NUMERICALLY GENERATED PATH STABILIZING CONTROLLERS: USE OF PRELIMINARY FEEDBACK

D. VON WISSEL, R. NIKOUKHAH, F. DELEBECQUE AND S. L. CAMPBELL[1]

A hybrid open-loop closed-loop control strategy for path following control problems is introduced. It is based on a strategy due to Jankowski et al., but modified by a preliminary feedback. An analysis is done in the linear case showing when the preliminary feedback is needed and how to compute it. The extension to nonlinear systems is discussed. A nonlinear example is presented showing the necessity of preliminary feedback. The approach is especially useful for complex nonlinear systems with high relative degrees.

1. INTRODUCTION

Given a physical system

$$F(x', x, t, u) = 0 \tag{1.1a}$$

with control u and state x there has been an extensive amount of research on determining u so that x satisfies a path-constraint

$$h(x) - \xi(t) = 0, \tag{1.1b}$$

where $\xi(t)$ is a desired trajectory. The most commonly studied situation is when (1.1a) is an ODE

$$x' = f(x,t) + g(x,t)\,u. \tag{1.2}$$

However, many physical systems, such as constrained mechanical systems, are most naturally initially modeled in the form of (1.1a) where $F_{x'}$ is identically singular. That is, (1.1a) is a system of *Differential Algebraic Equations* (DAEs) or a *Descriptor System*.

For the system consisting of (1.1b) and (1.2), a variety of control algorithms based on system inversion have been discussed in the literature. See [13] and the references therein for an overview and conditions of existence of such a control.

In this paper we are looking at nonlinear systems which have as many inputs as outputs. In this particular case a tracking control exists if (1.2) with the output

[1] Research supported in part by the National Science Foundation under DMS-9122745 and INT-9220802.

defined by (1.1b) is *invertible*; for square systems *left* and *right invertibility* as defined in [6, 7] are identical conditions. However, assume system (1.1) is invertible; if system (1.1) has high relative degree and if the equations are mildly complex, the usual nonlinear approaches can become unwieldy due to computational complexity. Such problems arise, for example, in constrained mechanical systems when actuator dynamics or joint flexibility are included and the number of bodies or links exceeds three. In order to overcome these difficulties Jankowski and Van Brussel proposed a predictive type open-loop closed-loop controller in [10] for the control of a flexible-joint robot. Simulations show that this approach which is a numerical implementation of system inversion, gives promising results. Careful analysis (started in [4]) however shows that this control strategy does not necessarily stabilize the system, a problem which can be remedied with the use of preliminary linear feedback. In this paper we present an analysis of the linear case and propose a construction method for the preliminary feedback, if needed. To show the applicability of this approach to nonlinear systems we present the simulation results of the control of a particular mechanical system with this method. Details related to this example and further analysis of it are presented in [14].

In Section 2 we summarize the approach of [8, 9], explain some of its potential advantages, and delineate the questions to be examined in this paper. In Section 3 we give an analysis of the linear case. Section 4 is devoted to a linear example and in Section 5 we give simulation results of the application of this control strategy to a nonlinear example.

We assume that the reader is familiar with such concepts as the relative degree [5] of a prescribed path control problem and the terminology used when discussing DAEs such as the index [1, 2].

2. THE GENERAL APPROACH

We begin by giving a general description of the method presented in this paper.

Intuitively, the system (1.1) can be converted to an ODE if sufficient differentiation of (1.1b), and sometimes (1.1a), is done. While the index of a nonlinear DAE is a somewhat subtle concept [2], in this paper we can consider the index of a DAE to be the number of times some subset of the DAEs must be differentiated in order to convert the DAE to an ODE. Let ν denote the index of System (1.1) in x and u. Similarly, the control u from the inverse problem (1.1) can often be found by sufficient differentiation of (1.1b) and (1.1a). In this case, the index of (1.1) as a system in x, u is closely related to, but not equivalent to the relative degree of (1.1) [3]. If (1.1a) is in the form of (1.2), then the relative degree is $\nu - 1$. In the linear case, the Silverman algorithm [11] yields ν and an expression for u in terms of x, $\xi(t)$ and derivatives of $\xi(t)$. See [6] for its nonlinear extension which will be refered to as the Hirschorn algorithm.

For many real applications the complexity of the equations involved increases rapidly with increased differentiation. Symbolic elimination and reduction of state dimension can also lead to rapid expansion of expression complexity. The idea behind the approach we are examining is to introduce just enough differentiation so that ν is reduced to the point that the DAE can be safely integrated numerically. While the DAE still possibly has index larger than one, the equations are simpler and quicker to integrate than they would be if they were reduced all the way to index one or zero.

As it is standard practice in control we begin by stabilizing the tracking error term (1.1b), if necessary, in order to have at most an index 3 system (for higher index systems numerical integrations of (1.1) may not converge). This can be done by applying a few steps of the Hirschorn algorithm replacing the differential operator $\frac{\mathrm{d}}{dt}$ by the stable polynomial $\alpha + \frac{\mathrm{d}}{dt}$. If we refer to the Silverman or Hirschorn algorithms we always suppose that this has been done. We shall refer to this system as the partially stabilized system. More specifically, let $W(s)$ be a stable polynomial matrix. Then the partially stabilized system (3.1) is

$$F(x', x, t, u) = 0 \tag{2.1a}$$

$$\widehat{h}(x, u, t) - \widehat{\xi}(t) \stackrel{\text{def}}{=} W\!\left(\frac{\mathrm{d}}{\mathrm{d}t}\right)(h(x) - \xi(t)) = 0. \tag{2.1b}$$

The effects of the partial stabilization were examined in [4]. In the sequel we shall drop the hat and suppose that (2.1) is initially stabilized. In the approach considered here, the numerical solution u of (2.1) is used as an open-loop control in (1.1a). In order to make this control act as a feedback control, the state x in (1.1a) is regularly measured and the integration of (2.1) is restarted with the true value of x after each measurement; for details of the control implementation see [4, 8, 9, 10, 14, 15]. The effectiveness of this method in solving large complex problems relies in large part on efficient implementations of the numerical integrator.

Since this control strategy uses the theory of DAEs or *Descriptor Systems* and since it is *predictive* in its implementation, we will call it *Descriptor Predictive Control* (DPC) throughout the remainder of this paper.

3. DPC FOR LINEAR SYSTEMS

In this section, we consider DPC for linear systems of the form

$$x' = Ax + Bu \tag{3.1a}$$

$$y = Cx + Du. \tag{3.1b}$$

The dimension of the state x is n, the dimensions of y and u are are both m. The goal is to find a stabilizing control $u = F(x, \xi(t), \xi'(t), \ldots, \xi^{(\nu)}(t))$, such that $y(t)$ converges exponentially to $\xi(t)$, the reference trajectory. The path $\xi(t)$ is assumed to be sufficiently smooth. To analyze the properties of DPC applied to system (3.1), we are going to make a few simplifying assumptions. The first assumption is that the solution of the DAE obtained by replacing y by ξ in (3.1) can be constructed without any error. The second assumption is that we consider the case where the time h between two measurements of the state is very small so that we can consider the limiting behavior as h goes to zero. Finally we restrict ourselves in the following linear analysis to the class of invertible linear systems. For the sake of simplicity we shall assume that (3.1) represents the partially stabilized system.

A key result which is needed in our analysis is how to construct the solution of a linear DAE. We use uppercase letters for the solutions of the (partially stabilized) system being numerically integrated as opposed to the original system.

Solution of a linear DAE. Consider the linear DAE

$$X' = AX + BU \tag{3.2a}$$

$$\xi(t) = CX + DU \tag{3.2b}$$

$$X(t_\rho) = x(t_\rho), \tag{3.2c}$$

where $x(t_\rho)$ is known and $\xi(t)$ is a given time function. Clearly (3.2) is the linear version of (2.1). The solution $U(t)$ of (3.2) can be constructed by noting that (3.2) can be expressed as

$$\begin{bmatrix} -sI + A & B \\ C & D \end{bmatrix} \begin{bmatrix} X \\ U \end{bmatrix} = \begin{bmatrix} 0 \\ \xi(t) \end{bmatrix}. \tag{3.3}$$

Let the matrix $V = \begin{bmatrix} V_1 & V_2 \\ V_3 & V_4 \end{bmatrix}$ perform a row-compression of $\begin{bmatrix} B \\ D \end{bmatrix}$. Left multiplication by V changes the system pencil in (3.3) to

$$\begin{bmatrix} -sV_1 + V_1A + V_2C & V_1B + V_2D \\ -sV_3 + V_3A + V_4C & V_3B + V_4D \end{bmatrix} = \begin{bmatrix} -Es + F & 0 \\ -Hs + J & I \end{bmatrix}$$

so that (3.3) becomes

$$\begin{bmatrix} -Es + F & 0 \\ -Hs + J & I \end{bmatrix} \begin{bmatrix} X \\ U \end{bmatrix} = \begin{bmatrix} V_2\xi(t) \\ V_4\xi(t) \end{bmatrix}. \tag{3.4}$$

The pencil $\{E, F\}$ is regular due to the invertibility assumption on (3.2). Thus there exist two matrices M and Q such that

$$M(-Es + F)Q = \operatorname{diag}\{-Is + A_1, -sN + I\} \tag{3.5}$$

is in Kronecker normal form, with N a nilpotent matrix. Left multiplication by Q implies a change of variable $Z = Q^{-1}X$. Obviously, the transformation into Kronecker normal form separates the new state Z into a continuous part Z_1 and a discontinuous or impulsive part Z_2. In the sequel we note $Z_1 = P_1^T Z$ and $Z_2 = P_2^T Z$, where $P_1^T = [I \; 0]$ and $P_2^T = [0 \; I]$. P_1^T determines the projection of X which is continuous at 0, and more importantly the projection of $x(t_\rho)$ which contributes to the solution of (3.2)[1]. Now (3.4) can be rewritten as

$$\begin{bmatrix} M & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} -Es + F & 0 \\ -Hs + J & I \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} Q^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} X \\ U \end{bmatrix} = \begin{bmatrix} MV_2\xi(t) \\ V_4\xi(t) \end{bmatrix}$$

[1] It is exactly this projection that DPC uses to generate the control. Thus, in some sense, the DPC can be thought of as an output feedback controller, the output being $P_1^T x$. This projection may be empty, but each step of the stabilized Hirschorn algorithm applied to the original system prior to the application of the numerical method increases the size of this projection. If the Hirschorn algorithm is carried through completely, then $P_1^T = I$ and (3.2) can be solved exactly by noting that in this case D is invertible.

or equivalently

$$\begin{bmatrix} -sI + A_1 & 0 & 0 \\ 0 & -sN + I & 0 \\ (-HQs + JQ)P_1 & (-HQs + JQ)P_2 & I \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ U \end{bmatrix} = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ V_4\xi(t) \end{bmatrix}, \tag{3.6}$$

where $\gamma_1 = P_1^T M V_2 \xi(t)$ and $\gamma_2 = P_2^T M V_2 \xi(t)$. From (3.6) we obtain an expression for the control at t_ρ^+

$$U(t_\rho^+) = HQP_1Z_1'(t_\rho^+) - JQP_1Z_1(t_\rho^+) + HQP_2Z_2'(t_\rho^+) - JQP_2Z_2(t_\rho^+) + V_4\xi(t_\rho). \tag{3.7}$$

We are only interested in the solution of U at t_ρ^+. DPC actually uses a numerical estimate for $U(t_\rho^+)$ but here we are studying the limiting case. Expression (3.7) does not specify $U(t_\rho^+)$ yet because even though $Z_1(t_\rho^+)$ is known to be $Z_1(t_\rho^+) = Z_1(t_\rho^-) = P_1^T x(t_\rho)$ the quantities $Z_1'(t_\rho^+), Z_2(t_\rho^+)$, and $Z_2'(t_\rho^+)$ are not known. The first two block-rows of (3.6) can be used to compute them. The first row gives $Z_1'(t_\rho^+)$ since $sZ_1 = A_1 Z_1 - \gamma_1$. Now N is nilpotent of index ν so that $(-sN + I)\, Z_2 = \gamma_2$ implies that $Z_2 = \sum_{i=0}^{\nu} (sN)^i \gamma_2$ and $sZ_2 = \sum_{i=0}^{\nu} s(sN)^i \gamma_2$ Substituting these expressions into (3.7) yields an equation depending exclusively on $Z_1(t_\rho^+) = P_1 Q^{-1} x(t_\rho^+)$, $\xi(t)$, and the derivatives of $\xi(t)$.

$$U(t_\rho^+) = (HQP_1A_1 - JQP_1)\, Z_1(t_\rho) + R_\xi \left(\frac{\mathrm{d}}{\mathrm{d}t}\right) \xi(t)|_{t=t_\rho} = K_J X(t_\rho) + R_\xi \left(\frac{\mathrm{d}}{\mathrm{d}t}\right) \xi(t)|_{t=t_\rho}, \tag{3.8}$$

where $K_J = (HQP_1A_1P_1^T - JQP_1P_1^T)Q^{-1}$ and

$$R_\xi(s) = V_4 - \left(HQP_1P_1^T - (sH - J)\, QP_2 \sum_{i=0}^{\nu} (sN)^i P_2^T \right) MV_2 = \sum_{i=0}^{\nu+1} R_i s^i. \tag{3.9}$$

Thus, in the $h \to 0$ limiting case, DPC gives the state-feedback control

$$u(t) \;=\; K_J x(t) + R_\xi \left(\frac{\mathrm{d}}{\mathrm{d}t}\right) \xi(t). \tag{3.10}$$

3.1. Stability analysis

DPC (3.10) is stable if and only if the closed-loop system

$$x' = (A + B\, K_J)\, x \tag{3.11}$$

is stable, which means that the eigenvalues of $A + B\,K_J$ have negative real parts. The stability property clearly is independent of the choice of coordinate system. The control (3.10) is computed from the system with stabilized tracking error but is to be applied to the original system. However, the A, B matrices are the same for both systems. Thus in order to determine the stability properties of $A + BK_J$ it suffices to see what happens if (3.10) is fed back into system (3.1). Accordingly, we analyze the stability of (3.11) in the coordinate system $z = Q^{-1}x$ where Q is defined in the previous section. In z coordinates, (3.1) can be expressed as

$$\begin{bmatrix} z_1' \\ z_2' \end{bmatrix} = \begin{bmatrix} \overline{A}_1 & \overline{A}_2 \\ \overline{A}_3 & \overline{A}_4 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_1 \\ \overline{B}_2 \end{bmatrix} u \tag{3.12a}$$

$$y = \begin{bmatrix} \overline{C}_1 & \overline{C}_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \overline{D}u. \tag{3.12b}$$

Theorem 3.1. Let K_J and Q be as defined in the previous section and let $\overline{K}_J = K_J Q$. Then

a) $\overline{K}_J = \begin{bmatrix} K_1 & 0 \end{bmatrix}$ and the application of $u = \overline{K}_J z$ sets $\overline{C}_1$ and $\overline{A}_3$ to zero, i.e.,

$$\begin{bmatrix} \overline{A}_1 & \overline{A}_2 \\ \overline{A}_3 & \overline{A}_4 \end{bmatrix} + \begin{bmatrix} \overline{B}_1 \\ \overline{B}_2 \end{bmatrix} \overline{K}_J = \begin{bmatrix} \overline{\overline{A}}_1 & \overline{A}_2 \\ 0 & \overline{A}_4 \end{bmatrix} \tag{3.13a}$$

$$\begin{bmatrix} \overline{C}_1 & \overline{C}_2 \end{bmatrix} + \overline{DK}_J = \begin{bmatrix} 0 & \overline{C}_2 \end{bmatrix} \tag{3.13b}$$

b) The eigenvalues of $\overline{\overline{A}}_1$ are the transmission zeros of (3.1).

c) The decomposition (z_1, z_2) isolates the largest output-nulling (A, B)-invariant subspace $\mathcal{V}^*$ of (3.1).

For the proof of the theorem see [12]. For a) and b) see Section III. B. eq. (50), eq. (51) and the following comments. For c) see Section III. A p. 349 and again Section III. B. p. 353.

This Theorem allows us to state the main result of this paper:

Theorem 3.2. Suppose System (3.1) is minimum phase, then the closed-loop system is stable iff $\overline{A}_4$ is a stable matrix. Eigenvalues of the system zeros are unaffected by DPC feedback.

Lemma 3.1. If (A, B) is a controllable pair, thus, so is $(\overline{A}_4, \overline{B}_2)$ and so there exists a matrix K_2 such that $\overline{A}_4 + \overline{B}_2 K_2$ is stable.

If the preliminary feedback $u = K_2 z_2 + v \overset{\text{def}}{=} \overline{K}_n z + v$ is applied before DPC , then System (3.12) becomes

$$\begin{bmatrix} z_1' \\ z_2' \end{bmatrix} = \begin{bmatrix} \overline{A}_1 & \overline{\overline{A}}_2 \\ \overline{A}_3 & \overline{\overline{A}}_4 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} \overline{B}_1 \\ \overline{B}_2 \end{bmatrix} v, \quad y = \begin{bmatrix} \overline{C}_1 & \overline{\overline{C}}_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \overline{D}v, \tag{3.14}$$

where $\overline{\overline{A}}_4 = \overline{A}_4 + \overline{B}_2 K_2$ is stable. If we now apply DPC to (3.1), we obtain a stable closed-loop system.

To summarize the preliminary feedback procedure, we first have to find the coordinate transformation matrix Q as explained in Section 3 and the new representation (3.12). Then test the stability of $\overline{A}_4$, and construct K_2 if necessary such that $\overline{A}_4 + \overline{B}_2 K_2$ is stable. Then let $K_n = \begin{bmatrix} 0 & K_2 \end{bmatrix} Q^{-1}$ and apply the preliminary feedback $u = K_n x + v$ to the original system (3.1). DPC can now be applied to the new system $\{x' = (A + B\,K_n)\,x + B\,v,\ y = (C + D\,K_n)\,x + D\,v\}$ with guaranteed success.

Of course, the preliminary feedback and more generally DPC is not needed for linear systems since in the linear case the Silverman algorithm can be applied. The above analysis is only presented to illustrate the idea. To apply the preliminary feedback idea in the nonlinear case, a simple approach would be to construct the preliminary feedback based on a linearized model of the system around some nominal operation point x^0. There is no guarantee that such a preliminary feedback does the job if the actual trajectory $x(t)$ of the system does not remain close to the nominal operating point x^0. But if DPC does not work or has poor performance such a preliminary feedback may improve the situation as we shall see later in an example.

In the nonlinear case, there may be other ways of constructing a preliminary feedback by taking into account the nonlinearities of the system. This problem is currently under investigation.

3.2. Tracking properties

Just as with the stability analysis, we can study tracking properties in any coordinate system. Let us consider the representation (3.13). It is clear that, after the application of the preliminary feedback if needed, DPC yields the following output $y = (\overline{\overline{C}}_2(s\,I - \overline{\overline{A}}_4)^{-1}\overline{B}_2 + \overline{D})\,R_\xi(s)\,\xi + \ \overline{\overline{C}}_2(s\,I - \overline{\overline{A}}_4)^{-1} z_2(0)$.

Theorem 3.3. $(\overline{\overline{C}}_2(s\,I - \overline{\overline{A}}_4)^{-1}\overline{B}_2 + \overline{D})\,R_\xi(s) = I$ and by construction $\overline{\overline{A}}_4$ is stable. Thus $e(t) = y(t) - \xi(t)$ converges exponentially to zero.

Proof. The system-representation (3.6) in Kronecker normal form is clearly nothing but another representation of (3.1). To compute U we just replaced $y(t)$ by $\xi(t)$. The transfer-matrix of the closed loop system-representations (3.6) and (3.1) is $H_b(s)$, as defined before. Since equation (3.8) is computed using (3.6), the polynomial matrix $R_\xi(s)$ is by construction the polynomial part $P(s)$ of the inverse of the transfer-matrix $H_b(s)^{-1} = P(s) + R(s)$, where $R(s)$ is the proper part of $H(s)$. In fact, the decomposition $(R(s), P(s))$ corresponds to the decomposition (z_1, z_2). Since after application of the feedback-matrix K_J to (3.1), z_1 becomes unobservable, the inverse of $H_b(s)$ is polynomial and we have $H_b(s)^{-1} = R_\xi(s)$. □

Note that $e(t)$ represents the tracking error for the system which is not the original system that we had considered, but the system obtained from possible applications of a few steps of the Hirschorn algorithm to the original system. But, clearly, if this tracking error converges to zero, then so does that of the original system.

4. LINEAR EXAMPLE

In this section we consider the linearized model of a wheel rolling on a plane. A detailed analysis of this example can be found in [14] (see the Appendix for the full nonlinear model). The model can be expressed in terms of the Euler angles $x = [\theta, \psi, \phi', \theta', \psi']^T$. To compute the linear model we linearize the nonlinear system around the nominal trajectory $x_0(t) = \left[\frac{\pi}{2}, -6t, 0, 0, -6\right]^T$ for $t > 0$ to get

$$x' = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -3 & 0 \\ 6.54 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} u \, , \quad y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} x. \tag{4.1}$$

The transfer matrix of (4.1) is $H(s) = \begin{bmatrix} \frac{5}{8.46s+s^3} & 0 \\ 0 & \frac{1}{s^2} \end{bmatrix}$. Note that (4.1) has no transmission zeros, and that the linearized model is completely decoupled (the non-linear model is not completely decoupled). We shall consider two controllers for this system. Both use one step of the Hirschorn algorithm and are applied first without and then with preliminary feedback. The complete solution based on the Hirschorn algorithm would require three steps (the index is four).

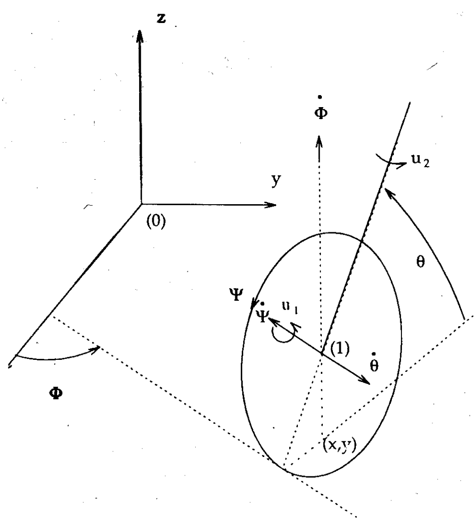

Fig. 5.1. Parametrization of a wheel rolling on a plane.

To apply the first step of the **Hirschorn** algorithm, we let $W(s) = \begin{bmatrix} 5+s & 0 \\ 0 & 5+s \end{bmatrix}$ and obtain the output $\widehat{y} = \begin{bmatrix} 5 & 0 & 0 & 1 & 0 \\ 0 & 5 & 0 & 0 & 1 \end{bmatrix} x$. Clearly the transfer function of the new (partially stabilized) system is $\widehat{H}(s) = W(s)H(s)$ and the system has two transmission zeros at -5. The trajectories to be followed by this new system are

$\widehat{\xi}(t) = W(\frac{\mathrm{d}}{\mathrm{d}t})y(t)$. Matrix V just reorders the rows of the system pencil. We obtain then

$$Q = \begin{bmatrix} -0.4739 & 0 & 0 & 0 & 0 \\ 0 & -0.5266 & 0 & 0 & 0 \\ -1.7498 & 0 & 1.0903 & 0 & -0.4108 \\ 2.3698 & 0 & -0.6619 & 0 & -0.6767 \\ 0 & 2.6332 & 0 & -1 & 0 \end{bmatrix}.$$

For the system in z coordinates we find that $\overline{A}_4 = \left[\begin{smallmatrix} 2.0972 & 0 & 5.2474 \\ 0 & 5 & 0 \\ -5.2162 & 0 & 2.9027 \end{smallmatrix}\right]$ is unstable since its eigenvalues are $\{5, 2.5 \pm 5.2163\, i\}$. To stabilize $\overline{A}_4$ we apply the preliminary feedback $v = \overline{K}_n z$, where

$$\overline{K}_n = \begin{bmatrix} 0 & 0 & -7.5455 & 0 & 15.16919 \\ 0 & 0 & 0 & -10 & 0 \end{bmatrix}$$

and start over. We compute $A_1 = \left[\begin{smallmatrix} -5 & 0 \\ 0 & -5 \end{smallmatrix}\right]$, $R_\xi(s) = \mathrm{diag}\{5 + 2s + 0.2s^2, 5 + s\}$

$$N = \begin{bmatrix} -0.0887 & 0 & -0.0907 \\ 0 & 0 & 0 \\ 0.0867 & 0 & 0.0887 \end{bmatrix} \qquad J = \begin{bmatrix} -11.16 & 0 & -15 & -16.308 & 0 \\ 0 & -50 & 0 & 0 & -10 \end{bmatrix}$$

$$H = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad \overline{K}_J = \begin{bmatrix} 15.858 & 0 & 0 & 0 & 0 \\ 0 & -13.166 & 0 & 0 & 0 \end{bmatrix}.$$

The inverse transfer-matrix of the system after preliminary feedback is

$$H^*(s)^{-1} = \begin{bmatrix} 5{+}2s{+}0.2s^2 & 0 \\ 0 & 5{+}s \end{bmatrix} + \frac{1}{25{+}10\,s{+}s^2} \begin{bmatrix} -33.46\,s - 167.3 & 0 \\ 0 & 25\,s{+}125 \end{bmatrix}.$$

The application of $\overline{K}_J$ on the transformed system sets the sub-matrix A_3 to zero and since sub-matrix $\overline{C}_1 = 0$, the continuous part z_1 is rendered unobservable. The transfer-function $H_b(s)$ of the closed-loop system with preliminary feedback is $H_b^*(s) = \left[\begin{smallmatrix} \frac{1}{5+2s+0.2s^2} & 0 \\ 0 & \frac{1}{5+s} \end{smallmatrix}\right]$ and it is easy to verify that $H_b^*(s)^{-1}$ is just the polynomial part of $H^*(s)^{-1}$ and that $R_\xi(s) = H_b^*(s)^{-1}$. Here the application of either $u = \overline{K}_J z$ or $u = \overline{K}_n z$ to system (4.1) results in an unstable system. Only the application of $u = (\overline{K}_J + \overline{K}_n)z$ on (4.1) results in a system with all poles stable.

5. NONLINEAR EXAMPLE

In this Section we give some numerical results of the simulation of DPC applied to the nonlinear example of Section 4. The complete nonlinear model is given in the Appendix.

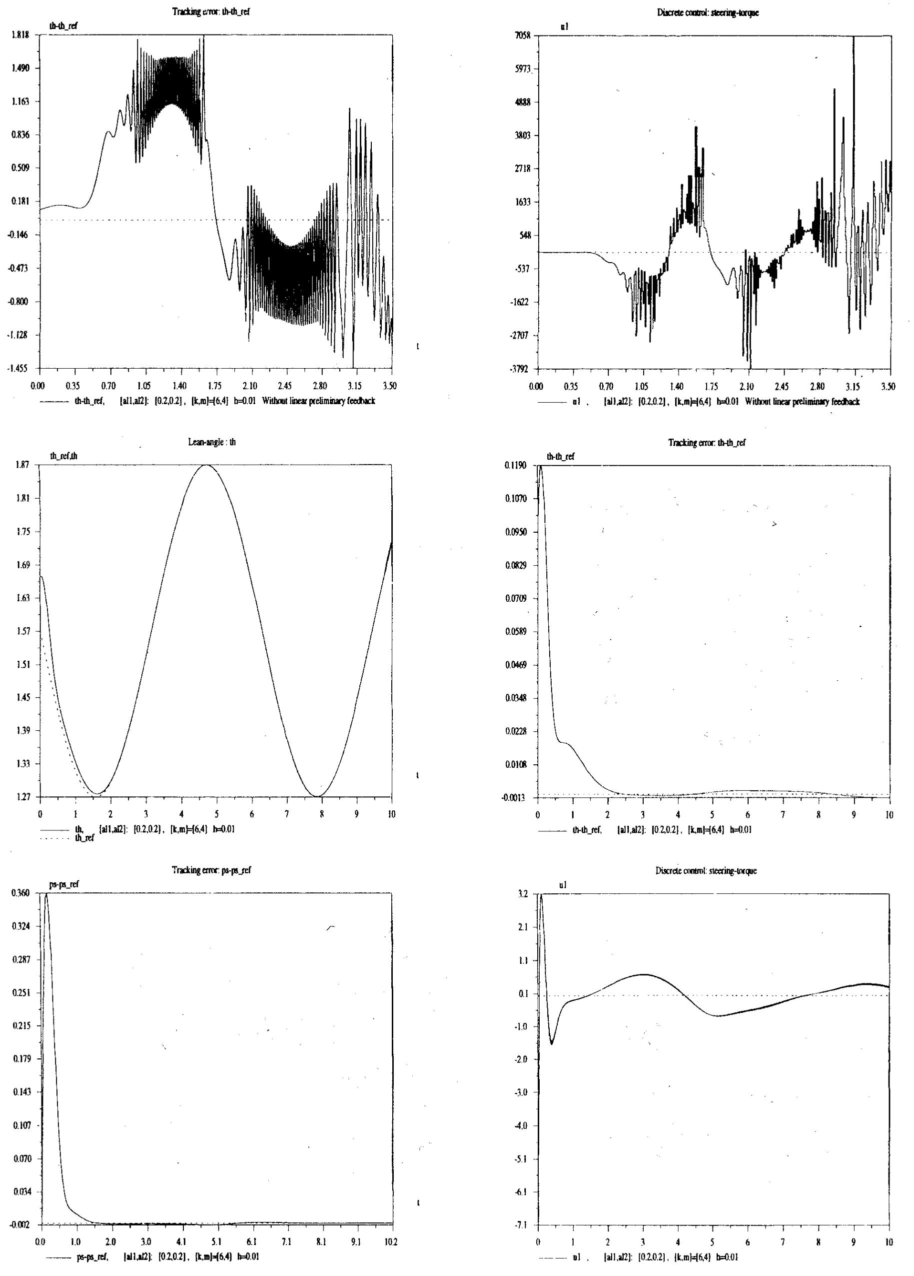

Fig. 5.2. Top: Simulation of the controlled system with discrete controller using DPC without linear preliminary feedback and one step of the Hirschorn algorithm on θ and ϕ. Left: $\theta(-)$, $\theta_{\rm ref}(\cdots)$; right: steering torque u_1.

Middle and bottom: DPC with preliminary feedback; middle left $\theta(-), \theta_{\rm ref}(\cdots)$; middle right: e_θ; bottom left: e_ψ; bottom right: control (steering torque).

We apply nonlinear DPC with the linear preliminary feedback computed in the previous section (for the linear system). The objective is to track the lean angle θ and the roll angle ψ. The control is taken constant on intervals of length h.

The control inputs are two torques: the steering torque normal to the direction of ψ' and normal to ϕ' and the pedalling torque in the direction of ψ'. As reference trajectory we have chosen for $\theta(t)$ to follow a sine-function and $\psi(t)$ is to track the integral of atan(t). Figure 5.2 shows a simulation of the DPC on the top, without, and in the middle and bottom, with, preliminary linear feedback. The first plot of Figure 5.2 shows the lean angle θ as a solid line and the reference function $\theta_{\mathrm{ref}}(t)$ as a dashed line. Next to it we have one of the controls, the steering torque $u_1(t)$. Clearly $e_\theta(t)$ does not converge to zero. Thus DPC without preliminary feedback does not work. The plots in the middle and on the bottom show DPC with preliminary feedback. We start in the first plot with θ as a solid line and θ_{ref} as a dashed line. Next to it we have the error $e_\theta(t) = \theta(t) - \theta_{\mathrm{ref}}(t)$ On the bottom we show on the left side, the tracking error $e_\psi(t) = \psi(t) - \psi_{\mathrm{ref}}(t)$ and on the right side, one of the control inputs, again the steering torque. With preliminary feedback $e_\theta(t)$ and $e_\psi(t)$ converge to zero. For $h > 0.2$ the error $e_\theta(t)$ starts to diverge. We see for this example that DPC with preliminary feedback works when applied on a nonlinear system. If we drop the preliminary feedback, DPC destabilizes the system.

6. CONCLUSION

In this paper, we have presented a predictive type, hybrid open-loop closed-loop strategy based on a control strategy introduced in [10]. In particular, we have shown that the controller in [10] can be applied to a much broader class of systems if it is modified by a preliminary feedback. We have done an analysis in the linear case and shown how such a preliminary feedback can be designed and how it can be applied to nonlinear systems.

We have only considered preliminary feedbacks that are static state feedback. If the state is only partially observable, it should be possible to design adequate dynamic preliminary feedbacks. It may also be interesting to study nonlinear preliminary feedbacks.

The results of this paper can trivially be generalized to the case where system dynamics is perturbed by a known disturbance function.

APPENDIX

We use the following constant parameters to model the wheel: m mass of the wheel ($= 1\,\mathrm{kg}$), r radius of the wheel ($= 1$ m), I_r radial moment of inertia ($= 0.5\,\mathrm{kg\,m^2}$), I_n normal moment of inertia ($= 0.25\,\mathrm{kg\,m^2}$) which give rise to the following expressions: $A_1 = I_n + m\,r^2$, $A_2 = I_r + m\,r^2$, $A_3 = m\,r^2 + I_n - I_r$, $A_4 = 2\,m\,r^2 + I_n$. The nonlinear wheel model is:

$$\begin{bmatrix} I_r\, s\theta^2 + A_1 c\theta^2 & 0 & A_1 c\theta \\ 0 & A_2 & 0 \\ A_1 c\theta & 0 & I_n + m\,r^2 \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} =$$

$$\begin{bmatrix} I_n \dot\theta\, \dot\psi\, s\theta + 2\, A_3 \dot\phi\, \dot\theta\, c\theta s\theta \\ -A_1 \dot\phi\, \dot\psi\, s\theta - A_3 \dot\phi^2 c\theta s\theta - m\, g\, r\, c\theta \\ A_4 \dot\phi\, \dot\theta\, s\theta \end{bmatrix} + \begin{bmatrix} u_1 s\theta \\ 0 \\ u_2 \end{bmatrix},$$

where $s\theta = \sin(\theta)$, $c\theta = \cos(\theta)$ and $s\phi = \sin(\phi)$, $c\phi = \cos(\phi)$. If needed the position of the center of mass of the wheel (x, y) can be obtained by integrating numerically the following equation.

$$\begin{bmatrix} \dot x \\ \dot y \end{bmatrix} = \begin{bmatrix} -r(\dot\phi\, c\theta s\phi - \dot\theta\, s\phi s\theta + \dot\psi\, s\phi) \\ -r(\dot\phi\, c\theta s\phi + \dot\theta\, s\phi s\theta + \dot\psi\, s\phi) \end{bmatrix}.$$

(Received January 31, 1995.)

REFERENCES

[1] K. E. Brenan, S. L. Campbell and L. R. Petzold: Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, Elsevier 1989.

[2] S. L. Campbell and C. W. Gear: The index of general nonlinear DAEs. Numer. Math., to appear.

[3] S. L. Campbell: High index differential algebraic equations. J. Mech. Structures and Machines *23* (1993), 199–222.

[4] S. L. Campbell, R. Nikoukhah and D. von Wissel: Numerically generated path stabilizing controllers I: Theoretical concerns. In: Proc. of ACC, 1994, pp. 1918–1920.

[5] A. Isidori: Nonlinear Control Systems: An Introduction. Springer, Berlin 1989.

[6] R. M. Hirschorn: Invertibility of multivariable nonlinear control systems. IEEE Trans. Automat. Control *AC-24* (1979), 6, 855–865.

[7] R. M. Hirschorn: Output tracking in multivariable nonlinear systems. IEEE Trans. Automat. Control *AC-26* (1981), 2, 593–595.

[8] K. P. Jankowski and H. ElMaraghy: Inverse dynamics and feedforward controllers for constrained flexible joint robots. In: Proc. 31 Conf. Dec. Contr., 1992, pp. 317–322.

[9] K. P. Jankowski and H. Van Brussel: Discrete-time inverse dynamics control of flexible joint robots. J. Dynamic Systems, Measurement and Control *114* (1992), 229–233.

[10] K. P. Jankowski and H. Van Brussel: An approach to discrete inverse dynamics control of flexible-joint robots. IEEE Trans. Robotics Automation *8* (1992), 651–658.

[11] L. M. Silverman: Inversion of Multivariable Linear Systems. IEEE Trans. Automat. Control *AC-14* (1969), 3, 270–276.

[12] L. M. Silverman: Discrete Riccati equations: Alternative algorithms, asymptotic properties, and system theory interpretations. Control and Dynamic Systems, Advances in Theory and Appl. *12* (1976), 313–386.

[13] W. Respondek and H. Nijmeijer: On local right-inveritibility of nonlinear control systems. Control-Theory and Advanced Technology *4* (1988), 3, 325–348.

[14] D. von Wissel and R. Nikoukhah: Hybrid Open-Loop Closed-Loop Path-following Control with Preliminary Feedback. Research Report No. 2173, INRIA, January, 1994.

[15] D. von Wissel, R. Nikoukhah and S. L. Campbell: On a new predictive control strategy: Application to a flexible-joint robot. In: CDC, Florida 1994, pp. 3025–3026.

[16] M. Wonham: Linear Multivariable Control. Springer-Verlag, New York 1972.

D. von Wissel, Dr. R. Nikoukhah and Dr. F. Delebecque, INRIA, Rocquencourt BP 105, 78513 Le Chesnay Cedex. France.

Dr. S. L. Campbell, Department of Mathematics, North Carolina State University, Raleigh, NC 27695-8205. U. S. A.