Proceedings of the American Control Conference
Albuquerque, New Mexico June 1997
0-7803-3832-4/97/$10.00 © 1997 AACC

MPC Extensions to Feedback Linearizable Systems

James A. Primbs*
Control and Dynamical Systems
California Institute of Technology
Pasadena, CA 91125
jprimbs@cds.caltech.edu

Vesna Nevistić
Automatic Control Laboratory
Swiss Federal Institute of Technology (ETH)
CH-8092 Zürich, Switzerland
vesna@aut.ee.ethz.ch

Abstract

We present and compare various extensions of Model Predictive Control (MPC) to the class of both full state and input/output feedback linearizable systems. The presented approaches are valid even for systems with unstable zero dynamics, with the goal being a reduction of the burdensome computational costs typically associated with nonlinear MPC. Finally, these various approaches are tested on examples.

1. Introduction

Determination of the optimal feedback law for nonlinear optimal control problems requires the solution of the Hamilton-Jacobi–Bellman (HJB) partial differential equation. Difficulties in solving the HJB equation for high dimensional systems have precluded its use except in specific areas, and have motivated the study of alternative control techniques. One such technique is model predictive control (MPC), also known as moving or receding horizon control, which uses on–line optimization and a receding horizon implementation [2]. In addition, MPC naturally and explicitly handles both multivariable input and output constraints, including hard constraints or saturation of the control input.

Because of the on-line implementation of MPC, computational issues become relevant to a discussion of its applicability. For linear systems with quadratic cost and linear constraints, MPC reduces to a quadratic program which can be efficiently solved. For nonlinear systems, MPC generally results in computationally demanding nonconvex optimization problems [4]. In these instances, the efficiency of the solution of these nonconvex optimization problems can influence the efficacy of MPC based control.

One approach developed to minimize the computational burden of MPC for nonlinear systems is that of feedback linearizing the system, when possible, and applying MPC in the new linearized coordinates [5]. While this transformation will produce linearized prediction dynamics, which greatly increase the computational efficiency of the prediction phase, it also effects the cost, which can be changed from quadratic to nonlinear and nonconvex. Furthermore, it raises a number of new possibilities and questions concerning the implementation of MPC on feedback linearizable systems.

In this paper we consider various techniques to reduce the computational burden associated with applying full nonlinear MPC to both full state and input/output feedback linearizable systems. In both cases, the approach is based on feedback linearizing the nonlinear system to obtain either a full or partial linear prediction model.

*Supported by NSF

2. Problem formulation

We will consider the following single-input single-output nonlinear system:

$$\begin{cases} \dot{x}(t) = f(x(t)) + g(x(t))u(t), \ f(0) = 0 \\ y = h(x(t)) \end{cases} \tag{1}$$

with $x \in \mathcal{R}^n$ and performance objective:

$$J(u) = \int_0^\infty (y^2(t) + ru^2(t)) \, dt. \tag{2}$$

The goal of such a problem is to find a state-feedback control law $u^* = \Phi(x)$ such that the performance objective (2) is minimized, subject to the nonlinear system dynamics (1). Model predictive control represents one approach to the optimal control problem.

3. Model Predictive Control

An MPC algorithm is conventionally formulated in discrete time by solving an on–line open loop finite horizon optimal control problem at each sampling time k, respecting the following objective function:

$$\min_{\mathcal{U}_k} \left[\sum_{i=1}^{P} h^2(x_{k+i}) + \sum_{i=0}^{M-1} r u_{k+i}^T u_{k+i} \right] \tag{3}$$

subject to:

$$c(x,u) \leq 0 \tag{4}$$

$$\dot{x} - f(x) - g(x)u = 0 \tag{5}$$

Here:

x_{k+i} : predicted state vector at time $k+i$ based on the states x_k at time k, obtained by using prediction model (5)

$\mathcal{U}_k$: control sequence u_{k+i}, $i = 0, ..., M-1$ computed by the optimization algorithm at time k; $u_{k+i} = 0$ for $i \geq M$; u_k is the control move to be implemented at time k.

M, P : *input* (*control*) and *output* (*prediction*) horizon, respectively; $M \leq P$.

Since constraints $(c(x, u))$ can be directly included in the optimization, MPC is considered as the only methodology which deals with both input and output constraints explicitly. Although the optimization problem solved at time k results in an optimal sequence of M present and future control moves, only the first control move, u_k, is implemented on the real plant over the time $[k, k+1]$. At time step $k+1$, the horizons M and P are shifted ahead by one step, and a new optimization problem with new initial condition x_{k+1}, is solved. This kind of implementation is known as the *receding* or *moving horizon* approach.

Determining an optimal open-loop control for a given initial state is a relatively simple task, and makes the MPC algorithms attractive in many situations. By repeatedly solving an on–line open-loop optimization for the current state, and applying the minimizing control for a short time before repeating the procedure, MPC avoids solving the HJB equation.

Various MPC–based methods have been employed to solve the optimal control problem for nonlinear systems. Basically, three approaches exist:

1. Nonlinear MPC (NLMPC)
2. MPC in combination with feedback linearization (MPC+FL).
3. Approximate MPC techniques (gain-scheduled linear MPC).

In this paper, we focus on NLMPC and MPC+FL, which will be applied to the examples in the next section.

3.1. NLMPC Technique

The standard nonlinear MPC (NLMPC) technique and its modifications use nonlinear models for prediction according to (3)–(5), which generally result in non–convex nonlinear programs (NLP), even if the cost function and constraint sets are convex. Therefore, finding a global optimum is a very difficult and computationally very demanding task. The most time consuming and computationally burdensome portion involves the repeated simulation of the prediction model (5) while the optimization scheme searches for a minimum of the cost (3).

3.2. MPC+FL Technique

The MPC+FL technique (Fig.1) attempts to gain computational efficiency by feedback linearizing the plant and restating the MPC problem in the new linearized coordinates [5]. This reduces the prediction phase of MPC to the simulation of a linear system for which efficient methods exist. At the same time, the transformation to linear coordinates may have converted a quadratic cost to a nonconvex cost, and linear or convex constraints to nonconvex constraints. This new optimization problem in the linearized coordinates is not necessarily easier to solve.

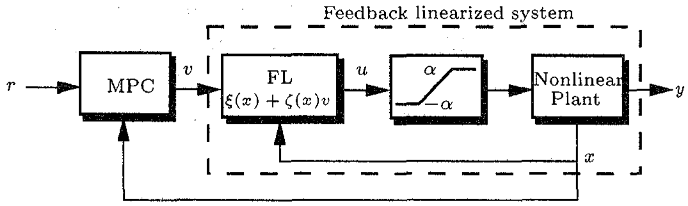

Figure 1: MPC+FL control structure

Below, we give more detailed descriptions of various options for the MPC+FL approach:

Full state feedback linearizable systems

If the system (1) has relative degree n where n is dimension of the state (cf. [3]), the transformation

$$\begin{aligned} \phi_1(x) &= h(x) \\ \phi_2(x) &= L_f h(x) \\ &\dots \\ \phi_n(x) &= L_f^{n-1} h(x) \end{aligned} \tag{6}$$

where $L_f h(x)$ denotes the Lie derivative of $h(x)$ with respect to $f(x)$, will bring the system to the following normal form where in the new coordinates $z_i = \phi_i(x)$, $1 \leq i \leq n$, the dynamics become.

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\dots \\ \dot{z}_{n-1} &= z_n \\ \dot{z}_n &= b(z) + a(z)u \end{aligned} \tag{7}$$

with $y = z_1$. Making the transformation $u = -\frac{b(z)-v}{a(z)}$, completes the transformation to linear dynamics with input v. The original cost in these new coordinates is given by,

$$J(v) = \int_0^\infty (z_1^2 + ru^2(t))\, dt \tag{8}$$

$$= \int_0^\infty (z_1^2 + r\left(\frac{b(z)-v}{a(z)}\right)^2)\, dt \tag{9}$$

which is in general nonlinear and nonconvex. We are now faced with some choices for our MPC+FL implementation. We list the possible approaches below:

- **Quadratic Cost:** Perhaps the simplest method of dealing with the introduction of the nonconvexity in the cost is to abandon it altogether in favor of a new cost in terms of the new linearized variables
$$J(v) = \int_0^\infty (z^T Q z + R v^2(t))\, dt \qquad (10)$$
where Q and R may be chosen in such a way to approximate the true nonlinear cost as closely as possible. In this case the nonlinear problem has been completely converted into a linear MPC problem for which efficient methods exist. In some cases, this lack of respect for the original cost can result in arbitrarily poor performance [6].

- **Original Cost:** A second alternative is to use the original nonlinear cost
$$J(v) = \int_0^\infty (z_1^2 + r\left(\frac{b(z) - v}{a(z)}\right)^2)\, dt$$
with the linear dynamics given in (7). Since the major computational burden associated with nonlinear MPC comes from the simulation of trajectories associated with the prediction model, which has been converted to a linear system, we still expect a dramatic improvement in the efficiency of this formulation over that of an NLMPC formulation.

It should be clear that constraints in the original coordinates would be transformed accordingly, and may change from linear and convex to nonlinear and nonconvex, or vice–versa.

Input/output feedback linearizable systems

For systems (1) with relative degree $r < n$, the system can be transformed to the following form.

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\cdots \\ \dot{z}_{r-1} &= z_r \\ \dot{z}_r &= b(\xi,\eta) + a(\xi,\eta)u \\ \dot{\eta} &= q(\xi,\eta) \end{aligned} \qquad (11)$$

where $\xi = (z_1, ..., z_r)$ and $\eta = (z_{r+1}, ..., z_n)$. Defining $u = -\frac{b(\xi,\eta)-v}{a(\xi,\eta)}$ converts the first r coordinates of (11) to a linearized system with input v. For purposes of the prediction phase in MPC, this linear portion has been decoupled from the nonlinear portion and can be simulated autonomously.

The remaining $n - r$ nonlinear states, η, are known as the zero dynamics of the system [3] (i.e. the dynamics associated with the system when the output is identically zero). It is clear that these dynamics will not be taken into account by any cost in terms of only y and v.

These considerations lead again to a number of alternative MPC formulations:

- **Quadratic Cost:** Once again consider using a quadratic cost in the new linearized coordinates:
$$J(v) = \int_0^\infty (z^T Q z + R v^2(t))\, dt \qquad (12)$$
where Q and R are design parameters as before. If the zero dynamics are stable and do not appear in constraints, there may be no need to simulate the zero dynamics since they do not appear in the cost. Hence the problem is reduced to one with linear prediction dynamics and a quadratic cost.

  If the zero dynamics are unstable, or appear in constraints, it becomes necessary to simulate this portion of the system. In this case, the advantage over NLMPC lies in the fact that the dimension of the nonlinearity that must be simulated has been reduced from the full system of dimension n to only the dimension of the zero dynamics, $n-r$. This reduction in the dimension of the nonlinearity should reduce the computational costs associated with the demanding prediction phase of the MPC algorithm.

- **Original Cost:** A final approach is to use the original cost
$$J(v) = \int_0^\infty (z_1^2 + r\left(\frac{b(\xi,\eta) - v}{a(\xi,\eta)}\right)^2)\, dt$$
and again rely on the fact that the dimension of the system nonlinearity has been reduced, which should result in improved computational efficiency over NLMPC. Note that the zero dynamics, η, appear in this cost and hence must be simulated.

4. Simulation Results

In this section we demonstrate and compare the different approaches to MPC+FL by applying them to examples. Our MPC software uses a DASSL integration routine for nonlinear systems, and NPSOL for nonlinear optimizations. For linear systems with quadratic costs, standard QP software is used.

4.1. Full state FL example

Consider the following nonlinear dynamics:

$$\begin{aligned} \dot{x_1} &= x_2 - (1 + \sin^2(x_1 + x_3))u \\ \dot{x_2} &= x_3 - \sin(x_3)\cos(x_3) + \cos(x_3)(1 + \sin^2(x_1 + x_3))u \\ \dot{x_3} &= -\sin(x_3) + (1 + \sin^2(x_1 + x_3))u \\ y &= x_1 + x_3 \end{aligned}$$

with cost
$$\int_0^\infty (y^2 + (0.1)u^2)dt$$

This system is full state feedback linearizable under the transformation:

$$z_1 = x_1 + x_3$$

Table 1

Controller	sec/iter.	Cost
NLMPC	11.8	4.42
MPC+FL(orig)	0.57	4.40
MPC+FL(quad)	0.02	4.53

Table 1: Comparison of computational costs for different techniques: $M = 5$, $P = 10$, initial condition [0.5 0 0.5]

$$\begin{aligned} z_2 &= x_2 - \sin(x_3) \\ z_3 &= x_3 \\ v &= -\sin(z_3) + (1 + \sin^2(z_1))u \end{aligned}$$

This reduces the MPC problem to the following linear prediction model

$$\dot{z_1} = z_2, \quad \dot{z_2} = z_3, \quad \dot{z_3} = v$$

with transformed cost:

$$\int_0^\infty (z_1^2 + (0.1)\left(\frac{(v + \sin(z_3))}{(1 + \sin^2(z_1))}\right)^2)dt \qquad (13)$$

which is equivalent to the full NLMPC problem in original coordinates.

Additionally, we will consider using the following quadratic cost in the new coordinates which represents a rough quadratic approximation to the true cost:

$$\int_0^\infty (z_1^2 + (0.1)z_3^2 + (0.1)v^2)dt \qquad (14)$$

An MPC algorithm using horizon lengths $M = 5$ and $P = 10$ and sample time $0.2s$ was applied in each of the three situations given above to the initial condition [0.5, 0, 0.5]. Table 1 presents the average computation time per MPC optimization iteration (sec/iter.) and the cost achieved (Cost) by nonlinear MPC (NLMPC) as well as MPC+FL using the original transformed cost (MPC+FL(orig)) (eqn (13)) and a quadratic approximation to the cost (MPC+FL(quad)) (eqn (14)). The results show a dramatic savings in time due to the MPC+FL techniques. Just by converting to a linear prediction model which avoids the costly simulations of a nonlinear prediction model, the time per iteration was reduced by a factor of approximately 20 from NLMPC to MPC+FL(orig). Furthermore, by abandoning the original cost and using the quadratic approximation given in (14), the nonconvex optimization is reduced to a QP. This reduced the average time per iteration by another factor of more than 20.

Trajectories from NLMPC and MPC+FL(quad) are shown in Figure 2 (MPC+FL(orig) is omitted since it is equivalent to NLMPC). Despite using only an approximation to the true cost, MPC+FL(quad) results in a cost (Table 1) and trajectories similar to those produced by NLMPC. In fact, MPC+FL(quad) was found to provide a reasonable approximation to NLMPC for a number of other initial conditions as well.

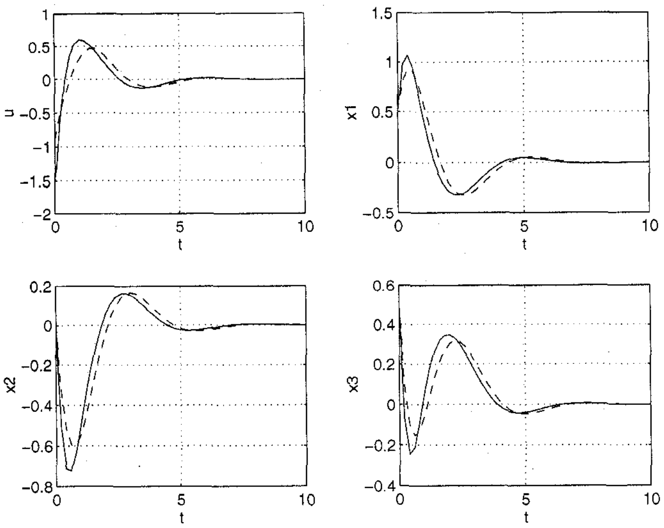

Figure 2: NLMPC (solid), MPC+FL(quad.) (dashed), Initial condition [0.5 0 0.5]

In the next example we consider a system that is only input/output feedback linearizable and cannot be completely converted to a linear system.

4.2. I/O FL example

Consider the following system with relative degree 2:

$$\begin{aligned} \dot{x_1} &= \frac{x_2}{2 + \sin(x_3)} \\ \dot{x_2} &= \frac{-x_2\cos(x_3)(x_1 + 1)}{2 + \sin(x_3)} + (2 + \sin(x_3))u \\ \dot{x_3} &= -x_1 + \sin(x_3) \\ y &= x_1 \end{aligned}$$

subject to the constraint $|x_3| \leq 2$. Under the following transformation:

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= \frac{x_2}{2 + \sin(x_3)} \\ z_3 &= x_3 \\ v &= -z_2\cos(z_3) + u \end{aligned}$$

it becomes:

$$\dot{z_1} = z_2, \quad \dot{z_2} = v, \quad \dot{z_3} = -z_1 + \sin(z_3)$$

which has unstable zero dynamics contained in z_3. In the new coordinates the constraint is $|z_3| \leq 2$.

For these dynamics we consider the following MPC cost:

$$\int_0^\infty (y^2 + (0.1)u^2)dt$$

Table 2

Controller	sec/iter.	Cost
NLMPC	165	33.06
MPC+FL(orig.)	16.1	33.18
MPC+FL(quad.)	13.9	35.35

Table 2: Comparsion of computational costs for different techniques: $M = 2$, $P = 5$, initial condition [0.5 0 0]

or in the feedback linearized coordinates:

$$\int_0^\infty (z_1^2 + (0.1)(v + z_2\cos(z_3))^2)dt$$

Again, we will also compare this with a rough quadratic approximation to the above cost:

$$\int_0^\infty (z_1^2 + (0.1)z_2^2 + (0.1)v^2)dt$$

Note that this cost does not "see" the zero dynamics at all, whereas the original cost does.

These approaches are compared in Table 2 and Figure 3 using an MPC algorithm with time step $0.2s$ and a control and prediction horizon of $M = 2$ and $P = 5$, respectively. This time, the effect of feedback linearization is to reduce the dimension of the nonlinearity from 3 to 1. In both MPC+FL schemes, the unstable zero dynamics must be simulated in order to satisfy the constraint. Nevertheless, the linear portion is decoupled from the nonlinear zero dynamics and can be simulated autonomously and much more efficiently. This reduces the time per MPC iteration by a factor of 10 for the initial condition [0.5, 0, 0] simulated in Figure 3.

The use of a quadratic cost does not result in that significant of a reduction in time per iteration, indicating that the majority of time is spent in the simulation of the nonlinear zero dynamics and the replacement of a non-convex cost with that of a quadratic cost does not provide great savings in this case.

Once again, the quadratic cost is reasonable in comparison with the cost of the full NLMPC problem, but in both cases, the MPC controllers are not able to bring the output to zero.

5. Concluding Remarks

In general, our simulations lead to the following observations. Due to the large computational differences between simulating nonlinear versus linear systems, it was always advantageous to feedback linearize before performing MPC. Any reduction in the dimension of the nonlinearity resulted in large computational savings due to faster simulation of the system. The use of an approximate quadratic cost was justified when the MPC problem could be reduced to a QP. In other cases, the use of a quadratic cost only mildly increased the speed of the MPC algorithms, perhaps not enough to justify the use of only an approximate cost. In general, combining feedback linearization with MPC and utilizing the structure present in many nonlinear systems appears to be a promising approach to avoiding some of the computational problems associated with MPC.

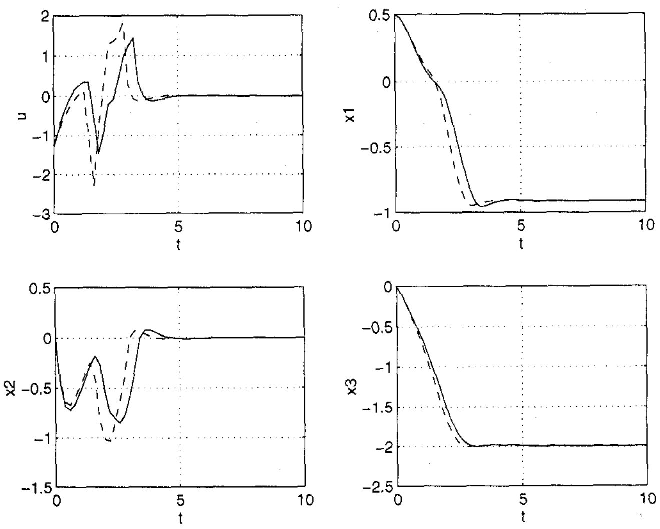

Figure 3: NLMPC (solid), MPC+FL(quad) (dashed), Initial condition [-0.5 0 0.5]

References

[1] A. E. Bryson and Y. Ho. *Applied Optimal Control.* Hemisphere Publ. Corp., Washington D.C., 1975.

[2] C. E. García, D. M. Prett, and M. Morari. Model predictive control: Theory and practice – A survey. *Automatica*, 25(3):335–348, May 1989.

[3] A. Isidori. *Nonlinear Control Systems.* Springer-Verlag, third edition, 1995.

[4] D.Q. Mayne and H. Michalska. An implementable receding horizon controller for the stabilization of nonlinear systems. In *Conf. on Decision and Control*, pages 3396–3397, Honolulu, HI, 1990.

[5] V. Nevistić and M. Morari. Constrained control of feedback-linearizable systems. In *Proceedings of the European Control Conference*, pages 1726–1731, Rome, Italy, 1995.

[6] V. Nevistić and J. A. Primbs. Constrained control of nonlinear systems: A converse HJB approach. CDS Technical Memo CIT-CDS 96-021, California Institute of Technology, Pasadena, CA 91125, Dec 1996.