

This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



For the full text of this licence, please go to:  
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

Model Predictive Control of Nonlinear Systems:  
Computational Burden and Stability

Wen-Hua Chen and Donald J. Ballance  
Centre for Systems & Control  
and Department of Mechanical Engineering

John O'Reilly  
Centre for Systems & Control  
and Department of Electronics and Electrical Engineering  
University of Glasgow  
Glasgow G12 8QQ, UK  
email: {wchen and D.Ballance}@mech.gla.ac.uk  
J.OReilly@elec.gla.ac.uk  
Tel: 0044 141 330 5490  
Fax: 0044 141 330 4343

January 22, 2003

## Abstract

Implementation of Model Predictive Control (MPC) for nonlinear systems requires on-line solution of a non-convex, constrained nonlinear optimisation problem. Computational delay and loss of optimality arise in the optimisation procedures. This paper presents a practical MPC scheme for nonlinear systems with guaranteed asymptotic stability. It is shown that when an initial control profile is chosen to satisfy an inequality condition in each on-line optimisation procedure, the nonlinear system under the proposed nonlinear MPC is asymptotically stable. The stability condition presented in this paper enables the “fictitious” terminal control to be nonlinear, rather than only linear, and thus the stability region is greatly enlarged. Furthermore it is pointed out that nominal stability is still guaranteed even though the global, or even the local, minimisation of the objective cost is not achieved within the prescribed computational time.

**Keywords:** Nonlinear systems, generalised predictive control, computational delay, optimisation, stability, constrained control  
online optimisation problem:

$$\min_{U(t)} \frac{1}{2} \int_0^T x(t, \tau)^T Q x(t + \tau) + u(t, \tau)^T R u(t, \tau) dt + x(t, T)^T P x(t, T) \quad (1)$$

subject to system dynamics,

$$\Gamma U(t) \geq \gamma \quad (2)$$

and

$$2(Ax(t, \tau) + Bu(t, \tau))^T P x(t, \tau) + x(t, \tau)^T Q x(t, \tau) + u(t, \tau)^T R u(t, \tau) \leq 0 \quad (3)$$

for  $\tau \in [0, \delta]$  where  $\delta$  denotes the sampling time interval,  $R > 0$ ,  $Q > 0$  or  $Q \geq 0$  and non-zero  $x$  is detectable in the performance index and  $P > 0$

Fact 1: The constraint (3) is convex if the matrix  $P$  satisfies (4).

$$x_U(\tau)^T (A^T + PA + Q)x_U(\tau) + U^*(\tau)^T B^T P x_U(\tau) + x_U(\tau)^T P B U^*(\tau) + U^*(\tau)^T R U^*(\tau) \geq 0 \quad (4)$$

The constraint (3) can be written as

$$\begin{bmatrix} x(t, \tau)^T & u(t, \tau)^T \end{bmatrix} \begin{bmatrix} A^T P + PA + Q & PB \\ B^T P & R \end{bmatrix} \begin{bmatrix} x(t, \tau) \\ u(t, \tau) \end{bmatrix} \leq 0 \quad (5)$$

Since the state and the control can be expressed as

$$x(t, \tau) = x_x(\tau)x(t) + x_U(\tau)U(t) \quad (6)$$

and

$$u(t, \tau) = U^*(\tau)U(t), \quad (7)$$

respectively, where  $x_x(\tau)$ ,  $x_U(\tau)$  are defined in (??) and (??), it follows that

$$\begin{bmatrix} x(t, \tau) \\ u(t, \tau) \end{bmatrix} = \begin{bmatrix} x_x(\tau) & x_U(\tau) \\ 0 & U^*(\tau) \end{bmatrix} \begin{bmatrix} x(t) \\ U(t) \end{bmatrix} \quad (8)$$

Substituting (8) into (5) gives

$$\begin{bmatrix} x(t)^T & U(t)^T \end{bmatrix} \begin{bmatrix} x_x(\tau) & x_U(\tau) \\ 0 & U^*(\tau) \end{bmatrix}^T \begin{bmatrix} A^T P + PA + Q & PB \\ B^T P & R \end{bmatrix} \begin{bmatrix} x_x(\tau) & x_U(\tau) \\ 0 & U^*(\tau) \end{bmatrix} \begin{bmatrix} x(t) \\ U(t) \end{bmatrix} \leq 0 \quad (9)$$

The above constraint is convex in terms of the variable  $U(t)$  if

$$\begin{bmatrix} x_U(\tau)^T & U^*(\tau)^T \end{bmatrix} \begin{bmatrix} A^T P + PA + Q & PB \\ B^T P & R \end{bmatrix} \begin{bmatrix} x_U(\tau) \\ U^*(\tau) \end{bmatrix} \geq 0, \quad (10)$$

which is equivalent to condition (4).

Therefore it can be shown that the constraint (9), i.e. (3), is convex in terms of the variables  $U(t)$  if condition (4) is satisfied.

Remark: In the online optimisation the constraint (3) is replaced by (9) where all the matrices can be calculated off-line. Since the original optimisation problem in PPP is a QP, together with the convex constraint (9), the new reformed optimisation problem in the PPP algorithm is convex and the global minimum can be found.

## 1 Introduction

Model Predictive Control (MPC) is now a recognised method in control engineering (Clarke, 1994), and much effort has been made to extend MPC from linear systems to nonlinear systems; for example, see Allgöwer and Zheng (1998). There are, however, two main obstacles to the extension of MPC from linear systems to nonlinear systems (Biegler and Rawlings, 1991; Mayne, 1996; Chen and Allgöwer, 1998; Qin and Badgwell, 1998). One obstacle is stability, the other is the on-line computational burden.

Stability is one of the main problems in MPC since early MPC for linear systems was criticised for its loss of stability (Bitmead *et al.*, 1990). This problem has now been solved for linear systems in several ways: infinite horizon predictive control (Rossiter *et al.*, 1996), terminal constraints (Mosca and Zhang, 1992; Demircioglu and Clarke, 1992) and Fake Algebraic Riccati Equation (Bitmead *et al.*, 1990). For nonlinear MPC, the first interesting stability result was given by Mayne and Michalska (1990). They show that, under certain assumptions, predictive control can stabilise plants when a terminal equality constraint is embedded in the optimisation problem; that is, it is required that the terminal state is at a point in the state space. Similar results also have been reported by Genceli and Nikolaou (1993) and Rawlings and Muske (1993). However, from the computational point of view, solving a nonlinear dynamic optimisation problem with equality constraints is highly computationally intensive and in many cases is impossible to perform within a limited time. Furthermore, the feasible region of the proposed MPC is very small. To avoid this, Michalska and Mayne (1993) replaced terminal equality constraints with terminal inequality constraints; that is, the allowable terminal state is a region rather than a point in the state space. To guarantee stability, a dual mode control scheme is proposed. The receding horizon control is employed outside the terminal region and once the state arrives on the boundary of the terminal region, a local linear state feedback controller is employed to steer the state to an equilibrium.

Along this line, a most interesting result was recently presented by (Chen and

Allgöwer, 1998). For a wide class of nonlinear systems, a MPC scheme with a virtual *linear* stabilising control, rather than a dual mode control, is developed. Stability is achieved by involving a terminal cost which bounds the infinite horizon cost of the nonlinear system starting from the terminal region and controlled by the local linear state feedback controller. However, the local state feedback control is never implemented. This so-called quasi-infinite MPC algorithm solves on-line the nonlinear optimisation problem with a performance index containing the terminal cost subject to system dynamics, input constraints and the terminal inequality constraints at each sampling time. Then the receding horizon control is employed until the system arrives the equilibrium.

Apart from the above work on the stability of nonlinear MPC, the stability of several existing nonlinear predictive control schemes has been examined by Magni and Sepulchre (1997) using an inverse optimality approach. In particular, it is shown that predictive control possesses the stability margin of optimal control and it is robust in some sense. A more rigorous result for MPC has been developed by Gyurkovics (1998). The receding horizon control is considered as a Bolza-type optimisation problem. Using the optimality of the value function, the stability condition is established.

Although rigorous stability results for nonlinear MPC have been established (Gyurkovics, 1998; Magni and Sepulchre, 1997), they are not applicable in practical implementations. A constrained nonlinear optimisation problem has to be solved on-line and the heavy on-line computational burden leads to two major problems in the implementation of nonlinear MPC. One issue is the computational delay which cannot be ignored and the other issue is that the global optimisation solution cannot be guaranteed in each optimisation cycle since, in general, it is a non-convex, constrained nonlinear optimisation problem. Neither of these issues have been considered by Gyurkovics (1998) or Magni and Sepulchre (1997). In view of the difficulty of guaranteeing the global minimisation of an objective cost, this requirement has been relaxed for discrete-time systems in Scokaert *et al.* (1999) and for continuous-time systems in Chen and Allgöwer (1998) where it is pointed out that feasibility implies stability for a particular form of MPC (with

linear terminal state feedback control).

The effects of computational delay in implementation of a controller have been investigated by many authors (Pierre and Pierre, 1995; Rattan, 1989). In model predictive control, several schemes taking into account computational delay have recently been proposed for linear systems (Gomma and Owens, 1998; Von Wissel *et al.*, 1997; Casavola and Mosca, 1998). Gomma and Owens (1998) gave a delayed generalised predictive control scheme where control only depends on past measurements rather than current and past measurement. This strategy was compared with Clarke *et al.* (1987) predictive control (Gomma and Owens, 1998). Von Wissel *et al.* (1997) investigated the effect of computational delay in descriptor predictive control. It is shown that the closed-loop system remains stable as long as the delay is smaller than a specified threshold. Design of reference governors in predictive control with computation delay was addressed in Casavola and Mosca (1998). Although the computational burden in nonlinear MPC is much heavier than the linear one, only the algorithm proposed by Ronco *et al.* (1999), Open-loop Intermittent Feedback Control, considers the computation delay in implementation of MPC for nonlinear systems. Similar to other schemes considering the computational delay the last control profile found by minimisation of the objective cost is used until the new optimisation process is finished and a new control profile is available. It has been shown that this scheme is promising by application to several plants. In addition to computational delay, sampling delay is considered in Chen and Allgöwer (1998). Sampling delay is different from the computational delay in that control depends on current measurement.

Recently Chen *et al.* (1999c) proposed a new nonlinear model predictive control algorithm which can remove the computational burden and stability issues. The proposed nonlinear predictive controller is given in a closed-form and thus no on-line optimisation is required. The stability is also guaranteed. It is also pointed out that the well-known computed torque method or dynamic inversion control is a special case of this MPC algorithm (Chen *et al.*, 1999b). However, this method is only applicable for nonlinear systems with stable zero dynamics and a well-defined relative degree.

This paper develops a practical MPC with guaranteed stability for general nonlinear systems. Several issues in the implementation of nonlinear MPC, including computational delay, loss of optimality in the optimisation procedure and stability, are addressed. A new stability condition is presented. The performance index in MPC is modified to take computational delay into account. At each time interval, the calculation of the control profile is divided into a two stage optimisation problem. In the first stage, a “classic” fixed horizon MPC is solved. In the second stage, a terminal control profile within a prescribed time interval is optimised according to the stability requirements. It should be noted that the terminal control is never implemented and it is only used to generate an initial control profile in each optimisation cycle.

## 2 Problem formulation

Consider a nonlinear control system

$$\dot{x}(t) = f(x(t), u(t)), \quad x(t_0) = x_0 \quad (11)$$

where  $x \in R^n$  and  $u \in R^m$  are the state and control vectors respectively. It is supposed that the control input is subject to the constraint

$$u(t) \in U \quad (12)$$

where  $0 \in U \subset R^m$  is a compact and convex set.

In order to distinguish the real variables from the variables in the moving horizon time frame, hatted variables are used in the moving horizon time frame in this paper. In general, a nonlinear MPC problem can be stated as: for any state  $x$  at time  $t$ , find a continuous function  $\hat{u}(\tau; x(t)) : [t, t + T] \rightarrow U$ , in a moving horizon time frame,  $T$ , such that the performance index

$$J_1 = g(\hat{x}(t + T)) + \int_0^T \hat{x}(t + \tau)^T Q \hat{x}(t + \tau + \hat{u}(t + \tau; x(t)))^T R \hat{u}(t + \tau; x(t)) d\tau \quad (13)$$

is minimised where  $Q \geq 0$ ,  $R > 0$  and  $\hat{u}(\cdot; x(t))$  explicitly indicates the control profile  $\hat{u}$  depends on the state measurement  $x(t)$  at time  $t$ . It is required that



$g(x)$  is a continuous, differentiable function of  $x$ ,  $g(0) = 0$  and  $g(x) > 0$  for all  $0 \neq x \in R^n$ . A typical choice of  $g(x)$  is given by

$$g(x) = x(t)^T P x(t) \quad (14)$$

where  $P \in R^{n \times n}$  is a positive definite matrix.

Then the nonlinear MPC law is determined by

$$u(t) = \hat{u}(t; x(t)) \quad (15)$$

Assumptions on the system (12) are imposed:

**A1**  $f : R^n \times R^m \rightarrow R^n$  is twice continuously differentiable and  $f(0, 0) = 0$ .  $0 \in R^n$  is an equilibrium of the system with  $u = 0$ .

**A2** System (12) has an unique solution for any initial condition  $x_0 \in R^n$  and any piece-wise continuous and right-continuous  $u(\cdot) : [0, \infty) \rightarrow U$ ;

**A3** The nonzero state of the system (12) is detectable in the cost. That is,  $Q^{1/2}x \neq 0$  for all nonzero  $x$  such that  $f(x, 0) = 0$  (Sepulchre *et al.*, 1996);

**A4** All states,  $x(t)$ , are available.

To implement the MPC, a constrained nonlinear optimisation problem must be solved on-line. On-line optimisation is a major issue in nonlinear MPC implementation (Allgöwer and Zheng, 1998; Mayne, 1996; Biegler and Rawlings, 1991). Due to the large computational burden, the computational delay time in MPC is significant and cannot be ignored. The optimisation process for MPC with computational delay is shown in Figure 1.

When the state measurement  $x(t)$  is available at the sampling time  $t$ , the optimal control profile  $\hat{u}(\tau; x(t))$ ,  $\tau \in [t, t + \delta]$  yielded by minimisation of the cost (14) is not available until time  $t + \delta$  where  $\delta$  is the computational delay. It makes no sense to optimise the control profile  $\hat{u}(\tau; x(t))$ ,  $t \leq \tau \leq t + \delta$  from time  $t$ . Moreover as will be shown later, it also causes mismatching in the MPC implementation and

thus degrades system performance. Hence, in this paper the performance index (14) is modified to

$$J = g(\hat{x}(t+T)) + \int_{\delta}^T \hat{x}(t+\tau)^T Q \hat{x}(t+\tau) + \hat{u}(t+\tau; x(t))^T R \hat{u}(t+\tau; x(t)) d\tau \quad (16)$$

At each measurement time, the optimisation problem (OP) to be solved on-line can be mathematically formulated as

$$\min_{\hat{u}: [t+\delta, t+T] \rightarrow U} J \quad (17)$$

subject to

$$\left\{ \begin{array}{l} \text{terminal state constraint } x(t+T) \in \Omega \\ \text{the system dynamics (12)} \end{array} \right. \quad (18)$$

There are two main advantages in formulating the model predictive control in the form of the above OP (18) and (19). The first advantage is that a lower computational burden exists when the optimisation occurs over the period  $[t+\delta, t+T]$  compared to the period  $[t, t+T]$ . This reduced optimisation period occurs because the result of the optimisation is not available until time  $t+\delta$  and so the optimisation cannot alter the control that is implemented in the period  $[t, t+\delta]$ . The second major advantage is that at time  $t+\delta$  when the new control is implemented, there will be a closer match between the initial state  $x(t+\delta)$  and the predicted state  $\hat{x}(t+\delta)$  than with the originally measured state  $x(t)$ .

It should be noted that, in addition to the terminal weighting in the performance index (17), a constraint on the terminal states is imposed in OP. As will be shown later, this is due to the stability requirement.

In MPC, one problem is to determine the computational delay off-line. This depends on the computational methods used in solving the OP. The computational issue is an active subject in nonlinear MPC and many algorithms have been proposed (Biegler and Rawlings, 1991; Zheng, 1997; Mayne, 1996). Several computational algorithms have been compared and implemented on a continuous stirred tank reactor (Sistu *et al.*, 1993). We will not consider the choice of optimisation routine in this paper. The difficulty in determining the computational delay off-line lies in that the computational times for different optimisation cycles are different even for the same optimisation algorithm. It is well known that

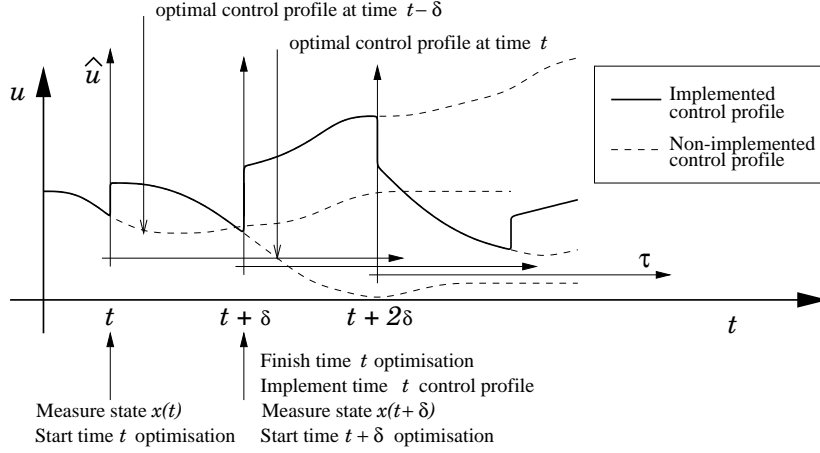


Figure 1: Nonlinear MPC with computational delay

the computational time is quite different in each nonlinear optimisation procedure for different initial points and optimum. This makes it difficult to determine the length of time taken to perform the optimisation off-line. However it will be shown that although the global optimal solution is not achieved and only a local minimum solution exits, or worse the optimisation procedure is not finished within the prescribed time period, stability can still be guaranteed in the MPC algorithm proposed in this paper. This makes it possible to determine the computational delay off-line.

*Remark 1:* Chen and Allgöwer (1998) have considered the sampling time issue in MPC implementation. That is, the control profile is implemented in an open-loop manner until the updating new measurement arrives. This is different from the computational delay considered in this paper. More specifically, the closed-loop control profile in Chen and Allgöwer (1998) is defined as

$$u(\tau) = \hat{u}(\tau; x(t)), \tau \in [t, t + \delta_1] \quad (19)$$

where  $\delta_1$  is the sampling time. This implies that the control depends on the current state. However the MPC algorithm with computational delay can be considered as

$$u(\tau) = \hat{u}(\tau; x(t)), \tau \in [t + \delta, t + 2\delta] \quad (20)$$

This implies that the control depends on the past state measurement. The sampling time is not considered in this paper since it is much smaller than the computational time in Nonlinear MPC. However if the sampling time needs to be considered, it is easy to handle it in our algorithm.

### 3 Terminal Region and MPC Algorithm

For the sake of simplicity, in this paper asymptotic stability of a system is taken to mean that the system has an (asymptotically) stable equilibrium at the origin.

*Definition 1:* The terminal region  $\Omega$  in MPC is defined as a region where, once the state  $\hat{x}(t+T)$ , under the control  $\hat{u}(\tau; x(t))$ ,  $\tau \in [t, t+T]$  yielded by solution of the OP (18–19), arrives, there exists a control  $\hat{u} : [t+T, \infty) \rightarrow U$  which can steer the state to the origin.

*Lemma 1:* Suppose that the system (12) satisfies Assumptions A1–A4, then, if there exists an  $\alpha > 0$  such that for any  $x \in \Omega(\alpha) = \{x \in R^n : g(x) \leq \alpha\}$ , there exists control  $u \in U$  satisfying

$$\frac{\partial g(x)}{\partial x} f(x, u) + x^T Q x(t) + u^T R u \leq 0, \quad (21)$$

then  $\Omega(\alpha)$  is a terminal region for the nonlinear system (12) under the MPC with the performance index (17). The proof of this is given in Appendix A.

Based on Lemma 1, a MPC algorithm with Guaranteed Stability (MPCGS) is proposed as follows.

#### **MPCGS Algorithm:**

1. Measure the state  $x(t_0)$ . No control is imposed during the time interval  $[t_0, t_0 + \delta]$ . In the meanwhile, solving the OP (18) and (19) at  $t = t_0$  and  $x(t) = x(t_0) = x_0$  yields  $\hat{u}(\tau; x(t))$ ,  $\tau \in [t + \delta, t + T]$ .
2. Calculate  $\hat{u}(\tau; x(t))$ ,  $\tau \in [t + T, t + T + \delta)$ , according to condition (22).
3. Implement the optimal open-loop control  $\hat{u}(\tau; x(t))$  for the time interval  $[t + \delta, t + 2\delta]$ . Measure the state at the time  $t + \delta$  and in the meantime, use

the control  $\hat{u}(\cdot; x(t))' : [t + 2\delta, t + T + \delta] \rightarrow U$ , defined as

$$\hat{u}(\tau; x(t + \delta))' = \begin{cases} \hat{u}(\tau; x(t), \tau \in [t + 2\delta, t + T] \\ \hat{u}(\tau; x(t)), \tau \in [t + T, t + T + \delta] \end{cases} \quad (22)$$

as an initial solution of  $\hat{u}(\tau; x(t + \delta))$  to solve the constrained optimisation problem (18) and (19).

4. Let  $t = t + \delta$  and Go to Step 2.

*Remark 2:* When solving the OP, global minimisation or even local minimisation of the objective cost in the above MPCGS algorithm is not required. This is important for engineering implementation. Since this problem is solved on-line, it is difficult to guarantee the global optimal solution, and sometimes even a local minimum cannot be achieved due to the time limitation. One of the key features of the MPCGS algorithm is the way it determines an initial control profile for each optimisation process based on the condition (22), which distinguishes it from other existing MPC algorithms. When the optimisation algorithm starts from this initial control profile, as we will show later, the feasibility and stability of the proposed MPC are guaranteed.

*Remark 3:* In each iteration of the MPCGS, two optimisation problems need to be solved. One is the “classic” MPC optimisation problem. The other is the optimisation problem in Step 2 to determine the terminal stabilising control. For any  $x \in \Omega$ , it is easy to solve this feasibility problem by a nonlinear optimisation procedure. Particularly, for an affine nonlinear system, it will be shown later that the non-linear terminal stabilising control is given in a closed-form and on-line optimisation is not required.

*Remark 4:* How to determine the computational delay off-line is important in the proposed algorithm. It involves a trade-off between the computational burden, robustness and the control performance. If the computation delay is chosen too small, the performance is degraded although the stability can be guaranteed since optimisation cannot be finished at each iteration. If the computation delay is too large, the proposed MPCGS has less robustness against uncertainties

and disturbances. How to analyse the robust stability of the closed-loop system with computational delay is an interesting subject and some results have been presented, for example see Chen *et al.* (1997), Chen and Gu (1991). However it is beyond the scope of this paper.

## 4 Stability and Feasibility

*Definition 2:* The feasibility region refers to a set of initial state points  $x(t_0)$  for which the state trajectory yielded by solving the OP terminates in the region  $\Omega(\alpha)$ .

Lemma 2 guarantees that if the nonlinear MPC problem is feasible at  $t_0$ , it is also feasible for all  $t > t_0$ . This follows from modification of the standard argument; for example, see Genceli and Nikolaou (1993), Michalska and Mayne (1993), Chen and Allgöwer (1998).

*Lemma 2:* Consider the nonlinear system (12) satisfying Assumptions A1–A4 with the input constraint (13). The feasibility of the MPCGS algorithm at time  $t_0$  implies feasibility for all  $t > t_0$ . The proof of this is given in Appendix B.

*Theorem 1:* Suppose that Assumptions A1–A4 are satisfied and the MPCGS algorithm is feasible at time  $t = t_0$ . The MPCGS algorithm for the system (12) is asymptotically stable if there exists control  $u(t)$  such that the following condition is satisfied

$$\frac{\partial g(x)}{\partial x} f(x, u) + x^T Q x + u^T R u \leq 0 \quad (23)$$

for any state  $x$  belonging to the terminal region  $\Omega$ . The proof of this is given in Appendix C.

*Corollary 1:* When the nonlinear system considered is affine, given by

$$\dot{x}(t) = f_1(x) + g_1(x)u \quad (24)$$

condition (22) reduces to that

$$\frac{\partial g(x)}{\partial x} f_1(x) - \frac{\partial g(x)}{\partial x} g_1(x) R^{-1} g_1(x)^T \left( \frac{\partial g(x)}{\partial x} \right)^T + x^T Q x \leq 0. \quad (25)$$

and

$$u = -R^{-1}g_1(x)^T \left( \frac{\partial g(x)}{\partial x} \right)^T \in U \quad (26)$$

The terminal region is given by

$$\Omega(\alpha) = \{x \in R^n : \text{conditions (26-27) hold for all } g(x) \leq \alpha\} \quad (27)$$

If condition (26) is satisfied, a control  $u(t)$  which steers the state to the origin is given by (27). In particular, for a linear system

$$\dot{x} = Ax + Bu, \quad (28)$$

when  $g(x)$  is chosen as (15), condition (26) becomes

$$A^T P + PA - PBR^{-1}B^T P + Q \leq 0. \quad (29)$$

If this condition is satisfied, the terminal region is the whole state space, that is, the MPCGS is global asymptotically stable.

*Remark 5:* Corollary 1 shows that for linear systems, the stability result in this paper reduces to that of the so-called Fake Algebraic Riccati Equation (Bitmead *et al.*, 1990; Poubelle *et al.*, 1988). Hence, the stability condition for linear MPC can be considered as a special case of our stability condition.

*Remark 6:* When the terminal control is restricted to be linear and the terminal weighting item is given by (15), the stability condition for nonlinear MPCGS proposed in this paper reduces to the result in Chen and Allgöwer (1998), Michalska and Mayne (1993). That is, the terminal region is given in (24) by replacing  $u(t)$  with  $u(t) = kx(t)$  where  $k \in R^{m \times n}$  is a constant matrix.

*Remark 7:* Before using the proposed MPCGS, the terminal region should be determined off-line. To determine the maximum attraction region of the proposed MPCGS, it is required to calculate the maximum  $\alpha$  such that (24) holds for any  $x \in \Omega(\alpha)$ . This can be done by making an iteration of the optimisation problem

$$\max_{x \in \Omega(\alpha)} \min_{u \in U} \left\{ \frac{\partial g(x)}{\partial x} f(x, u) + x^T Q x + u^T R u \right\} \quad (30)$$

This is a max/min optimisation problem. The maximum  $\alpha$  is given by increasing  $\alpha$  such that the value in (31) is nonnegative. In general it is not trivial to solve this

max/min problem. A Linear Matrix Inequalities (LMI's) approach is developed to approximately solve this problem by Chen *et al.* (1999a).

However for an affine nonlinear system (25), with the help of the result in Corollary 1, the maximum terminal attraction region can be achieved by solving the following optimisation problem

$$\max \alpha \quad \text{s.t. conditions (26–27) hold for all } g(x) \leq \alpha. \quad (31)$$

This is a standard semi-definite optimisation problem which has been extensively discussed; for example, see Polak *et al.* (1984) for a general setting or see Michalska and Mayne (1993) for MPC.

## 5 A Simulation Example

Consider the van de Vusse reactor kinetic scheme where at an operating point, the process model is given by (Mutha *et al.*, 1997; Doyle *et al.*, 1995):

$$\begin{aligned} \dot{x}_1 &= -50x_1 - 10x_1^2 + (10 - x_1)u \\ \dot{x}_2 &= 50x_1 - 100x_2 - x_2u \\ y &= x_2 \end{aligned} \quad (32)$$

and  $x_1$  and  $x_2$  are the deviation variables for the concentration of components A and B respectively. The control  $u$  is the inlet flow rate.

The control problem for this system is to control the concentration of the component B,  $y$ , by the inlet flow rate,  $u$ . The nonlinear model predictive control proposed in this paper is designed for this plant.

The performance index is chosen as (17) with the weighting matrices

$$P = \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix}; \quad (33)$$

$$Q = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}; \quad (34)$$



and

$$R = 0.1 \quad (35)$$

To determine the applicability and the stability region of the proposed nonlinear MPC, the terminal region needs to be determined.

Since this is an affine nonlinear system, the terminal region is determined by condition (26) in Corollary 1. For the plant (33) with the above weighting matrices in the performance index (17), it can be shown that condition (26) is satisfied for all states in the state space. The closed-loop system under the proposed nonlinear model predictive controller is globally stable.

Following (27), the terminal stabilising nonlinear controller is given by

$$\begin{aligned} u(t) &= -10 \begin{bmatrix} 10 - x_1 & -x_2 \end{bmatrix} Px \\ &= -5(10 - x_1)x_1 + x_2^2 \end{aligned} \quad (36)$$

Now we consider the case where the terminal control is linear (Chen and Allgöwer, 1998; Michalska and Mayne, 1993). First the nonlinear system (33) is linearised around the original point. The linearised system is given by

$$\dot{x} = \begin{bmatrix} -50 & 0 \\ 50 & -100 \end{bmatrix} x + \begin{bmatrix} 10 \\ 0 \end{bmatrix} u \quad (37)$$

Solving the LQ optimal control problem with the weighting matrices  $Q$  and  $R$  in (35–36) yields the feedback gain

$$k = \begin{bmatrix} 0.3148 & 0.3248 \end{bmatrix}; \quad (38)$$

The terminal stabilising control  $u$  is given by

$$u(t) = kx(t), \quad (39)$$

The stability condition is given in (24) by replacing  $u(t)$  with (40). By solving this semi-infinity optimisation problem (Chen and Allgöwer, 1998; Michalska and Mayne, 1993), the terminal region is given by

$$\Omega = \{x \in R^2 | x^T Px \leq 9.63\} \quad (40)$$

Figure 2: Control history

Figure 3: Response

Comparing with the global stability of the nonlinear terminal control law, the above terminal region given by the linear terminal control law (40) is much smaller. Hence, the new stability condition in this paper provides a much less conservative result for nonlinear MPCGS.

The simulation results under the proposed nonlinear MPCGS are shown in Figures 2 and 3. In the simulation, the predictive time is chosen as 0.05 hour. The computational delay is chosen as 0.01 hour (36 seconds). The 4–5th order Runge-Kutta numerical integration method with a fixed step size  $0.002h(7.2s)$  is used for the optimisation. The whole simulation is performed in the Matlab environment with the nonlinear optimisation implemented using the Optimisation Toolbox. The initial state of the plant is measured at time  $t = 0$  and MPCGS starts the optimisation procedure. Before the control profile is yielded by optimisation, no control effort is imposed on the reactor, that is, the control effort is zero. When the control profile is available after the computational delay, the control profile is implemented and new states are measured for starting the new optimisation procedure. It's difficult to determine whether or not the global minimisation of the cost is achieved in each optimisation cycle. Actually in some optimisation cycles, the optimisation process is terminated since the prescribed maximum number of evaluations is arrived at (the maximum number of evaluations in the optimisation is set to be 100 times the number of variables in the simulation). This implies that in some cases a local minimum has not been achieved. The resulting control profile is shown in Figure 2. This may not be as smooth as expected but the stability is guaranteed by the proposed MPCGS algorithm.

## 6 Conclusion

A practical MPC scheme for general nonlinear systems has been presented in this paper. The main advantages and features of the proposed control scheme are that

- the computational delay caused by on-line optimisation is accounted for;
- no global solution to the on-line nonlinear optimisation is required;

- a nonlinear terminal control can be chosen if required and hence the applicability and stability regions of the nonlinear model predictive control is enlarged;
- the feasibility of the proposed scheme implies stability when the proposed initial control profile is used in each optimisation routine.

The proposed nonlinear MPC has been illustrated by a simulation example. Compared to the linear terminal control the stability region is much enlarged due to the new stability condition derived in this paper.

Several obstacles to implementation of MPC of nonlinear systems, including loss of optimality in on-line nonlinear optimisation, computation delay due to extensive computational burden and stability issue, have been removed. As a consequence, the paper takes a step toward the development of practical MPC for general nonlinear systems with unstable zero dynamics or an ill-defined relative degree. Further investigation will focus on robustness of nonlinear MPC and reducing the on-line computational burden using control parametrisation techniques (Gawthrop *et al.*, 1998; Vassiliadis *et al.*, 1994).

## Acknowledgements

This work has been supported by the UK Engineering and Physical Sciences Research Council under grant No. GR/L 62665. The first author would like to thank Professor Peter Gawthrop who introduced him to the subject of predictive control for nonlinear systems. His comments on an earlier draft of this paper are also gratefully acknowledged.

## A Proof of Lemma 1

Suppose that there exists a control  $u(t) \in U$  such that the condition (22) is satisfied for all  $x \in \Omega(\alpha)$ . Then a corresponding  $u(t) \in U$  can be calculated according to the condition (22) for any  $x \in \Omega(\alpha)$ . Let  $u(t)$ ,  $t \in [T + t, \infty)$  be such a control

profile. Let  $V(x) = g(x)$  be a Lyapunov function candidate associated with the state trajectory of the system (12) under this control profile. It is noted that  $g(x)$  is a continuous, differentiable function of  $x$ . It follows from (22) that

$$\dot{V}(x) = \frac{\partial g(x)}{\partial x} f(x, u) \leq -x^T Qx - u^T Ru \leq -x^T Qx \leq 0 \quad (41)$$

The inequality (42) implies that the Lyapunov function  $V(x)$  is non-increasing along the state trajectory. If the state of the system is within the region  $\Omega(\alpha)$ , it remains in this region. The inequality (42) also implies that  $\dot{V}(x) = 0$  only if  $u(t) = 0$  and the state trajectory belongs to the following set

$$N = \{x(t) \in R^n : x(t)^T Qx(t) = 0, t \in R\}. \quad (42)$$

Since  $Q^{1/2}x \neq 0$  for all nonzero  $x$  such that  $f(x, 0) = 0$ , one can prove that the only positive half trajectory contained entirely in  $N$  is  $x(t) \equiv 0$  (Sepulchre *et al.*, 1996; Gyurkovics, 1998). Remember that  $V(0) = g(0) = 0$  and  $V(x) = g(x) > 0$  for all  $x \neq 0$ . This implies that

$$\|x(t)\| \rightarrow 0 \text{ as } t \rightarrow \infty \quad (43)$$

Hence under the control satisfying the condition (22),  $\Omega(\alpha)$  is an attraction region of the system (12) with an equilibrium point  $x = 0$ . According to Definition 1,  $\Omega(\alpha)$  is a terminal region of the MPC. QED.

## B Proof of Lemma 2

When a MPC algorithm is implemented, due to the computational delay and no prior information about the control input, in MPCGS proposed in this paper, no control effort is imposed in the interval  $[t_0, t_0 + \delta)$ . The system (12) becomes

$$\dot{x} = f(x, 0), \quad x(t_0) = x_0 \quad (44)$$

At time  $t_0 + \delta$ , the free system (12) arrives at the state  $x(t_0 + \delta)$ . Since the MPCGS algorithm is feasible at time  $t_0$ , it implies that we can find an (open-loop) control profile  $\hat{u}(\cdot; x_0): [t_0 + \delta, t_0 + T] \rightarrow U$  which can drive the state  $x(t_0 + \delta)$

to arrive in the terminal set  $\Omega(\alpha)$  at the terminal time  $t_0 + T$  by solving the OP (18–19) at time  $t_0$ .

Now we consider the feasibility of the NGPC at time  $t_0 + \delta$ . At the time  $t_0 + \delta$ , the new state measurement  $x(t_0 + \delta)$  is available and the optimisation problem OP with the initial state  $x(t_0 + \delta)$  needs to be solved. In our MPCGS algorithm, this procedure starts from the following initial control profile  $\hat{u}(\cdot, x(t_0 + \delta))' : [t_0 + 2\delta, t_0 + T + \delta] \rightarrow U$ , defined as in (23). The assumption that the MPCGS is feasible at the time  $t_0$  implies that the first part of  $\hat{u}(\cdot, x(t_0 + \delta))'$  can steer the state into the terminal region at the time  $t_0 + T$ . The second part of  $\hat{u}(\cdot, x(t_0 + \delta))'$  is calculated in Step 2 in MPCGS which satisfies the condition (22). According to the MPCGS Algorithm in Section 3,  $\hat{u}(\tau, x_0), \tau \in [t_0 + \delta, t_0 + T]$  is implemented in the time interval  $[t_0 + \delta, t_0 + 2\delta]$ , that is,  $u(\tau) = \hat{u}(\tau; x_0), \tau \in [t_0 + \delta, t_0 + 2\delta]$ . As a result, the system state at the time  $t + 2\delta$  is given by  $x(t_0 + 2\delta) = \hat{x}(t_0 + 2\delta)$  under the assumption that there is no uncertainty and noise. Hence according to Lemma 1, the terminal state  $\hat{x}(t_0 + T + \delta)'$  under the control profile  $\hat{u}'$  remains in the terminal region  $\Omega$ . Clearly, the optimisation problem at  $t_0 + \delta$  is feasible since at least  $\hat{u}(\cdot, x(t_0 + \delta))'$  is a feasible solution. Repeating the above deduction, one can prove that the feasibility of the MPCGS algorithm at the time  $t = t_0$  implies feasibility for all  $t > t_0$ . QED

## C Proof of Theorem 1

First we show that  $x = 0$  is an equilibrium of the closed loop system under the proposed MPCGS.

When  $x(t_0) = 0$  and under no control effort, it follows from  $f(0, 0) = 0$  that

$$x(\tau) = 0 \text{ for } \tau \in [t_0, t_0 + \delta] \quad (45)$$

That is,  $x(t_0 + \delta) = 0$ . It is clear that the optimal control profile which minimise the cost (17) is given by

$$\hat{u}(\tau; x(t_0)) = 0, \tau \in [t_0 + \delta, t_0 + T] \quad (46)$$

and as a result,

$$u(\tau) = 0, \text{ for } \tau \in [t_0 + \delta, t_0 + 2\delta]. \quad (47)$$

and

$$x(\tau) = 0 \text{ for } \tau \in [t_0 + \delta, t_0 + 2\delta]. \quad (48)$$

Repeating the above process, one can show that  $x = 0$  is an equilibrium of the closed-loop system consisting of the nonlinear system (12) and the proposed MPCGS.

For the sake of notational simplicity, in the proof of Theorem 1,  $\hat{u}(\tau; x(t))$  is replaced by  $\hat{u}(\tau)$  when this will not cause confusion. Similarly, the state under the control  $\hat{u}(\tau; x(t))$  in the moving horizon time frame,  $T$ , is denoted by  $\hat{x}(\tau)$ . Clearly, the performance index (17) at time  $t$  depends on the initial  $x(t)$  and the control profile  $\hat{u}(\tau; x(t))^*$  used in the moving horizon time frame, denoted by  $J(x(t), \hat{u}(\cdot; x(t))^*)$  where  $\hat{u}(\tau; x(t))^*$  is any control profile. Let  $V(x(t + \delta))$  denote such a performance cost under the control profile  $\hat{u}(\tau; x(t))$ , that is,

$$\begin{aligned} V(x(t)) &= J(x(t), \hat{u}(\cdot; x(t))) \\ &= g(\hat{x}(t + T)) + \int_{\delta}^T \hat{x}(t + \tau)^T Q \hat{x}(t + \tau) + \hat{u}(t + \tau)^T R \hat{u}(t + \tau) d\tau \end{aligned} \quad (49)$$

Let  $V(x)$  be a candidate Lyapunov function for the closed-loop system. Similar to the proof of  $x = 0$  being an equilibrium of the closed-loop system under the proposed MPCGS scheme in Lemma 2, it is trivial to show that  $V(0) = 0$  since  $g(0) = 0$ .

Now we prove that the Lyapunov function  $V(x)$  along the state trajectory under the proposed MPCGS algorithm is non-increasing. Consider the Lyapunov function value  $V(x(t + \delta))$ . According to MPCGS Algorithm in Section 3, the control profile  $\hat{u}(\tau; x(t + \delta))$ ,  $\tau \in [t + 2\delta, t + T + \delta)$ , is obtained by starting the optimisation procedure from the initial control profile  $u'$  defined in (23).

Define the performance cost in (17) for the system under this control profile



$\hat{u}(\tau; x(t + \delta))'$  in (23) as

$$V(x(t + \delta))' = g(\hat{x}(t + T + \delta)') + \int_{2\delta}^{T+\delta} \hat{x}(t + \tau)'^T Q \hat{x}(t + \tau)' + \hat{u}(t + \tau)'^T R \hat{u}(t + \tau)' d\tau \quad (50)$$

where  $\hat{x}(t + \tau)'$ ,  $\tau \in [2\delta, T + \delta]$  denotes the system's state under the control profile  $\hat{u}(\tau; x(t + \delta))'$  in (23).

The equation (51) can also be written as

$$\begin{aligned} V(x(t + \delta))' &= g(\hat{x}(t + T + \delta)') \\ &+ \int_{2\delta}^T \hat{x}(t + \tau)'^T Q \hat{x}(t + \tau)' + \hat{u}(t + \tau)'^T R \hat{u}(t + \tau)' d\tau \\ &+ \int_T^{T+\delta} \hat{x}(t + \tau)'^T Q \hat{x}(t + \tau)' + \hat{u}(t + \tau)'^T R \hat{u}(t + \tau)' d\tau \end{aligned} \quad (51)$$

Comparing (50) with (52) yields

$$\begin{aligned} V(x(t + \delta))' - V(x(t)) &= g(\hat{x}(t + T + \delta)') - g(\hat{x}(t + T)) \\ &+ \int_T^{T+\delta} \hat{x}(t + \tau)'^T Q \hat{x}(t + \tau)' + \hat{u}(t + \tau)'^T R \hat{u}(t + \tau)' d\tau \\ &- \int_{\delta}^{2\delta} \hat{x}(t + \tau)'^T Q \hat{x}(t + \tau)' + \hat{u}(t + \tau)'^T R \hat{u}(t + \tau)' d\tau \end{aligned} \quad (52)$$

Remember that  $g(x)$  is continuously differentiable with respect to  $x$ . One has

$$\begin{aligned} &g(\hat{x}(t + T + \delta)') - g(\hat{x}(t + T)') \\ &= \int_T^{T+\delta} \frac{\partial g(\hat{x}(t + \tau)')}{\partial \hat{x}(t + \tau)'} \dot{\hat{x}}(t + \tau)' d\tau \\ &= \int_T^{T+\delta} \frac{\partial g(\hat{x}(t + \tau)')}{\partial x(t + \tau)'} f(\hat{x}(t + \tau)', \hat{u}(t + \tau)') d\tau \end{aligned} \quad (53)$$

Following from the definition of  $\hat{u}'$  in (23),

$$\hat{u}(t + \tau; x(t + \delta))' = \hat{u}(t + \tau; x(t)) \text{ for } \tau \in [t + \delta, t + T]. \quad (54)$$

This implies that

$$g(\hat{x}(t + T))' = g(\hat{x}(t + T)). \quad (55)$$

In the time interval  $[t + T, t + T + \delta]$ , the control  $\hat{u}(\tau)'$  satisfies condition (24).

Hence the following inequality holds

$$\begin{aligned} &\int_T^{T+\delta} \frac{\partial g(\hat{x}(t + \tau)')}{\partial x(t + \tau)'} f(\hat{x}(t + \tau)', \hat{u}(t + \tau)') d\tau \\ &\leq \int_T^{T+\delta} \hat{x}(t + \tau)'^T Q \hat{x}(t + \tau)' + \hat{u}(t + \tau)'^T R \hat{u}(t + \tau)' d\tau \end{aligned} \quad (56)$$

Invoking (56) and (57) with (54) gives

$$g(\hat{x}(t+T+\delta)) - g(\hat{x}(t+T)) \leq \int_T^{T+\delta} \hat{x}(t+\tau)^T Q \hat{x}(t+\tau) + \hat{u}(t+\tau)^T R \hat{u}(t+\tau) d\tau \quad (57)$$

Combining the above inequality with (53) yields

$$V(x(t+\delta)) - V(x(t)) \leq - \int_\delta^{2\delta} \hat{x}(t+\tau)^T Q \hat{x}(t+\tau) + \hat{u}(t+\tau)^T R \hat{u}(t+\tau) d\tau \quad (58)$$

In the proposed MPCGS algorithm, the optimisation procedure at time  $t+\delta$  starts from the control profile (23). Minimisation of the performance index (17) yields the control profile  $\hat{u}(\tau; x(t+\delta))$ . The performance cost under this control profile is given by  $J(x(t+\delta), \hat{u}(\cdot; x(t+\delta)))$ , ie.,  $V(x(t+\delta))$ . The Lyapunov function  $V(x(t+\delta))$  along the state trajectory satisfies

$$V(x(t+\delta)) \leq V(x(t+\delta))' \quad (59)$$

Combining (59) and (60) gives

$$V(x(t+\delta)) - V(x(t)) \leq - \int_\delta^{2\delta} \hat{x}(t+\tau)^T Q \hat{x}(t+\tau) + \hat{u}(t+\tau)^T R \hat{u}(t+\tau) d\tau \quad (60)$$

Remember that the control profile  $\hat{u}(\cdot; x(t))$  is implemented in the time interval  $[t+\delta, t+2\delta]$  until the next control profile is yielded by optimisation. That is,

$$u(t+\tau) = \hat{u}(t+\tau; x(t)), \tau \in [\delta, 2\delta] \quad (61)$$

and

$$x(t+\tau) = \hat{x}(t+\tau), \tau \in [\delta, 2\delta] \quad (62)$$

We have

$$\int_{t+\delta}^{t+2\delta} \hat{x}(\tau)^T Q \hat{x}(\tau) + \hat{u}(\tau)^T R \hat{u}(\tau) d\tau = \int_{t+\delta}^{t+2\delta} x(\tau)^T Q x(\tau) + u(\tau)^T R u(\tau) d\tau \quad (63)$$

The inequality (61) is equivalent to that of

$$V(x(t+\delta)) - V(x(t)) \leq - \int_\delta^{2\delta} x(t+\tau)^T Q x(t+\tau) + u(t+\tau)^T R u(t+\tau) d\tau \leq 0 \quad (64)$$

where the latter inequality follows from  $Q \geq 0$  and  $R > 0$ . Hence the Lyapunov function (50) is non-increasing along the closed-loop system's trajectory.

Repeatedly using the inequality (65) yields

$$\begin{aligned} V(x(\infty)) - V(x(t_0)) &\leq - \int_{t_0+\delta}^{\infty} x(\tau)^T Q x(\tau) + u(\tau)^T R u(\tau) d\tau \\ &\leq - \int_{t_0+\delta}^{\infty} x(\tau)^T Q x(\tau) d\tau. \end{aligned} \quad (65)$$

Since  $V(x)$  is monotonically non-increasing and bounded from below by zero, the integral in the right side of the inequality (66) converges and hence exists and is bounded.

In addition, it can be shown that  $x(t)$  is uniformly continuous in  $t$  on  $[t_0, \infty)$  under Assumption A1 (Desoer and Vidyasagar, 1975). Using Baralema's Lemma, this implies that  $x^T Q x \rightarrow 0$  as  $t \rightarrow \infty$  (Hahn, 1967, proof of Theorem 26.4) or (Khalil, 1992). When  $Q > 0$  or  $Q \geq 0$  and  $Q^{1/2}x \neq 0$  for all nonzero  $x$  such that  $f(x, 0) = 0$ , it is concluded that  $x(t) \rightarrow 0$  as  $t \rightarrow \infty$  (Sepulchre *et al.*, 1996; Magni and Sepulchre, 1997). Hence the equilibrium point  $x = 0$  of the system (12) is asymptotically stable. QED

## References

- Allgöwer, F. and Zheng, A., Eds.) (1998). *Proceedings of International Symposium on Nonlinear Model Predictive Control: Assessment and Future Directions*. Ascona, Switzerland.
- Biegler, L. T. and J. B. Rawlings (1991). Optimisation approaches to nonlinear model predictive control. In: *Proceedings of Conf. Chemical Process Control*. South Padre Island, Texas. pp. 543–571.
- Bitmead, R. R., M. Gevers and V. Wertz (1990). *Adaptive Optimal Control: The Thinking Man's GPC*. Prentice-Hall. New York.
- Casavola, A. and E. Mosca (1998). Predictive reference governor with computational delay. *European Journal of Control* **4**(3), 241–248.
- Chen, H. and F. Allgöwer (1998). A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica* **34**(10), 1205–1217.

- Chen, S. H., J. H. Chou and L. A. Zheng (1997). Robust stability measures for discrete-time linear regulators with computational delays. *JMSE International Journal, Series C* **40**(2), 267–271.
- Chen, Wen-Hua and Xingyuan Gu (1991). Robust stability analysis for discrete-time systems with computational delay. In: *Proceedings of Asia-Pacific measurement and Control Conference*. Guangzhou, P. R. of China.
- Chen, Wen-Hua, D. J. Ballance and J. O'Reilly (1999a). On attraction domain of model predictive control of nonlinear systems with input/state constraints. CSC Research Report CSC-99009. Centre for Systems and Control, University of Glasgow. URL <http://www.mech.gla.ac.uk/Control/reports.html>.
- Chen, Wen-Hua, D. J. Ballance and P. J. Gawthrop (1999b). Nonlinear generalised predictive control and optimal dynamic inversion control. In: *Proceedings of the 14th IFAC World Congress*. Vol. E. Beijing, P.R.C.. pp. 415–420.
- Chen, Wen-Hua, D. J. Ballance, P. J. Gawthrop, J. J. Gribble and J. O'Reilly (1999c). Nonlinear PID predictive controller. *IEE Proceedings Pt-D: Control Theory and Applications*. To appear.
- Clarke, D. W. (1994). *Advances in Model-based Predictive Control*. Oxford University Press.
- Clarke, D. W., C. Mohtadi and P. S. Tuffs (1987). Generalised predictive control—part I. the basic algorithm. *Automatica* **23**(2), 137–148.
- Demircioglu, H. and D. W. Clarke (1992). CGPC with guaranteed stability properties. *Proc. IEE* **139**(4), 371–380.
- Desoer, C. A. and M. Vidyasagar (1975). *Feedback Systems: Input-Output Properties*. Academic Press. London.
- Doyle, F. J., A. Babatunde, A. Ogunnaike and R. K. Pearson (1995). Nonlinear model-based control using second-order Volterra models. *Automatica* **31**(5), 697–714.

- Gawthrop, P. J., H. Demircioglu and I. Siller-Alcala (1998). Multivariable continuous-time generalised predictive control: A state-space approach to linear and nonlinear systems. *Proc. IEE Pt. D: Control Theory and Applications* **145**(3), 241–250.
- Genceli, H. and M. Nikolaou (1993). Robust stability analysis of constrained  $L_1$ -norm model predictive control. *AIChE. J.* **39**(12), 1954–1965.
- Gomma, H. W. and D. H. Owens (1998). Delayed generalised predictive control. In: *Proceedings of the UKACC conference "Control '98"*. Swansea, U.K.. pp. 793–797.
- Gyurkovics, E. (1998). Receding horizon control via Bolza-type optimisation. *Systems & Control Letters* **35**, 195–200.
- Hahn, W. (1967). *Stability of Motion*. Springer-Verlag. New York.
- Khalil, H. K. (1992). *Nonlinear Systems*. Macmillan. New York.
- Magni, L. and L. Sepulchre (1997). Stability margins of nonlinear receding-horizon control via inverse optimality. *Systems and Control Letters* **32**, 241–245.
- Mayne, D. Q. (1996). Nonlinear model predictive control: an assessment. In: *Proceedings of Conf. Chemical Process Control*. Lake Tahoe, CA.
- Mayne, D. Q. and H. Michalska (1990). Receding horizon control of nonlinear systems. *IEEE Trans. on Automatic Control* **35**(7), 814–824.
- Michalska, H. and D. Q. Mayne (1993). Robust receding horizon control of constrained nonlinear systems. *IEEE Trans. on Automatic Control* **38**(11), 1623–1633.
- Mosca, E. and J. Zhang (1992). Stable redesign of predictive control. *Automatica* **28**(6), 1229–1233.
- Mutha, R. K., W. R. Cluett and A. Penlidis (1997). Nonlinear model-based predictive control of control nonaffine systems. *Automatica* **33**(5), 907–913.

- Pierre, D. A. and J. W. Pierre (1995). Digital controller design— alternative emulation approaches. *ISA Transactions* **34**(3), 219–228.
- Polak, E., D. Mayne and D. M. Stimler (1984). Control system design via semi-infinite optimisation: a review. *Proceedings of the IEEE* **72**(12), 1777–1794.
- Poubelle, M. A., R. R. Bitmead and M. R. Gevers (1988). Fake algebraic Riccati techniques and stability. *IEEE Trans. on Automatic Control* **33**, 379–381.
- Qin, S. J. and T. A. Badgwell (1998). An overview of nonlinear model predictive control applications. In: *Proceedings of International Symposium on Nonlinear Model Predictive Control: Assessment and Future Directions* (F. Allgöwer and A. Zheng, Eds.). Ascona, Switzerland. pp. 128–145.
- Rattan, K. S. (1989). Compensating for computational delay in digital equivalent of continuous control systems. *IEEE Trans. on Automatic Control* **34**, 895–899.
- Rawlings, J. B. and K. R. Muske (1993). Nonlinear predictive control of feedback linearisable systems and flight control system design. *IEEE Trans. on Automatic Control* **38**(10), 1512–1516.
- Ronco, E., T. Arsan and P. J. Gawthrop (1999). Open-loop intermittent feedback control: Practical continuous-time GPC. *IEE Proceedings Pt-D.: Control Theory and Applications* **146**(5), 426–434.
- Rossiter, J. A., J. R. Gossner and B. Kouvaritakis (1996). Infinite horizon stable predictive control. *IEEE Trans. on Automatic Control* **41**(10), 1522–1527.
- Scokaert, P. O. M., D. Q. Mayne and J. B. Rawings (1999). Suboptimal model predictive control (feasibility implies stability). *IEEE Trans. on Automatic Control* **44**(3), 648–654.
- Sepulchre, R., M. Jankovic and P. V. Kokotovic (1996). *Constructive Nonlinear Control*. Springer-Verlag. Berlin.

- Sistu, P. B., R. S. Gopinath and B. W. Bequette (1993). Computational issues in nonlinear predictive control. *Computer and Chemical Engineering* **17**(4), 361–366.
- Vassiliadis, V. S., R. W. H. Sargent and C. C. Pantelides (1994). Solution of a class of multistage dynamic optimisation problems 1: problem without path constraints. *Ind. Eng. and Chem. Research* **33**(9), 2111–2122.
- Von Wissel, D., R. Nikoukhah, S. L. Campbell and F. Delebecque (1997). Effects of computational delay in descriptor-based trajectory tracking control. *International Journal of Control* **67**(2), 251–273.
- Zheng, A. (1997). A computationally efficient nonlinear MPC algorithm. In: *Proceedings of American Control Conference*. pp. 1623–1627.