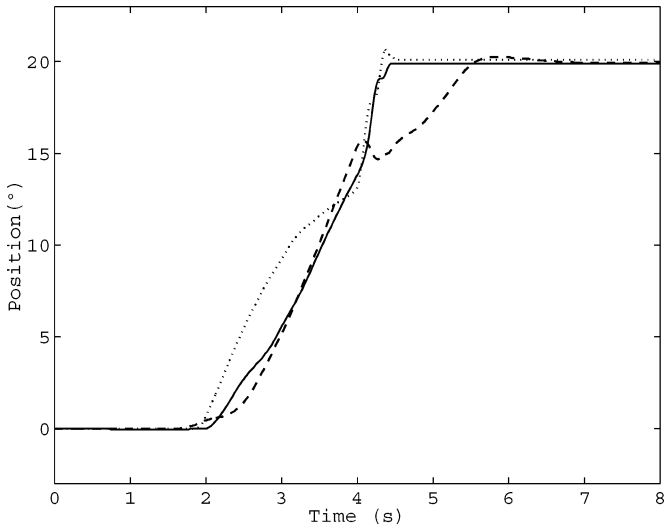

Fig. 19. Joint position and $\mathcal{H}_\infty$ Markovian control.

The experiment was realized with initial position $q(0) = [0^\circ\ 0^\circ\ 0^\circ]^T$ and desired final position $q(T) = [20^\circ\ 20^\circ\ 20^\circ]^T$, where the vector $T = [4.0\ 4.0\ 4.0]$ s is the trajectory duration for each joint. The minimal γ found was $\gamma = 10$. The reference trajectory, q^d, is a fifth degree polynomial. The initial configuration is AAA, with the operation point starting at 1. The changes of operation points are defined according to the actual position of the manipulator joints. To validate the proposed fault-tolerant controller, an artificial fault was introduced at $t = 4.0$ s, changing the Markovian chain from the first configuration, AAA, to the second one, APA_u, maintaining the operation point. At this time, the joint positions are approximately 15° for all joints, similar to the experiment described in Section VI-C. A preliminary PD controller utilized was defined as:

$$K_{P_{\text{AAA}}} = \begin{bmatrix}0.2 & 0 & 0\\ 0 & 0.15 & 0\\ 0 & 0 & 0.12\end{bmatrix}$$

$$K_{D_{\text{AAA}}} = \begin{bmatrix}0.02 & 0 & 0\\ 0 & 0.02 & 0\\ 0 & 0 & 0.02\end{bmatrix}$$

$$K_{P_{\text{APA}_\text{u}}} = \begin{bmatrix}-0.66 & -0.03\\ -0.04 & 0.33\end{bmatrix}$$

$$K_{D_{\text{APA}_\text{u}}} = \begin{bmatrix}-0.07 & -0.01\\ -0.04 & 0.06\end{bmatrix}$$

$$K_{P_{\text{APA}_\text{l}}} = \begin{bmatrix}1.25 & 0.19\\ 0.06 & 0.29\end{bmatrix}$$

and

$$K_{D_{\text{APA}_\text{l}}} = \begin{bmatrix}0.27 & 0.02\\ 0.01 & 0.01\end{bmatrix}.$$

The experimental results: joint position, applied torque, and Markov chain for the $\mathcal{H}_\infty$ Markovian controller are shown in Figs. 19–21, respectively. Even after the artificial fault introduction, the system kept the stability and reached the desired setpoint. Again, we consider that an actual fault detection system is in use.

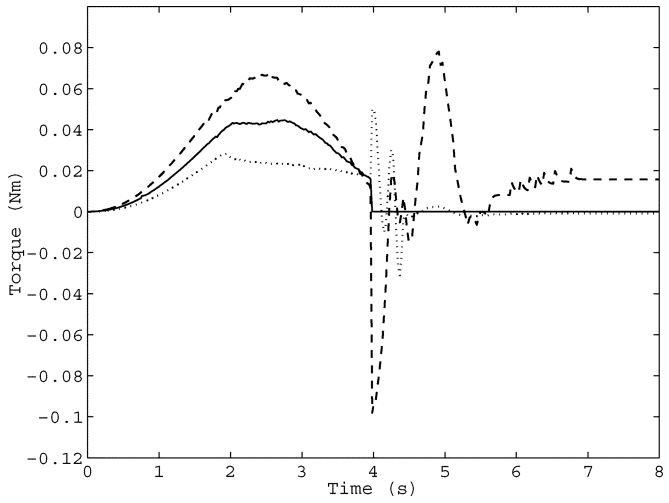

Fig. 20. Applied torque and $\mathcal{H}_\infty$ Markovian control.

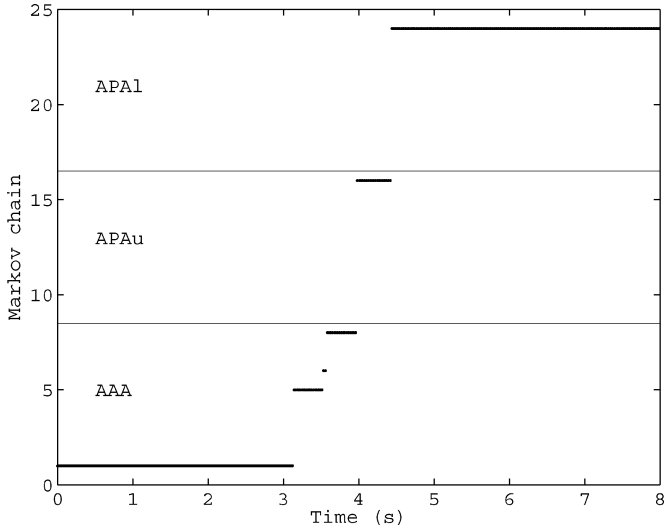

Fig. 21. Markov chain and $\mathcal{H}_\infty$ Markovian control.

Note the fast response of the controller after the fault occurrence with a high-torque variation rate. This behavior is not observed in the same experiment with the quasi-LPV and game theory controllers, Figs. 11 and 13. Despite this fast response, the maximum torque with the Markovian controller is lower than the ones obtained with the other controllers when the brakes were used, Figs. 15 and 17. We can also compare these controllers observing the total energy spent in the experiments (Figs. 15, 17, and 20), computed by the sum of the absolute value of the torque of all joints, Table IV. The lower consumption of energy of the Markovian controller can be considered as another advantage of this methodology, for it eliminates the use of brakes in the reconfiguration phase, which was the objective proposed in Section VI-C. Finally, it was investigated the possibility of defining a limit to the maximum torque of the Markovian controller. Two Markovian controllers with control constraint were found in the literature, [7] and [12]. Both controllers were applied in our control system and, unfortunately, do not work for this system.

TABLE IV
ENERGY SPENT DURING THE MANIPULATOR MOTION

Controller	Energy (N.m.s)
$GameTheory$	0.94
$Quasi-LPV$	0.78
$Markovian$	0.39

VIII. CONCLUSION

Two nonlinear $\mathcal{H}_\infty$ control methodologies to control fully actuated and underactuated manipulators were compared in this article. Experimental results have shown that the quasi-LPV technique has presented better robustness in comparison with the game theory technique. When we apply a fault in both controllers, we do not guarantee control stability. A Markovian model developed for the UArm II manipulator was used to calculate an $\mathcal{H}_\infty$ Markovian controller which kept the stability of the system even when a fault occurs. It was considered in the Markovian model only the occurrence of fault in the second joint, but this procedure can be easily extended to the general case where a fault can occur in any joint.

APPENDIX

The matrices $M(q)$ and $C(q,\dot{q})$ for manipulator utilized in the experiments are given by

$$M(q)=\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$\begin{aligned}
M_{11} =& m_1 l_{c_1}^2 + m_2(l_1^2 + l_{c_2}^2 + 2 l_1 l_{c_2}\cos(q_2)) \\
&+ m_3(l_1^2 + l_2^2 + l_{c_3}^2 + 2 l_1 l_2 \cos(q_2) + 2 l_2 l_{c_3}\cos(q_3)) \\
&+ 2 m_3 l_1 l_{c_3}\cos(q_2+q_3) + I_1 + I_2 + I_3 \\
M_{12} =& m_2(l_{c_2}^2 + 2 l_1 l_{c_2}\cos(q_2)) + m_3(l_2^2 + l_1 l_2 \cos(q_2)) \\
&+ m_3(l_{c_3}^2 + l_1 l_{c_3}\cos(q_2+q_3) + 2 l_2 l_{c_3}\cos(q_3)) \\
&+ I_2 + I_3 \\
M_{13} =& I_3 + m_3(l_{c_3}^2 + l_1 l_{c_3}\cos(q_2+q_3) + l_2 l_{c_3}\cos(q_3)) \\
M_{21} =& M_{12} \\
M_{22} =& I_2 + I_3 + m_2(l_{c_2}^2) + m_3(l_2^2 + l_{c_3}^2 + 2 l_2 l_{c_3}\cos(q_3)) \\
M_{23} =& I_3 + m_3(l_{c_3}^2 + l_2 l_{c_3}\cos(q_3)) \\
M_{31} =& M_{13} \\
M_{32} =& M_{23} \\
M_{33} =& I_3 + m_3(l_{c_3}^2)
\end{aligned}$$

and

$$C(q,\dot{q})=\begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}$$

$$\begin{aligned}
C_{11} =& -[(m_2 l_1 l_{c_2}\sin(q_2) + m_3 l_1 l_2 \sin(q_2) \\
&+ m_3 l_1 l_{c_3}\sin(q_2+q_3))\dot{q}_2 + (m_3 l_1 l_{c_3}\sin(q_2+q_3) \\
&+ m_3 l_2 l_{c_3}\sin(q_3))\dot{q}_3] \\
C_{12} =& -[(m_2 l_1 l_{c_2}\sin(q_2) + m_3 l_1 l_2 \sin(q_2) \\
&+ m_3 l_1 l_{c_3}\sin(q_2+q_3))(\dot{q}_1+\dot{q}_2) \\
&+ (m_3 l_1 l_{c_3}\sin(q_2+q_3) + m_3 l_2 l_{c_3}\sin(q_3))\dot{q}_3]
\end{aligned}$$

$$
\begin{aligned}
C_{13} &= -\left[(m_3 l_1 l_{c_3}\sin(q_2+\theta_3)+m_3 l_2 l_{c_3}\sin(\theta_3))\right.\\
&\qquad \left.(\dot{q}_1+\dot{q}_2+\dot{q}_3)\right]\\
C_{21} &= (m_2 l_1 l_{c_2}\sin(q_2)+m_3 l_1 l_2\sin(q_2)\\
&\qquad +m_3 l_1 l_{c_3}\sin(q_2+q_3))\dot{q}_1 - m_3 l_2 l_{c_3}\sin(q_3)\dot{q}_3\\
C_{22} &= -m_3 l_2 l_{c_3}\sin(q_3)\dot{q}_3\\
C_{23} &= -m_3 l_2 l_{c_3}\sin(q_3)(\dot{q}_1+\dot{q}_2+\dot{q}_3)\\
C_{31} &= (m_3 l_1 l_{c_3}\sin(q_2+q_3)+m_3 l_2 l_{c_3}\sin(q_3))\dot{q}_1\\
&\qquad +m_3 l_2 l_{c_3}\sin(q_3)\dot{q}_3\\
C_{32} &= m_3 l_2 l_{c_3}\sin(q_3)(\dot{q}_1+\dot{q}_2)\\
C_{33} &= 0
\end{aligned}
$$

where m_i, l_i, l_{c_i}, I_i, q_i, and $\dot{q}_i$, are the mass, the length, the center of mass, the inertia momentum, the angular position, and the angular velocity of the i-link, respectively.

REFERENCES

[1] H. Arai and S. Tachi, "Position control of a manipulator with passive joints using dynamic coupling," *IEEE Trans. Robot. Automat.*, vol. 7, pp. 528–534, Aug. 1991.

[2] H. Arai, K. Tanie, and N. Shiroma, "Feedback control of a 3-dof planar underactuated manipulator," in *Proc. 1997 IEEE Int. Conf. Robotics and Automation*, Albuquerque, NM, 1997, pp. 703–709.

[3] ——, "Time-scaling control of an underactuated manipulator," in *Proc. 1998 IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, 1998, pp. 2619–2626.

[4] H. Arai, K. Tanie, and S. Tachi, "Dynamic control of a manipulator with passive joints in operation space," *IEEE Trans. Robot. Automat.*, vol. 9, pp. 85–93, Feb. 1993.

[5] T. Basar and P. Berhard, *$\mathcal{H}_\infty$-Optimal Control and Related Minimax Problems*. Cambridge, MA: Birkhauser, 1990.

[6] M. Bergerman, "Dynamics and control of underactuated manipulators," Ph.D. dissertation, Carnegie Mellon Univ., Pittsburgh, PA, 1996.

[7] E. K. Boukas and A. Benzaouia, "Stability of discrete-time linear systems with Markovians jumping parameters and constrained control," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 516–521, Mar. 2002.

[8] Y. C. Chang and B. S. Chen, "A nonlinear adaptive $\mathcal{H}_\infty$ tracking control design in robotic systems via neural networks," *IEEE Trans. Contr. Syst. Technol.*, vol. 5, pp. 13–29, Jan. 1997.

[9] B. S. Chen, Y. C. Chang, and T. C. Lee, "Adaptive control in robotic systems with $\mathcal{H}_\infty$ tracking performance," *Automatica*, vol. 33, no. 2, pp. 227–234, 1997.

[10] B. S. Chen, T. S. Lee, and J. H. Feng, "A nonlinear $\mathcal{H}_\infty$ control design in robotic systems under parameter perturbation and external disturbance," *Int. J. Control*, vol. 59, no. 2, pp. 439–461, 1994.

[11] O. L. V. Costa and J. B. R. do Val, "Full information $\mathcal{H}_\infty$-control for discrete-time infinite Markov jump parameter systems," *J. Math. Anal. Appl.*, vol. 202, pp. 578–603, 1996.

[12] O. L. V. Costa, E. O. A. Filho, E. K. Boukas, and R. P. Marques, "Constrained quadratic state feedback control of discrete -time Markovian jump linear systems," *Automatica*, vol. 35, pp. 617–626, 1999.

[13] A. de Luca, R. Mattone, and G. Oriolo, "Stabilization of underactuated robots: Theory and experiments for a planar 2r manipulator," in *Proc. 1997 IEEE Int. Conf. Robotics and Automation*, Albuquerque, NM, 1997, pp. 3274–3280.

[14] Y. Huang and A. Jadbabaie, "Nonlinear $\mathcal{H}_\infty$ control: An enhanced quasilpv approach," in *Proc. Workshop $\mathcal{H}_\infty$ Nonlinear Control, IEEE Int. Conf. Decision and Control*, Tampa, FL, 1998.

[15] M. C. Hwang and X. Hu, "A robust position/force learning controller of manipulators via nonlinear $\mathcal{H}_\infty$ control and neural networks," *IEEE Trans. Syst., Man, Cybern. B*, vol. 30, pp. 310–321, Apr. 2000.

[16] Y. Ji, H. J. Chizeck, X. Feng, and K. A. Loparo, "Stability and control of discrete-time jump linear systems," *Control Theory Adv. Technol.*, vol. 7, pp. 247–270, 1991.

[17] R. Johansson, "Quadratic optimization of motion coordination and control," *IEEE Trans. Automat. Contr.*, vol. 35, pp. 1197–1208, Nov. 1990.

[18] R. P. Marques, "Algoritmos De Controle Para Sistemas Sujeitos a Saltos Markovianos," Ph.D. dissertation, Elect. Eng. Dept., Univ. São Paulo, São Paulo, Brazil, 1997.

[19] I. Postlethwaite and A. Bartoszewicz, "Application of nonlinear $\mathcal{H}_\infty$ control to the tetrabot robot manipulator," *Proc. Inst. Mech. Eng. Part I: J. Syst. Contr. Eng.*, vol. 212, no. 16, pp. 459–465, 1998.

[20] A. A. G. Siqueira and M. H. Terra, "Control of underactuated manipulators using nonlinear $\mathcal{H}_\infty$ techniques," in *Proc. IEEE Int. Conf. Decision Control*, Las Vegas, NV, 2002, pp. 2032–2037.

[21] M. H. Terra and R. Tinós, "Fault detection and isolation in robotic manipulators via neural networks – A comparison among three architectures for residual analysis," *J. Robot. Syst.*, vol. 18, pp. 357–374, 2001.

[22] F. Wu, X. H. Yang, A. Packard, and G. Becker, "Induced $\mathcal{L}_2$ -norm control for lpv systems with bounded parameter variation rates," *Int. J. Robust Nonlinear Contr.*, vol. 6, pp. 983–998, 1996.

[23] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

Adriano Almeida Gonçalves Siqueira was born in Botelhos, Brazil, in 1976. He received the B.S. degree in mechanical engineering and the Ph.D. degree in electrical engineering, both from the University of São Paulo (USP), São Carlos, Brazil, in 1999 and 2004, respectively.

His research interests include underactuated robots, cooperative robots, robust control and nonlinear control. He recieved the Institute of Engineering Prize, Brazil, in 1999, the Laurea of Academic Excellence of the University of São Paulo, Brazil, in 1999, and the Engineering, Architecture, and Agronomy Regional Council Prize, Brazil, in 1999. He was a finalist of the Best Student Paper Award of the Conference on Decision and Control in 2002 and a finalist of the Best Student Paper Award , IEEE Conference on Control Applications, CCA, in 2004.

Marco Henrique Terra (M'96) received the Ph.D. degree in electrical engineering from the University of São Paulo (USP), São Carlos, in 1995.

He is currently an Associate Professor of electrical engineering at USP. His research interests comprehend filtering, estimation and control theories, fault detection and isolation problems, and robotics.