

- [9] J. M. Grasmeyer and M. T. Keennon, "Development of the black widow micro air vehicle," in *Proc. 39th Aerosp. Sci. Meeting Exhib.*, 2001, pp. 1–9.
- [10] S. M. Ettinger, M. C. Nechyba, P. G. Ifju, and M. Waszak, "Vision-guided flight stability and control for micro air vehicles," *Adv. Robot.*, vol. 17, no. 3, pp. 617–640, 2003.
- [11] H. Wu, D. Sun, and Z. Zhou, "Model identification of a micro air vehicle in loitering flight based on attitude performance evaluation," *IEEE Trans. Robot.*, vol. 20, no. 4, pp. 702–712, Aug. 2004.
- [12] The cyclogyros: Planned paddle-wheel aeroplanes. (2005). [Online]. Available: <http://www.dself.dsl.pipex.com/MUSEUM/TRANSPORT/cyclogyro/cyclogyro.htm>.
- [13] T. Hase, "A cyclogyro-based flying robot," in *Proc. SVBL Conf. UEC*, Jul. 2001, p. 12.
- [14] T. Hase, T. Suzuki, K. Tanaka, and T. Emaru, "A flying robot with variable attack angle mechanism," in *Proc. 21st Annu. Conf. Robot. Soc. Jpn.*, Tokyo, Japan, May 2003, CD-ROM, Paper 3B22.
- [15] Y. Higashi, K. Tanaka, T. Emaru, and H. O. Wang, "Development of a cyclogyro-based flying robot with variable attack angle mechanisms," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Beijing, China, Oct. 2006, pp. 3261–3266.

Computationally Efficient Predictive Robot Control

Vincent Duchaine, Samuel Bouchard, and Clément M. Gosselin

Abstract—Conventional linear controllers (PID) are not really suitable for the control of robot manipulators due to the highly nonlinear behavior of the latter. Over the last decades, several control methods have been proposed to circumvent this limitation. This paper presents an approach to the control of manipulators using a computationally-efficient-model-based predictive control scheme. First, a general predictive control law is derived for position tracking and velocity control, taking into account the dynamic model of the robot, the prediction and control horizons, and also the constraints. However, the main contribution of this paper is the derivation of an analytical expression for the optimal control to be applied that does not involve a numerical procedure, as opposed to most predictive control schemes. In the last part of the paper, the effectiveness of the approach for the control of a nonlinear plant is illustrated using a direct-drive pendulum, and then, the approach is validated and compared to a PID controller using an experimental implementation on a 6-DOF cable-driven parallel manipulator.

Index Terms—Cable-driven mechanism, nonlinear control, parallel mechanism, position control, predictive control, robot manipulators, velocity control.

I. INTRODUCTION

The majority of existing industrial manipulators are controlled using proportional derivative (PD) controllers. This type of basically linear control does not represent an optimal solution for the control of robots, which exhibit highly nonlinear kinematics and dynamics. In fact, in order to accommodate configurations where the gravity and inertia terms reach their minimum amplitude, the gain associated with the derivative feedback (D) has to be set to a relatively large value, thereby leading to a

generally overdamped behavior that limits the performance. Nevertheless, in most current robotic applications, PD controllers are functional and sufficient due to the high reduction ratio of the transmissions used. However, this assumption is no longer valid for manipulators with low transmission ratios or those intended to perform high accelerations, such as parallel robots.

Therefore, several other approaches have been proposed by researchers. These alternative control schemes can be classified into two main categories [1], namely: 1) dynamic control, which is based on a rigid-body dynamic model of the robot and 2) adaptive control, which is based on an online adjustment of the dynamics of the system or its controller. Examples of dynamic control schemes (computed torque) are given for instance in [2]–[4], while examples of adaptive control methods are provided in [5]–[7]. Although these control schemes represent significant improvements over classical PD controllers, many of them are not suitable for an industrial context due to their lack of robustness with respect to model uncertainties (e.g., variable payload) or due to their computational complexity.

Over the last few decades, a new class of control approach based on the so-called Model Predictive Control (MPC) algorithm was proposed. Arising from the work of Kalman [8], [9] in the 1960s, predictive control can be said to provide the possibility of controlling a system using a proactive rather than reactive scheme. Common applications of this approach are in slow processes such as the petroleum and chemical industries or more recently in aeronautics and aerospace control [10]. A little more than a decade ago, it was also proposed to apply predictive control to nonlinear robotic systems [11], [12]. However, in the latter references, only a restricted form of predictive control was presented and the implementation issues—including the computational burden—were not addressed. Later, predictive control was applied to a broader variety of robotic systems such as a 2-DOF serial manipulator [13], robots with flexible joints [14], or electrical motor drives [15]. More recently, simplified approach using a limited Taylor expansion was presented in [16] and [17]. Due to their relatively low computation time, the latter open the avenue for real implementations. Finally, Poinet and coworkers [18]–[20] experimentally demonstrated predictive control on a 4-DOF parallel mechanism using a linear model in the optimization combined with a feedback linearization.

This paper presents a new simplified approach of predictive control applied to general robotic manipulators that includes directly the complete dynamic model into the cost function. The proposed approach is first derived for velocity control schemes. A general solution is presented, and then, a simplified approach in which no online optimization is required is developed. Then, the approach is derived for the position tracking problem (position control) using similar assumptions. Finally, an experimental validation of the proposed predictive approach is presented to illustrate its performance. Experimental results obtained from the implementation on a 1-DOF pendulum and on a 6-DOF cable-driven parallel mechanism are included.

II. VELOCITY CONTROL

Velocity control is rarely implemented in conventional industrial manipulators since the majority of the tasks to be performed by robots require precise position tracking. However, over the last few years, several researchers have developed a new generation of robots that are capable of working in collaboration with humans [21]–[23]. For this type of tasks, velocity control seems more appropriate [24] due to the fact that the robot is not constrained to given positions, but rather has to follow the movement of the human collaborator. The predictive control approach presented in this paper can be useful in this context.

Manuscript received October 13, 2006; revised May 7, 2007. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) and in part by the Canada Research Chair Program. Recommended by Technical Editor Z. Lin.

The authors are with the Department of Mechanical Engineering, Université Laval, Quebec City, QC G1K 7P4, Canada (e-mail: vincent.duchaine.1@ulaval.ca; samuel.bouchard.1@ulaval.ca; gosselin@gmc.ulaval.ca).

Digital Object Identifier 10.1109/TMECH.2007.905722

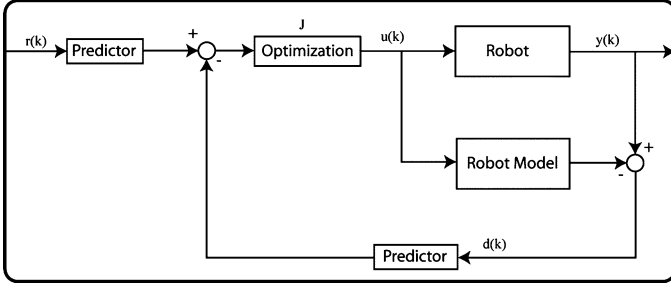


Fig. 1. MPC applied to manipulator.

A. General Formulation

The proposed approach is based on a conventional-model-based predictive control strategy. In this scheme, in order to predict the correct control input to be applied to the system, it is required to minimize a quadratic cost function over a prediction horizon. The cost function is composed of two parts, namely, a quadratic function of the deterministic and stochastic components of the process and a quadratic function of the constraints. The function to be minimized can be written as

$$J = \sum_{n=1}^{H_p} (\mathbf{y}_{k+n} - \mathbf{r}_{k+n})^T \mathbf{Q} (\mathbf{y}_{k+n} - \mathbf{r}_{k+n}) + \sum_{m=1}^{H_c} \psi^T \lambda \psi \quad (1)$$

where

- k the current time step;
- H_p the prediction horizon;
- H_c the control horizon;
- \mathbf{Q} a weighting factor;
- λ a weighting factor;
- \mathbf{r}_{k+n} the reference input;
- \mathbf{y}_{k+n} the output of the system;
- \mathbf{u}_{k+m} the control input signal;
- ψ a constraint function.

In this paper, a constraint on the variation of the control input signal over a prediction horizon will be used

$$\psi = (\mathbf{u}_{k+m} - \mathbf{u}_{k+m-1}) \quad (2)$$

which amounts to minimizing the variation between two consecutive inputs in order to obtain a smoother response. Fig. 1 provides a schematic representation of the proposed scheme, where \mathbf{d}_k represents the error between the output of the system and the output of the model.

For velocity control, the reference input is usually relatively constant, especially considering the high servo rates used. Therefore, it is reasonable to assume that the reference velocity remains constant over the prediction horizon. With this assumption, the stochastic predictor of the reference velocity becomes

$$\begin{aligned} \tilde{\mathbf{r}}(k+1) &= \mathbf{r}(k) \\ \tilde{\mathbf{r}}(k+2) &= \mathbf{r}(k) \\ &\vdots \\ \tilde{\mathbf{r}}(k+H_c) &= \mathbf{r}(k) \end{aligned} \quad (3)$$

where $\tilde{\mathbf{r}}(j)$ stands for the predicted value of \mathbf{r} at time step j .

The error \mathbf{d} is obtained by computing the difference between the system's output and the model's output. Taking into account this difference in the cost function will help to increase the robustness of the control to model mismatch. The error can be decomposed in two parts. The first one is the error associated directly with model uncertainties. Often, this component will produce an offset proportional to the mismatch. The error may also include a zero-mean white noise given by the noise of the encoder or some random perturbation that cannot be included in the deterministic model. Since the error term is partially composed of zero-mean white noise, it is difficult to define a good stochastic predictor of the future values. However, in the case considered here, a future error equal to the present one will be simply assumed. This can be expressed as

$$\begin{aligned} \tilde{\mathbf{d}}(k+1) &= \mathbf{d}(k) \\ \tilde{\mathbf{d}}(k+2) &= \mathbf{d}(k) \\ &\vdots \\ \tilde{\mathbf{d}}(k+H_c) &= \mathbf{d}(k) \end{aligned} \quad (4)$$

where $\tilde{\mathbf{d}}(j)$ is the predicted value of \mathbf{d} at time step j .

B. Application To Manipulators

The dynamic model of a robot manipulator can be expressed as

$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{V}\dot{\boldsymbol{\theta}} + \mathbf{g}(\boldsymbol{\theta}) \quad (5)$$

where

- $\boldsymbol{\tau}$ the vector of actuator torques;
- $\mathbf{M}(\boldsymbol{\theta})$ the generalized inertia matrix;
- $\ddot{\boldsymbol{\theta}}$ the vector of actuator accelerations;
- $\mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ the vector of centrifugal and Coriolis forces;
- $\mathbf{V}\dot{\boldsymbol{\theta}}$ the vector of viscous friction torques;
- $\mathbf{g}(\boldsymbol{\theta})$ the vector of gravity torques.

The acceleration resulting from a torque applied on the system can be found by inverting (5), which leads to

$$\ddot{\boldsymbol{\theta}} = -\mathbf{M}(\boldsymbol{\theta})^{-1}[\mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{V}\dot{\boldsymbol{\theta}} + \mathbf{g}(\boldsymbol{\theta}) - \boldsymbol{\tau}] \quad (6)$$

where $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$ are the positions and velocities measured by the encoders. Assuming that the acceleration is constant over one time period, the previous expression can be substituted into the equations associated with the motion of a body undergoing constant acceleration, which leads to

$$\dot{\boldsymbol{\theta}}_{k+1} = \dot{\boldsymbol{\theta}}_k - \mathbf{M}(\boldsymbol{\theta})^{-1}[\mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{V}\dot{\boldsymbol{\theta}} + \mathbf{g}(\boldsymbol{\theta}) - \boldsymbol{\tau}]T_s \quad (7)$$

where T_s is the sampling period. Since robots usually run on a discrete controller with a very small sampling period, assuming a constant acceleration over a sample period is a reasonable approximation that will not induce significant errors.

Since the system input is a current and not a torque, the aforementioned equations must be modified. Neglecting the dynamic time response of the actuators—which is justified by the high bandwidth of the latter—one can define a matrix \mathbf{K}_τ relating the current and the torque at the actuators. Thus, let us suppose firstly that the relation between the torque and the current for an actuator can be approximated as a simple gain K_m . Since actuators are generally coupled with a transmission, the generalized torque constant will be

$$\mathbf{K}_\tau = \mathbf{K}_m \quad (8)$$

where \mathbf{K}_m is a diagonal matrix whose i th diagonal entry is the product of the gear ratio and the torque to current gain of the i th actuator. Substituting (8) into (7) leads to

$$\dot{\theta}_{k+1} = \dot{\theta}_k - \mathbf{M}(\theta)^{-1}[\mathbf{h}(\theta, \dot{\theta}) + \mathbf{V}\dot{\theta} + \mathbf{g}(\theta) - \mathbf{K}_\tau \mathbf{u}]T_s \quad (9)$$

where \mathbf{u} represents the vector of control input (current in the actuators).

Equation (9) represents the behavior of the robot over a sampling period. However, in predictive control, this behavior must be determined over a number of sampling periods given by the horizon of prediction. Since the dynamic model of the manipulator is nonlinear, it is not straightforward to compute the necessary recurrence over this horizon, especially considering the limited computational time available. This is the main reason why predictive control is still not commonly used for manipulator control.

Instead of computing exactly the nonlinear evolution of the manipulator dynamics, it can be more efficient to make some assumptions that will simplify the calculations. For the accelerations normally encountered in most manipulator applications, the gravitational term is the one that has the most impact on the dynamic model. The evolution of this term over time is a function of the position. The position is obtained by integrating the velocity over time. Even a large variation of velocity will not lead to a significant change of position since it is integrated over a very short period of time. From this point of view, the high sampling rate that is typically used in robot controllers allows us to assume that the nonlinear terms of the dynamic model are constant over a prediction horizon. Obviously, this assumption will induce some error, but this error can easily be managed by the error term included in the minimization.

It is known from the literature that, for an infinite prediction horizon and for a stabilizable process, as long as the objective function weighting matrices are positive definite, predictive control will always stabilize the system [25]. However, the simplifications that have been made earlier on the representation of the system prevent us from concluding on stability since the errors in the model will increase nonlinearly with an increasing prediction horizon. It is not trivial to determine the duration of the prediction horizon that will ensure the stability of the control method. The latter will depend on the dynamic model, the geometric parameters, and also on the conditioning of the manipulator at a given pose.

From the earlier derivations, combining the deterministic and stochastic components and the constraint on the input variable leads to the general cost function to be optimized as a function of the prediction and control horizons. This function can be divided into two sums in order to manage distinctively the prediction horizon and control horizon. One has

$$J = \sum_{n=1}^{H_c-1} [\mathbf{A}(n)^T \mathbf{Q} \mathbf{A}(n) + \Delta \mathbf{u}_n^T \lambda \Delta \mathbf{u}_n] + \sum_{n=H_c}^{H_p} [\mathbf{B}(n)^T \mathbf{Q} \mathbf{B}(n) + \Delta \mathbf{u}_{H_c}^T \lambda \Delta \mathbf{u}_{H_c}] \quad (10)$$

with

$$\mathbf{A}(n) = \dot{\theta} + n\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u}_n - \mathbf{h}_N(\theta, \dot{\theta}))T_s - (\mathbf{r} - \mathbf{d}) \quad (11)$$

$$\mathbf{B}(n) = \dot{\theta} + n\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u}_{H_c} - \mathbf{h}_N(\theta, \dot{\theta}))T_s - (\mathbf{r} - \mathbf{d}) \quad (12)$$

being the integration form for $\dot{\theta}_{k+n}$ of the linear equation (9) and where

- $\mathbf{h}_N(\theta, \dot{\theta})$ the gravity, friction, centripetal, and Coriolis term;
- \mathbf{Q} diagonal matrix of weighting factors;
- λ diagonal matrix of weighting factors;

- \mathbf{u} the vector of input variables (current);
- $\Delta \mathbf{u}_j$ $\mathbf{u}_j - \mathbf{u}_{j-1}$.

An explicit solution to the minimization of J can be found for given values of H_p and H_c . However, it is more difficult to find a general solution that would be a function of H_p and H_c . Nevertheless, a minimum of J can easily be found numerically. From (10), it is clear that J is a quadratic function of \mathbf{u} . Moreover, because of its physical meaning, the minimum of J is reached when the derivative of J with respect to \mathbf{u} is equal to zero. The problem can, thus, be reduced to finding the root of the following equation:

$$\frac{\partial J}{\partial \mathbf{u}} = \sum_{n=1}^{H_c-1} ([\mathbf{C}(n)\mathbf{D}(n) - (\mathbf{r} - \mathbf{d})]T_s) + 2\lambda\Delta \mathbf{u}_n + \sum_{n=H_c}^{H_p} ([\mathbf{C}(n)\mathbf{E}(n) - (\mathbf{r} - \mathbf{d})]T_s) + 2\lambda\Delta \mathbf{u}_{H_c} = 0 \quad (13)$$

with

$$\mathbf{C}(n) = 2n\mathbf{Q}\mathbf{K}_m \quad (14)$$

$$\mathbf{D}(n) = (\dot{\theta} + n\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u}_n - \mathbf{h}_N(\theta, \dot{\theta}))T_s) \quad (15)$$

$$\mathbf{E}(n) = (\dot{\theta} + n\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u}_{H_c} - \mathbf{h}_N(\theta, \dot{\theta}))T_s) \quad (16)$$

An exact and unique solution to this equation exists since it is linear. However, the computation of the solution involves the resolution of a system of linear equation whose size increases linearly with the control horizon. Another drawback of this approach is that the generalized inertia matrix must be inverted, which can be time-consuming. The next section will present strategies to avoid these drawbacks.

C. Analytical Solution of the Minimization Problem

The previous section provided a general formulation of the MPC applied to robot manipulators with an arbitrary number of DOFs and arbitrary chosen prediction and control horizons. However, in this section, only the prediction horizon will be considered. This simplification of the general approach of the predictive control will make it possible to find an exact expression of the optimal control input signal for any prediction horizon, thereby reducing dramatically the computing time.

Many predictive schemes presented in the literature [11], [16], [17] consider only the prediction horizon and disregard the control horizon, which greatly simplifies the formulation. Also, the constraint imposed on the input variable can be eliminated. At high servo rates, neglecting this constraint does not have a major impact since the input signal does not usually vary much from one period to another. Thus, the aggressiveness of the control (\mathbf{u}) that will result from the elimination of the constraint function can easily be compensated for by the use of a longer prediction horizon. The aforementioned simplifications lead to a new cost function given by

$$J = \sum_{n=1}^{H_p} \mathbf{F}(n)^T \mathbf{F}(n) \quad (17)$$

where

$$\mathbf{F}(n) = \dot{\theta} + n\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u} - \mathbf{h}_N(\theta, \dot{\theta}))T_s - (\mathbf{r} - \mathbf{d}) \quad (18)$$

Computing the derivative of (17) with respect to \mathbf{u} and setting it to zero, a general expression of the optimal control input signal as a function of the prediction horizon is obtained, namely

$$\mathbf{u} = \mathbf{K}_\tau^{-1} \left(\mathbf{h}_N(\theta, \dot{\theta}) - \frac{3\mathbf{M}(\dot{\theta} - \mathbf{r} + \mathbf{d})}{(1 + 2H_p)T_s} \right) \quad (19)$$

The algebraic manipulations that lead to (19) from (17) are summarized in Appendix A.

It is noted that this solution does not require the computation of the inverse of the generalized inertia matrix. Moreover, since the solution is analytical, an online numerical optimization is no longer required.

III. POSITION CONTROL

A. General Formulation

The position tracking scheme of control follows a formulation similar to the one that was presented before for velocity control. The main differences are the stochastic predictor of the future reference position and the deterministic model of the manipulator that must now predict the future positions instead of velocities.

In the velocity control scheme, it was assumed that the reference input was constant over the prediction horizon. This assumption was justified by the high servo rate and by the fact that the velocity does not usually vary drastically over a sampling period even in fast trajectories. However, this assumption cannot be used for position tracking. In particular, in the context of human–robot cooperation, no trajectory is established *a priori*, and the future reference input must be predicted from current positions and velocities. A simple approximation that can be made is to use the time derivative of the reference to linearly predict its future. This can be written as

$$\begin{aligned}\tilde{\mathbf{r}}(k) &= \mathbf{r}(k) \\ \tilde{\mathbf{r}}(k+1) &= \mathbf{r}(k) + \Delta\mathbf{r} \\ &\vdots \\ \tilde{\mathbf{r}}(k+H_c) &= \mathbf{r}(k) + H_c\Delta\mathbf{r}\end{aligned}\quad (20)$$

where $\Delta\mathbf{r}$ is given by

$$\Delta\mathbf{r} = \mathbf{r}(k) - \mathbf{r}(k-1). \quad (21)$$

Since the error term $\mathbf{d}(k)$ is again partially composed of zero-mean white noise, one will consider the future of this error equal to the present. Therefore, (4) is also used here.

B. Application To Manipulators

As shown in the previous section, the velocity can be predicted using (7). Integrating the latter equation once more with respect to time—and assuming constant acceleration—the prediction on the position is obtained as

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \dot{\boldsymbol{\theta}}_k T_s - \frac{1}{2}\mathbf{M}(\boldsymbol{\theta})^{-1}[\mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{V}\dot{\boldsymbol{\theta}} + g(\boldsymbol{\theta}) - \tau]T_s^2. \quad (22)$$

Including the deterministic model and the stochastic part inside the function to be minimized, the general function of predictive control for the manipulator is obtained

$$\begin{aligned}J &= \sum_{n=1}^{H_c-1} [\mathbf{G}(n)^T \mathbf{Q} \mathbf{G}(n) + \Delta\mathbf{u}_n^T \lambda \Delta\mathbf{u}_n] \\ &\quad + \sum_{n=H_c}^{H_p} [\mathbf{L}(n)^T \mathbf{Q} \mathbf{L}(n) + \Delta\mathbf{u}_{H_c}^T \lambda \Delta\mathbf{u}_{H_c}] \quad (23)\end{aligned}$$

with

$$\mathbf{G}(n) = \mathbf{N}(n) + \frac{n^2}{2}\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u}_n - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})) T_s^2 \quad (24)$$

$$\mathbf{L}(n) = \mathbf{N}(n) + \frac{n^2}{2}\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u}_{H_c} - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})) T_s^2 \quad (25)$$

$$\mathbf{N}(n) = \boldsymbol{\theta} + n\dot{\boldsymbol{\theta}}T_s - (\tilde{\mathbf{r}} - \mathbf{d}). \quad (26)$$

Taking the derivative of this function with respect to \mathbf{u} and setting it to zero leads to a linear equation that will give the minimum of the cost function:

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{u}} &= \sum_{n=1}^{H_c-1} (n^2 \mathbf{Q} \mathbf{M}^{-1} \mathbf{K}_\tau \mathbf{G}(n) T_s^2 + 2\lambda \Delta\mathbf{u}_n) \\ &\quad + \sum_{n=H_c}^{H_p} (n^2 \mathbf{Q} \mathbf{M}^{-1} \mathbf{K}_\tau \mathbf{L}(n) T_s^2 + 2\lambda \Delta\mathbf{u}_{H_c}) = 0. \quad (27)\end{aligned}$$

C. Exact Solution To the Minimization

Since the aforementioned result requires the use of a numerical procedure and also the inversion of the inertia matrix, the same assumptions that were made for simplifying the cost function for velocity control will be used again here. These assumptions lead to a simplified predictive control law that allows to find a direct solution to the minimization without using a numerical procedure. This function can be written as

$$J = \sum_{n=1}^{H_p} \mathbf{s}(n)^T \mathbf{s}(n) \quad (28)$$

where

$$\mathbf{s}(n) = \boldsymbol{\theta} + n\dot{\boldsymbol{\theta}}T_s + \frac{n^2}{2}\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u} - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}))T_s^2 - (\tilde{\mathbf{r}} - \mathbf{d}). \quad (29)$$

Setting the derivative of this function with respect to \mathbf{u} equal to zero and after some manipulations summarized in Appendix B, the following optimal solution is obtained:

$$\mathbf{u} = \mathbf{K}_\tau^{-1} \left(\mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \frac{2\mathbf{M}(P_2 \dot{\boldsymbol{\theta}}T_s + \boldsymbol{\theta} - \mathbf{r} + P_3 \Delta\mathbf{r} + \mathbf{d})}{P_1 T_s^2} \right) \quad (30)$$

with

$$P_1 = \frac{(3H_p^2 + 3H_p - 1)}{5} \quad (31)$$

$$P_2 = \frac{3H_p(H_p + 1)}{2(2H_p + 1)} \quad (32)$$

$$P_3 = \frac{-3H_p^2 + H_p + 2}{4H_p + 2} \quad (33)$$

where H_p is the horizon of prediction.

It is again pointed out that the direct solution of the minimization given by (30) does not require the computation of the inverse of the inertia matrix.

IV. EXPERIMENTAL VALIDATION

A. Goal of the Experiment

The predictive control algorithm presented in this paper aims at providing a more accurate control of robots. The first goal of the experiment is, thus, to compare the performance of the predictive controller to the performance of a PID controller on an actual robot. The second objective is to verify that the simplifying assumptions that were made in this paper hold in practice. The argument in favor of the predictive

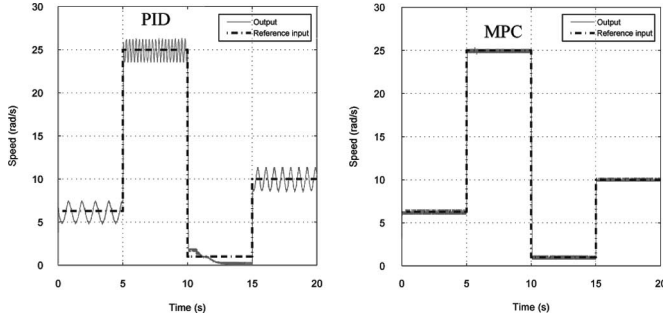


Fig. 2. Speed response of the direct-drive pendulum for PID and MPC control.

controller is that it should lead to better performances than does a PID control scheme since it takes into account the dynamics of the robot and its future behavior while requiring almost the same computation time. In order to illustrate this phenomenon, the control algorithms were first used to actuate a simple 1-DOF pendulum. Then, the position and velocity control were implemented on a 6-DOF cable-driven parallel mechanism. The controllers were implemented on a real-time QNX computer with a servo rate of 500 Hz—a typical servo rate for robotics applications. The PID controllers were tuned experimentally by minimizing the square norm of the error of the motors summed over the entire trajectories.

B. Illustration With the 1-DOF Pendulum

A simple pendulum attached to a direct-drive motor was controlled using a PID scheme and the predictive controller. This system, which represents one of the worst candidates for PID controllers, has been used to demonstrate how our assumption on the dynamical model does not affect the capability of the proposed predictive controller to stabilize nonlinear systems. The use of a direct-drive motor maximizes the impact of the nonlinear terms of the dynamic model, making the system difficult to control by a conventional regulator.

Also, the simplicity of the system helps to obtain accurate estimations of the parameters of the dynamic model that allow to test the ideal case. Despite the fact that its inertia remains constant over time, under constant angular velocity, the gravitational torque is the dominating term in the dynamic model. This setup also makes it possible to test the velocity control at high speed without having to consider angular limitations.

Fig. 2 provides the response of the system (angular velocity) to a given sequence of input reference velocities for the different controllers. The predictive control has been implemented according to (19), and an experimentally determined prediction horizon of four was used for the tests. It can be easily seen that PID control is inappropriate for this nonlinear dynamic mechanism. The sinusoidal error corresponds to the gravitational torque that varies with the rotation of the pendulum. The predictive control follows the reference input more adequately as it anticipates the variation in this term.

C. 6-DOF Cable-Driven Robot Parallel

A 6-DOF cable-driven robot with an architecture similar to the one presented in [26] is used in the experiment. It is shown in Fig. 3 where the frame is a cube with 2 m edges. The end-effector is suspended by six cables. The cables are wound on pulleys actuated by motors fixed at the top of the frame.

1) *Kinematic Modeling*: For a given end-effector pose \mathbf{x} , the necessary cable lengths ρ can be calculated using the inverse kinematic

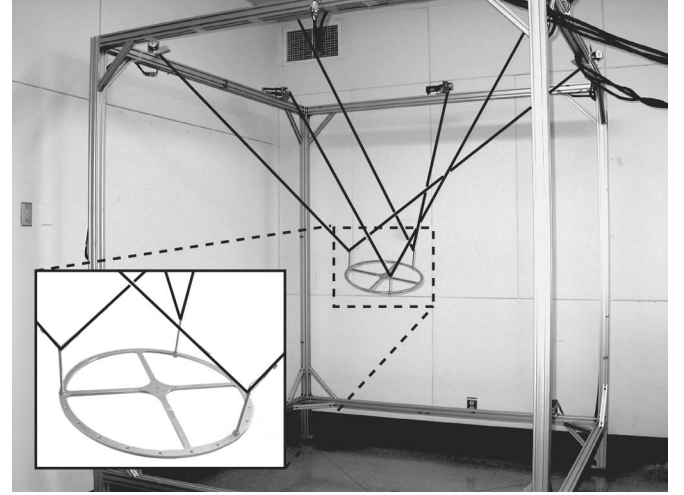


Fig. 3. Cable robot used in the experiment.

problem (IKP). The length of cable i can be calculated by

$$\rho_i^2 = \mathbf{v}_i^T \mathbf{v}_i \quad (34)$$

where

$$\mathbf{v}_i = \mathbf{a}_i - \mathbf{b}_i. \quad (35)$$

In (35), \mathbf{b}_i and \mathbf{a}_i are, respectively, the position of the attachment point of cable i on the frame and on the end-effector, expressed in the global coordinate frame. Thus, vector \mathbf{a}_i can be expressed as

$$\mathbf{a}_i = \mathbf{c} + \mathbf{Q}\mathbf{a}'_i \quad (36)$$

with \mathbf{a}'_i being the attachment point of cable i on the end-effector, expressed in the reference frame of the end-effector and \mathbf{Q} being the rotation matrix expressing the orientation of the end-effector in the fixed reference frame. Vector \mathbf{c} is defined as the position of the reference point on the end-effector in the fixed frame.

Considering a fixed pulley radius r , the lengths of the cable can be related to the angular positions $\boldsymbol{\theta}$ of the actuators

$$\rho = r\boldsymbol{\theta}. \quad (37)$$

Substituting ρ in (34) and differentiating with respect to time, one obtains the velocity equation

$$\mathbf{A}\dot{\mathbf{x}} = \mathbf{B}\dot{\boldsymbol{\theta}} \quad (38)$$

where $\dot{\mathbf{x}}$ is the twist of the end-effector

$$\dot{\mathbf{x}} = [\dot{\mathbf{c}}^T \quad \mathbf{w}^T]^T \quad (39)$$

$$\boldsymbol{\theta} = [\theta_1 \quad \dots \quad \theta_6]^T \quad (40)$$

$$\mathbf{B} = \text{diag}[r^2\theta_1, \dots, r^2\theta_6] \quad (41)$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_6^T \end{bmatrix} \quad (42)$$

vector \mathbf{c}_i being

$$\mathbf{c}_i = \begin{bmatrix} (\mathbf{a}_i - \mathbf{b}_i) \\ (\mathbf{Q}\mathbf{a}'_i) \times (\mathbf{a}_i - \mathbf{b}_i) \end{bmatrix} \quad (43)$$

and where \mathbf{w} is the angular velocity of the end-effector.

2) *Dynamic Modeling*: In this paper, the cables are considered straight, massless, and infinitely stiff. The assumption of straight cables is justified since the robot is small and the mass of the end-effector is much larger than the mass of the cables, which induces no sag. The measurements are made for chosen trajectories for which the mechanism has positive tensions in the cables at all time. The inertia of the wires is negligible compared to the combined inertia of the pulleys and end-effector. Although it is of research interest, the elasticity of the cables is not considered in the dynamic model for this research. The elastic behavior is not exhibited strongly because of the high stiffness of the cables under relatively low accelerations (maximum 9.81 m/s^2) of a 600 g end-effector. The balance between the dynamic properties of the different parts of the robot make these assumptions acceptable.

Equation (38) can be rearranged as

$$\dot{\mathbf{x}} = \mathbf{J}\dot{\boldsymbol{\theta}} \quad (44)$$

where

$$\mathbf{J} = \mathbf{A}^{-1}\mathbf{B}. \quad (45)$$

From (44) and using the principle of virtual work, the following dynamic equation can be obtained:

$$\boldsymbol{\tau} = \mathbf{I}_p \ddot{\boldsymbol{\theta}} + \mathbf{K}_\nu \dot{\boldsymbol{\theta}} + \mathbf{J}^T \mathbf{M}(\ddot{\mathbf{x}} + \mathbf{w}_g) \quad (46)$$

where \mathbf{I}_p is the inertia matrix of the pulleys and motors combined

$$\mathbf{I}_p = \text{diag}[I_{p1}, \dots, I_{p6}] \quad (47)$$

\mathbf{K}_ν is the matrix of the viscous friction at the actuators

$$\mathbf{K}_\nu = \text{diag}[k_{\nu 1}, \dots, k_{\nu 6}] \quad (48)$$

\mathbf{M}_e is the inertia matrix of the end-effector

$$\mathbf{M}_e = \begin{bmatrix} \text{diag}[m_e \ m_e \ m_e] & [\mathbf{0}]_{(3 \times 3)} \\ [\mathbf{0}]_{(3 \times 3)} & \mathbf{I}_e \end{bmatrix} \quad (49)$$

m_e being the mass of the end-effector and \mathbf{I}_e its inertia matrix given by the computer-aided design (CAD) model. Vector \mathbf{w}_g is the wrench applied by gravity to the end-effector.

By differentiating (44) with respect to time, one obtains

$$\ddot{\mathbf{x}} = \dot{\mathbf{J}}\dot{\boldsymbol{\theta}} + \mathbf{J}\ddot{\boldsymbol{\theta}}. \quad (50)$$

Substituting this expression for $\ddot{\mathbf{x}}$ in (46), the dynamics can be expressed with respect to the joint variables $\boldsymbol{\theta}$:

$$\boldsymbol{\tau} = [\mathbf{I}_p + \mathbf{J}^T \mathbf{M} \mathbf{J}] \ddot{\boldsymbol{\theta}} + \mathbf{J}^T \mathbf{M} \dot{\mathbf{J}} \dot{\boldsymbol{\theta}} - \mathbf{K}_\nu \dot{\boldsymbol{\theta}} + \mathbf{J}^T \mathbf{M} \mathbf{w}_g. \quad (51)$$

Equation (51) has the same form as (5) where

$$\mathbf{M}(\boldsymbol{\theta}) = \mathbf{I}_p + \mathbf{J}^T \mathbf{M}_e \mathbf{J} \quad (52)$$

$$\mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{J}^T \mathbf{M}_e \dot{\mathbf{J}} \dot{\boldsymbol{\theta}} \quad (53)$$

$$\mathbf{V} = \mathbf{K}_\nu \quad (54)$$

$$\mathbf{g}(\boldsymbol{\theta}) = \mathbf{J}^T \mathbf{M}_e \mathbf{w}_g. \quad (55)$$

3) *Trajectories*: The trajectories are defined in the Cartesian space. For the experiment, the selected trajectory is a displacement of 0.95 m along the vertical axis performed in 1 s. The displacement follows a fifth-order polynomial with respect to time, with zero speed and acceleration at the beginning and at the end. This smooth displacement is chosen in order to avoid inducing vibrations in the robot since the elasticity of the cables is not taken into account in the dynamic model.

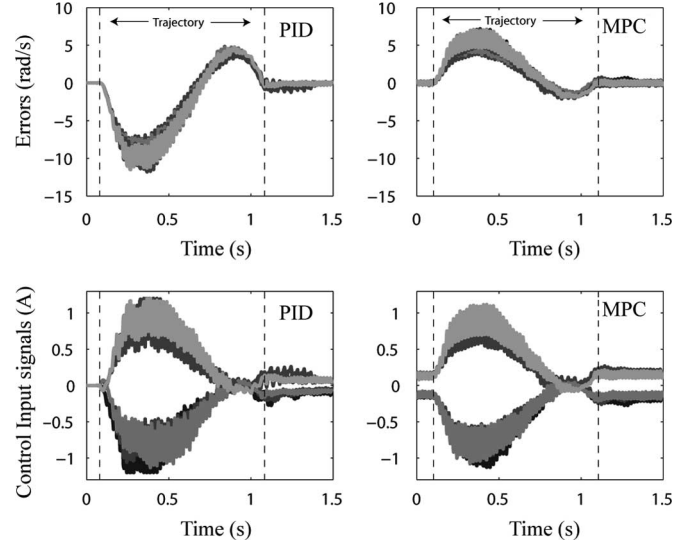


Fig. 4. Velocity control—error between the output and the reference input of the six motors for the PID (top left) and the predictive control (top right). The corresponding control input signals are shown at the bottom.

As mentioned earlier, it was verified prior to the experiment that this trajectory does not require compression in any of the cables.

The cable robot is controlled using the joint coordinates $\boldsymbol{\theta}$, the angular position, and velocity of the pulleys. The Cartesian trajectories are, thus, converted in the joint space using the IKP and the velocity equations. A numerical solution to the direct kinematic problem is also implemented to determine the pose from the information of the encoders. This estimated pose is used to calculate the terms that depend on \mathbf{x} .

D. Experimental Results for the 6-Dof Robot

1) *Velocity Control*: Fig. 4 provides the error between the response of the system (joint velocities) and the time derivative of the joint trajectory described before for the two different controllers. The corresponding control input signals are also shown in this figure. The predictive control algorithm was implemented according to (19), and an experimentally determined horizon of prediction $H_p = 11$ was used.

It can be observed that the magnitude of the error is smaller with the proposed predictive control than with the PID. The control input signals also appears to be smoother with the proposed approach than with the conventional linear controller. The PID suffers from the use of the second derivative of the encoder for the derivative gain (D) that reduces the stability of the control. The predictive control, according to (19) requires only the encoder signal and its first derivative.

2) *Position Control*: A predictive position controller was implemented according to (30). An experimentally determined horizon of prediction $H_p = 14$ was used.

Fig. 5 illustrates the capability of the proposed scheme to perform position tracking compared to a PID. The error over the trajectory for the six motors and the control input signals are presented in the same figure. It can be seen from these figures that the magnitude of the error is in the same range for the two control methods. The main difference occurs at the end of the trajectory where the PID leads to a small overshoot and takes some time to stabilize. Indeed, that is where the predictive control exhibits a clear advantage of performance over the PID. One can also note that during the trajectory, the PID error appears to have a more random distribution and variation than

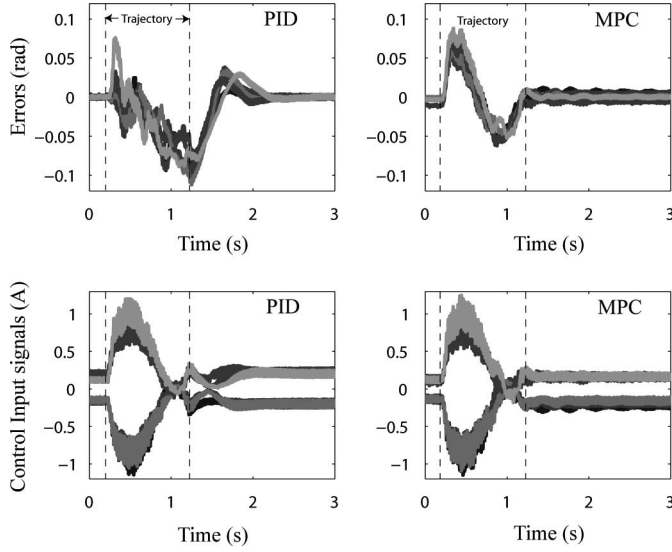


Fig. 5. Position control—error between the output and the reference input of the six motors for the PID (top left) and the predictive control (top right). The corresponding control input signals are shown at the bottom.

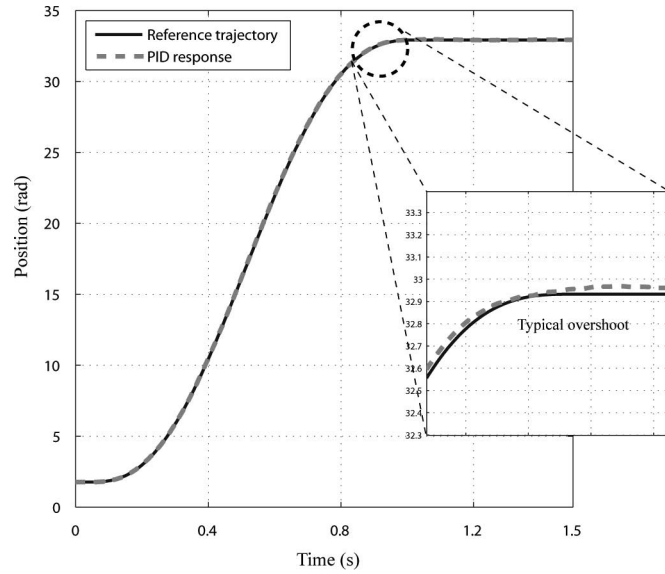


Fig. 6. Angular position trajectory of a motor using the PID.

the errors obtained with the predictive control. In fact, for the latter, the errors follow exactly the velocity profile of the trajectory probably as a consequence of an inaccurate estimation of the friction parameter (\mathbf{K}_v). The PID is tuned to follow the trajectory as closely as possible. Even if the magnitude of the acceleration is the same at the end of the trajectory than at the beginning, the velocity is different. At the beginning, the velocity is small. The errors that feed the PID build up fast enough in a short amount of time to provide a good trajectory tracking. At the end of the trajectory, the velocity is higher, causing this time—with the integrator effect—an overshoot at the end. This type of behavior is common for a PID, and is illustrated with experimental data in Fig. 6. If it is tuned to track closely a trajectory, there is an overshoot at the end. If it is tuned in such a way that there is no overshoot, the tracking is not as good. The predictive controller does not suffer from this problem. It is possible to have a controller that tracks closely a

TABLE I
COMPUTATION TIME REQUIRED FOR EACH CONTROLLER

Controller	Mean Computation time (μs)
PID controller	506
Predictive control	281

fast trajectory, without overshooting at the end. The reason is that the controller anticipates the required torques, the future reference input, and takes into account the dynamics and the actual conditions of the system. Experimentally, another advantage of the predictive controller is that it appears to be much easier to tune than the PID. With a good dynamic model, only the prediction horizon has to be adjusted in order to obtain a controller that is more or less aggressive.

3) *Computation Time*: The computation time was also determined for each controller during position tracking. The results are given in Table I. The PID controller requires a longer calculation time at each step. The integrator in the RT-Lab/QNX computer is the longer term to calculate in the PID. Actually, if it is removed to obtain a PD, the computation time drops to 209 μs . Each controller requires computation times of the same order of magnitude, which means that it is fair to compare them with the same servo rate.

V. CONCLUSION

This paper presented a simplified approach to predictive control adapted to robot manipulators. Control schemes were derived for velocity control as well as position tracking, leading to general predictive equations that do not require online optimization. Several justified simplifications were made on the deterministic part of the typical predictive control in order to obtain a compromise between the accuracy of the model and the computation time. These simplifications can be seen as a means of combining the advantages of predictive control with the simplicity of implementation of a computed torque method and the fast computing time of a PID.

Despite all these simplifications, experimental results on a 6-DOF cable-driven parallel manipulator demonstrated the effectiveness of the method in terms of performance. The method using the exact solution of the optimal control appears to alleviate two of the main drawbacks of predictive control for manipulators, namely, the complexity of the implementation and the computational burden.

Further investigations should focus on the stability analysis using Lyapunov functions and also on the demonstration of the robustness of the proposed control law.

APPENDIX A

EQUATION (19) OBTAINED FROM THE DERIVATIVE OF (17)

$$J = \sum_{n=1}^{H_p} (\mathbf{F}(n)^T \mathbf{F}(n)) \quad (56)$$

where

$$\mathbf{F}(n) = \dot{\boldsymbol{\theta}} + n\mathbf{M}^{-1}(\mathbf{K}_\tau \mathbf{u} - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}))T_s - (\mathbf{r} - \mathbf{d}). \quad (57)$$

A minimum is obtained when

$$\frac{\partial J}{\partial \mathbf{u}} = \sum_{n=1}^{H_p} \left(\frac{\partial \mathbf{F}(n)}{\partial \mathbf{u}} \right)^T \mathbf{F}(n) = 0. \quad (58)$$

Using (57) then leads to

$$\sum_{n=1}^{H_p} (T_s \mathbf{M}^{-1} \mathbf{K}_t)^T \times (n^2 T_s \mathbf{M}^{-1} (\mathbf{K}_t \mathbf{u} - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})) + n(\dot{\boldsymbol{\theta}} - \mathbf{r} + \mathbf{d})) = 0. \quad (59)$$

Since matrix $(T_s \mathbf{M}^{-1} \mathbf{K}_t)$ is constant, the optimal control signal as a function of the prediction horizon is obtained by finding the root of the vector part of the equation, i.e.,

$$\sum_{n=1}^{H_p} n^2 T_s \mathbf{M}^{-1} \mathbf{K}_t \mathbf{u} = \sum_{n=1}^{H_p} (n^2 T_s \mathbf{M}^{-1} \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - n(\dot{\boldsymbol{\theta}} - \mathbf{r} + \mathbf{d})) \quad (60)$$

$$\sum_{n=1}^{H_p} n^2 \mathbf{u} = \mathbf{K}_t^{-1} \sum_{n=1}^{H_p} \left(n^2 \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - n \frac{\mathbf{M}(\dot{\boldsymbol{\theta}} - \mathbf{r} + \mathbf{d})}{T_s} \right) \quad (61)$$

$$\mathbf{u} = \mathbf{K}_t^{-1} \left(\mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \frac{\sum_{n=1}^{H_p} n}{\sum_{n=1}^{H_p} n^2} \frac{\mathbf{M}(\dot{\boldsymbol{\theta}} - \mathbf{r} + \mathbf{d})}{T_s} \right). \quad (62)$$

Recalling that

$$\sum_{n=1}^{H_p} n = \frac{H_p(H_p + 1)}{2} \quad (63)$$

$$\sum_{n=1}^{H_p} n^2 = \frac{H_p(H_p + 1)(2H_p + 1)}{6} \quad (64)$$

leads to

$$\frac{\sum_{n=1}^{H_p} n}{\sum_{n=1}^{H_p} n^2} = \frac{3}{2H_p + 1}. \quad (65)$$

Substituting in (62), one finally has

$$\mathbf{u} = \mathbf{K}_t^{-1} \left(\mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \frac{3\mathbf{M}(\dot{\boldsymbol{\theta}} - \mathbf{r} + \mathbf{d})}{(1 + 2H_p)T_s} \right). \quad (66)$$

APPENDIX B

EQUATION(30) OBTAINED FROM THE DERIVATIVE OF (28)

$$\mathbf{J} = \sum_{n=1}^{H_p} \mathbf{s}(n)^T \mathbf{s}(n) \quad (67)$$

where

$$\mathbf{s}(n) = \boldsymbol{\theta} + n\dot{\boldsymbol{\theta}}T_s + \frac{n^2}{2}\mathbf{M}^{-1}(\mathbf{K}_t \mathbf{u} - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}))T_s^2 - (\tilde{\mathbf{r}} - \mathbf{d}). \quad (68)$$

A minimum is reached when

$$\frac{\partial \mathbf{J}}{\partial \mathbf{u}} = \sum_{n=1}^{H_p} \left(\frac{\partial \mathbf{s}(n)}{\partial \mathbf{u}} \right)^T \mathbf{s}(n) = 0. \quad (69)$$

Substituting (68) into (69) leads to

$$\sum_{n=1}^{H_p} (T_s^2 \mathbf{M}^{-1} \mathbf{K}_t)^T \left(\frac{n^4}{4} T_s^2 \mathbf{M}^{-1} (\mathbf{K}_t \mathbf{u} - \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})) + \frac{n^3}{2} (\dot{\boldsymbol{\theta}}T_s - \Delta \mathbf{r}) + \frac{n^2}{2} (\boldsymbol{\theta} - \mathbf{r} + \Delta \mathbf{r} + \mathbf{d}) \right) = 0 \quad (70)$$

where $\tilde{\mathbf{r}}$ has been replaced according to (20) by

$$\tilde{\mathbf{r}} = \mathbf{r} + (n - 1)\Delta \mathbf{r}. \quad (71)$$

Since matrix $(T_s \mathbf{M}^{-1} \mathbf{K}_t)$ is constant, the optimal control signal as a function of the prediction horizon is obtained by finding the root of the vector part of the equation, i.e.,

$$\begin{aligned} \sum_{n=1}^{H_p} \frac{n^4}{4} T_s^2 \mathbf{M}^{-1} \mathbf{K}_t \mathbf{u} &= \sum_{n=1}^{H_p} \left(\frac{n^4}{4} T_s^2 \mathbf{M}^{-1} \mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \frac{n^3}{2} (\dot{\boldsymbol{\theta}}T_s - \Delta \mathbf{r}) \right. \\ &\quad \left. - \frac{n^2}{2} (\boldsymbol{\theta} - \mathbf{r} + \Delta \mathbf{r} + \mathbf{d}) \right) \end{aligned} \quad (72)$$

$$\begin{aligned} \mathbf{u} = \mathbf{K}_t^{-1} &\left(\mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \frac{2\mathbf{M}}{T_s^2} \left(\frac{\sum_{n=1}^{H_p} n^3}{\sum_{n=1}^{H_p} n^4} (\dot{\boldsymbol{\theta}} - \Delta \mathbf{r}) \right. \right. \\ &\quad \left. \left. + \frac{\sum_{n=1}^{H_p} n^2}{\sum_{n=1}^{H_p} n^4} (\boldsymbol{\theta} - \mathbf{r} + \Delta \mathbf{r} + \mathbf{d}) \right) \right). \end{aligned} \quad (73)$$

Recalling that

$$\sum_{n=1}^{H_p} n^2 = \frac{H_p(H_p + 1)(2H_p + 1)}{6} \quad (74)$$

$$\sum_{n=1}^{H_p} n^3 = \left(\frac{H_p(H_p + 1)}{2} \right)^2 \quad (75)$$

$$\sum_{n=1}^{H_p} n^4 = \frac{H_p(H_p + 1)(2H_p + 1)(3H_p^2 + 3H_p - 1)}{30} \quad (76)$$

we can define P_1 as

$$P_1 = \frac{\sum_{n=1}^{H_p} n^2}{\sum_{n=1}^{H_p} n^4} = \frac{5}{3H_p^2 + 3H_p + 1}. \quad (77)$$

Similarly, we have

$$\frac{\sum_{n=1}^{H_p} n^3}{\sum_{n=1}^{H_p} n^4} = \frac{15H_p(H_p + 1)}{2(2H_p + 1)(3H_p^2 + 3H_p + 1)} = \frac{P_2}{P_1} \quad (78)$$

where P_2 is given by

$$P_2 = \frac{3H_p(H_p + 1)}{2(2H_p + 1)}. \quad (79)$$

Collecting the $\Delta \mathbf{r}$ terms leads to

$$\left(\frac{1}{P_1} - \frac{P_2}{P_1} \right) = \frac{P_3}{P_1} \quad (80)$$

where

$$P_3 = -\frac{3H_p(H_p + 1)}{2(2H_p + 1)} + 1 = \frac{-3H_p^2 + H_p + 2}{4H_p + 2}. \quad (81)$$

Finally, substituting these three polynomial expressions into (73) leads to

$$\mathbf{u} = \mathbf{K}_\tau^{-1} \times \left(\mathbf{h}_N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \frac{2\mathbf{M}(P_2\dot{\boldsymbol{\theta}}T_s + \boldsymbol{\theta} - \mathbf{r} + P_3\frac{\Delta}{\mathbf{r}} + \mathbf{d})}{P_1T_s^2} \right). \quad (82)$$

REFERENCES

- [1] C. Samson, "Commande non-linéaire robuste des robots manipulateurs," *INRIA, Rapport de recherche*, vol. 1, no. 182, pp. 1–53, 1983.
- [2] M. Uebel, I. Minis, and K. Cleary, "Improved computed torque control for industrial robots," in *Proc. 1992 Int. Conf. Robot. Autom.*, vol. 1, pp. 528–533.
- [3] R. J. Anderson, "Passive computed torque algorithms for robot," in *Proc. 28th Conf. Decis. Control*, 1989, vol. 2, pp. 1638–1644.
- [4] Y. Bestaoui and D. Benmerzouk, "A sensity analysis of the computed torque technique," in *Proc. Amer. Control Conf.*, 1995, vol. 6, pp. 4458–4459.
- [5] J.-J. E. Slotine, "The robust control of robot manipulators," *Int. J. Robot. Res.*, vol. 4, no. 2, pp. 49–64, 1985.
- [6] J.-J. E. Slotine and W. Li, "Adaptive manipulator control a case study," in *Proc. 1987 IEEE Int. Conf. Robot. Autom.*, vol. 4, pp. 1392–1400.
- [7] J.-J. E. Slotine and W. Li, "Adaptive strategies in constrained manipulation," in *Proc. 1987 IEEE Int. Conf. Robot. Autom.*, vol. 4, pp. 595–601.
- [8] R. Kalman, "Contributions to the theory of optimal control," *Bull. Soc. Math. Mex.*, vol. 5, pp. 102–119, 1960.
- [9] R. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, pp. 35–45, 1960.
- [10] J. Shi, A. G. Kelkar, and D. Soloway, "Stable reconfigurable generalized predictive control with application to flight control," *Trans. ASME, J. Dyn. Syst., Meas. Control*, vol. 128, no. 6, pp. 371–378, 2006.
- [11] F. Berlin and P. M. Frank, "Robust predictive robot control," in *Proc. 5th Int. Conf. Adv. Robot.*, 1991, vol. 2, pp. 1493–1496.
- [12] J. M. Compas, P. Decarreau, G. Lanquetin, J. Estival, and J. Richalet, "Industrial application of predictive functional control to rolling mill, fast robot, river dam," in *Proc. 3rd IEEE Conf. Control Appl.*, 1994, vol. 3, pp. 1643–1655.
- [13] Z. Zhang and W. Wang, "Predictive function control of a two link robot manipulator," in *Proc. Int. Conf. Mechatron. Autom.*, 2005, vol. 4, pp. 2004–2009.
- [14] D. von Wissel, R. Nikoukhah, and S. L. Campbell, "On a new predictive control strategy: Application to a flexible-joint robot," in *Proc. 33rd IEEE Conf. Decis. Control*, 1994, vol. 3, no. 14, pp. 3025–3026.
- [15] R. Kennel, A. Linder, and M. Linke, "Generalized predictive control (GPC)—Ready for use in drive application?," in *Proc. 32nd IEEE Power Electron. Spec. Conf.*, 2001, pp. 1839–1844.
- [16] R. Hedjar, R. Toumi, P. Boucher, and D. Dumur, "Finite horizon nonlinear predictive control by the taylor approximation: Application to robot tracking trajectory," *Int. J. Appl. Math. Sci.*, vol. 15, no. 4, pp. 527–540, 2005.
- [17] R. Hedjar and P. Boucher, "Nonlinear receding-horizon control of rigid link robot manipulators," *Int. J. Adv. Robot. Syst.*, vol. 2, no. 1, pp. 15–24, 2005.
- [18] F. Lydoire and P. Poignet, "Non linear model predictive control via interval analysis," in *Proc. 44th IEEE Conf. Decis. Control*, 2005, pp. 3771–3776.
- [19] P. Poignet and M. Gautier, "Nonlinear model predictive control of a robot manipulator," in *Proc. 6th Int. Workshop Adv. Motion Control 2000*, pp. 401–406.
- [20] A. Vivas, P. Poignet, and F. Pierrot, "Predictive functional control for a parallel robot," in *Proc. Int. Conf. Intell. Robots Syst.*, 2003, vol. 3, pp. 2785–2790.
- [21] R. Berbarth, D. Lu, and Y. Dong, "A novel cobot and control," in *Proc. 5th World Congr. Intell. Control*, 2004, vol. 5, pp. 4635–4639.
- [22] M. Peshkin, J. Colgate, W. Wannasupphrasit, C. Moore, and R. Gillespie, "Cobot architecture," *IEEE Trans. Robot. Autom.*, vol. 17, no. 4, pp. 377–390, Aug. 2001.
- [23] O. M. Al-Jarrah and Y. F. Zheng, "Arm-manipulator coordination for load sharing using compliant control," in *Proc. IEEE 1996 Int. Conf. Robot. Autom.*, vol. 2, pp. 1000–1005.
- [24] V. Duchaine and C. Gosselin, "General model of human–robot cooperation using a novel velocity based variable impedance control," in *Proc. IEEE World Haptics*, 2007, pp. 446–451.
- [25] S. Qin and T. Badgwell, "An overview of industrial model predictive control technology," in *Proc. Chem. Process Control V*, 1997, pp. 232–256.
- [26] S. Bouchard and C. M. Gosselin, "Workspace optimization of a very large cable-driven parallel mechanism for a radiotelescope application," presented at the ASME Int. Des. Eng. Tech. Conf., Mech. Robot. Conf., Las Vegas, NV, 2007.