SRC TR 89-4

Robust Control of Processes with Hard Constraints

by

E. Zafiriou

Robust Control of Processes with Hard Constraints *

Evanghelos Zafiriou
Chemical and Nuclear Engineering Department
and Systems Research Center
University of Maryland
College Park, MD 20742

Presented at the 1988 Annual AIChE Mtg., Washington, D. C.

Abstract

A significant number of Model Predictive Control algorithms solve on-line an appropriate optimization problem and do so at every sampling point. The major attraction of such algorithms, like the Quadratic Dynamic Matrix Control (QDMC), lies in the fact that they can handle hard constraints on the inputs (manipulated variables) and outputs of a process. The presence of such constraints results in an on-line optimization problem that produces a nonlinear controller, even when the plant and model dynamics are assumed linear. This paper provides a theoretical framework within which the stability and performance properties of such algorithms can be studied. Necessary and/or sufficient conditions for nominal and robust stability are derived and two examples are used to demonstrate their effectiveness.

1 Introduction

The problem of input saturation is present in almost every chemical system, even when the process dynamics can be assumed linear. In addition to the input constraints, safety and certain performance specifications also require the presence of hard constraints on some output and state variables. The urgency of rigorous theoretical work in this area has been repeatedly pointed out by the industry (e.g., [8]). An approach that has been tried in the chemical industry during the past few years is to on-line solve an appropriate optimization problem and to do so at every sampling point. The repeated application of such methods (e.g., Quadratic Dynamic Matrix Control (QDMC) [5] on industrial problems with considerable success indicate that sufficient degrees of freedom exist in these formulations. A drawback that has prohibited their widespread use is the fact that no exact tuning procedure for the optimization parameters exist and such tuning often has to be carried out on-line by experienced designers.

The presence of hard constraints in the on-line optimization problem produces a nonlinear controller even when the plant and model dynamics are assumed linear. The fact that the overall control system (plant + controller) is nonlinear makes the study of its properties quite involved, especially since no analytic expression is available for the controller. The problems are compounded when robustness with respect to model-plant mismatch is also considered, because no straighforward extension of the results of the Robust Linear Control Theory to this particular problem exists, even though the plant and model dynamics are assumed linear. Some efforts have been made recently [8,1] to achieve robustness by modifying the "min" optimization problem that is solved on-line to a "min max" problem that minimizes the objective function over all possible plants. One of the problems of this approach is that either the computations for solving the optimization problem are too time consuming to be carried out on-line at every sample point or to simplify the computations one has to use simplistic model uncertainty descriptions that are unrealistic. Another, potentially serious problem is the fact that these methods inherently assume that by solving the

*Supported in part by the National Science Foundation's Engineering Research Centers Program: NSFD CDR 8803012 Additional support was provided by Shell Development Co. through an unrestricted research grant.

"min max" problem to obtain a sequence of future inputs (manipulated variables) and then implementing the first one and repeating the computation at the next sample point, one is guaranteed robust stability and performance, provided that a sufficiently long horizon is used in the objective function. However, feedback from an uncertain plant exists in reality and it is not taken into account in the formulation of the optimization problem, which is an open-loop minimization of the objective function over all possible plants. This fact can conceivably lead to performance deterioration and instability. Note that the situation is quite different from studying (and guaranteeing) a stabilizing control algorithm when no model error is present, in which case the assumption is reasonable, although not proven for the general case.

The problems discussed just above, cannot possibly be satisfactorily addressed without considering the problem in its proper nonlinear framework. It is the author's opinion that instead of augmenting the objective functions to add robustness, an action that dramatically increases the computational load and at the same time produces no rigorous robustness guarantees, one should study the problem in its nonlinear nature, obtain conditions that guarantee nominal and robust stability and performance and tune the parameters of the original optimization problems (e.g., QDMC) to satisfy them.

2 Preliminaries

Although control algorithms of the type described in Section 1 have been applied to systems with nonlinear dynamic models (QDMC [4]), it is usually assumed that the dynamics are linear, the nonlinearity of the problem arising from the hard constraints. The properties of the controller are independent of the type of model description used for the plant (see, e.g., [9]). The impulse response description is a convenient one:

$$y(k+1) = H_1 u(k) + H_2 u(k-1) + \ldots + H_N u(k-N+1) \tag{1}$$

where y is the output vector, u is the input vector and N is an integer sufficiently large for the effect of inputs more than N sample points in the past on y to be negligible.

The QDMC-type algorithms [5,6,7,9] use a quadratic objective function that includes the square of the weighted norm of the predicted error (setpoint - predicted output) over a finite horizon in the future as well as penalty terms on u or Δu. The minimization of the objective function is carried out over the values of $\Delta u(\bar{k})$, $\Delta u(\bar{k}+1)$,..., $\Delta u(\bar{k}+M-1)$, where $\bar{k}$ is the current sample point and M a specified parameter. The minimization is subject to possible hard constraints on the inputs u, their rate of change Δu, the outputs y and other process variables usually referred to as associated variables. The details on the formulation of the optimization problem can be found in [10]. After the problem is solved on-line at $\bar{k}$, only the optimal value for the first input vector $\Delta u(\bar{k})$ is implemented and the problem is solved again at $\bar{k}+1$. The optimal $u(\bar{k})$ depends on the tuning parameters of the optimization problem, the current output measurement $y(\bar{k})$ and the past inputs $u(\bar{k}-1)$,..., $u(\bar{k}-N)$ that are involved in the model output prediction. Let f describe the result of the optimization:

$$u(k) = f(y(k), u(k-1), \ldots, u(k-N)) \tag{2}$$

The optimization problem of the QDMC algorithm can be written as a standard Quadratic Programming problem:

$$\min_{v} q(v) = \frac{1}{2} v^T G v + g^T v \tag{3}$$

subject to

$$A^T v \geq b \tag{4}$$

where

$$v = [\ \Delta u(\bar{k}) \ \ \ldots \ \ \Delta u(\bar{k}+M-1)\ ]^T \tag{5}$$

and the matrices G, A, and vectors g, b are functions of the tuning parameters (weights, horizon, M, some of the hard constraints). The vectors g, b are also linear functions of $y(\bar{k})$, $u(\bar{k}-1)$,..., $u(\bar{k}-N)$. For the optimal solution v^* we have [3]:

$$\begin{bmatrix} G & -\hat{A} \\ -\hat{A}^T & 0 \end{bmatrix} \begin{bmatrix} v^* \\ \lambda^* \end{bmatrix} = - \begin{bmatrix} g \\ \hat{b} \end{bmatrix} \tag{6}$$

where $\hat{A}^T$, $\hat{b}$ consist of the rows of A^T, b that correspond to the constraints that are active at the optimum and λ^* is the vector of the Lagrange multipliers. The optimal $\Delta u(\bar{k})$, descibed by (2), corresponds to the first m elements of the v^* that satisfies (6), where m is the dimension of u.

The special form of the LHS matrix in (6) allows the numerically efficient computation of its inverse in a partitioned form [3]:

$$\begin{bmatrix} G & -\hat{A} \\ -\hat{A}^T & 0 \end{bmatrix}^{-1} = \begin{bmatrix} H & -T \\ -T^T & U \end{bmatrix} \tag{7}$$

Then

$$v^* = -Hg + T\hat{b} \tag{8}$$

$$\lambda^* = T^T g - U\hat{b} \tag{9}$$

and

$$u(\bar{k}) = u(\bar{k}-1) + \begin{bmatrix} I & 0 & \dots & 0 \end{bmatrix} v^* \stackrel{\text{def}}{=} f(y(k), u(k-1), \dots, u(k-N)) \tag{10}$$

3 Stability Conditions

Some recent work by the author [11] used the Operator Control Theory framework [2], to study the properties of the overall nonlinear system. In this approach, the stability and performance of the nonlinear system can be studied by applying the contraction mapping principle on the operator F that maps the "state" of the system (plant + controller) at sample point k to that at sample point $k+1$. The fact that the plant dynamics are assumed linear allows us to obtain results and carry out computations that are not yet feasible in the general case. We can define as the "state" of the system at sample point k the following vector

$$x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_N(k) \end{bmatrix} \tag{11}$$

where

$$\begin{aligned} x_1(k+1) \stackrel{\text{def}}{=} u(k) &= f(y(k), u(k-1), \dots, u(k-N)) \\ &= f(H_1 u(k-1) + \dots + H_N u(k-N), \\ &\quad u(k-1), \dots, u(k-N)) \\ &\stackrel{\text{def}}{=} \Psi(u(k-1), \dots, u(k-N)) \\ &= \Psi(x(k)) \\ x_2(k+1) \stackrel{\text{def}}{=} u(k-1) &= x_1(k) \\ \vdots \qquad &\quad \vdots \\ x_N(k+1) \stackrel{\text{def}}{=} u(k-N+1) &= x_{N-1}(k) \end{aligned} \tag{12}$$

The "state" vector $x(k)$ is defined so that knowledge of it allows the computation of $x(k+1)$ by applying the plant and controller equations on it. Indeed the operator F that maps $x(k)$ to $x(k+1)$ is given by

$$x(k+1) = F(x(k)) = \begin{bmatrix} \Psi(x(k)) \\ x_1(k) \\ \vdots \\ x_{N-1}(k) \end{bmatrix} \tag{13}$$

Note, however, that although f is known, since it describes the on-line optimizing control algorithm and it involves only the process model, Ψ is not exactly known, because it involves the "true" plant impulse response coefficients $H_1, \dots, H_N$

Convergence of the successive substitution $x(k+1) = F(x(k))$ to the unique fixed point of the contraction implies stability of the overall nonlinear system; fast convergence implies good performance. The use of the contraction mapping principle allows the development of conditions for robust stability and performance in terms of some induced matrix norm of the derivative F' of the above operator F.

Let J_i be a set of indices for the active constraints of (3) and $J_1,\ldots, J_n$ correspond to all possible active sets of constraints when all xs in the domain of F are considered. Every such J_i corresponds to an $\hat{A}_i$ and a $\hat{b}_i$. It was shown in [11] that for all xs that correspond to the same J_i and for which an infinitecimal change in their value does not change the set of active constraints, the derivative of Ψ and therefore of F exist and it has the same value that depends on the particular set J_i:

$$F'_{J_i} = \begin{bmatrix} (\nabla_{x_1}\Psi)_{J_i} & (\nabla_{x_2}\Psi)_{J_i} & \ldots & (\nabla_{x_{N-1}}\Psi)_{J_i} & (\nabla_{x_N}\Psi)_{J_i} \\ I & 0 & \ldots & 0 & 0 \\ 0 & I & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & I & 0 \end{bmatrix} \tag{14}$$

where from (12) it follows that

$$(\nabla_{x_j}\Psi)_{J_i} = (\nabla_{x_j}f)_{J_i} + (\nabla_y f)_{J_i} H_j \tag{15}$$

The derivatives of f can be computed easily from (10):

$$(\nabla_{x_j}f)_{J_i} = \begin{bmatrix} I & 0 & \ldots & 0 \end{bmatrix} (-H_{J_i}\nabla_{x_j}g + T_{J_i}\nabla_{x_j}\hat{b}_i) \tag{16}$$

where the derivatives of g, b_i are constant since g, b are linear functions of $y(\bar{k})$, $u(\bar{k}-1),\ldots, u(\bar{k}-N)$. The same expression as in (16) is also true for the derivative with respect to $y(\bar{k})$, the current measurement. Also note that in the case of x_1, the identity matrix I should be added to the RHS of (16).

It turns out that $F(x)$ is quasi-linear and that it is differentiable everywhere except the points where an infinitecimal change will change the set of active constraints at the optimum of (3). The following theorems were proven in [11]. The terms stability and instability of the control system are used in the global sense over the domain of F under consideration.

Theorem 1 *F is a contraction if and only if there exists a consistent matrix norm $||.||$, for which*

$$||F'_{J_i}|| < 1, \quad i = 1, \ldots, n \tag{17}$$

The practical use of (17) is limited by the fact that finding an appropriate consistent norm is not a trivial task. The following three theorems provide conditions which are more readily computable.

Theorem 2 *The control system is asymptotically stable if*

$$||\ (\nabla_{x_1}\Psi)_{J_i} \quad (\nabla_{x_2}\Psi)_{J_i} \quad \ldots \quad (\nabla_{x_N}\Psi)_{J_i}\ ||_\infty < 1, \quad i = 1, \ldots, n \tag{18}$$

where

$$||B||_\infty = \max_i \sum_{j=1}^{N} |b_{ij}| \tag{19}$$

Note that for single-input single-output plants (18) becomes

$$\sum_{j=1}^{N} |\frac{\partial \Psi_{J_i}}{\partial x_j}| < 1, \quad i = 1, \ldots, n \tag{20}$$

which for the unconstrained case is simply a sufficient condition for the closed-loop poles to lie inside the Unit Circle.

Theorem 3 *F can be a contraction only if*

$$\rho(F'_{J_i}) < 1, \quad i = 1, \ldots, n \tag{21}$$

where $\rho(A)$ is the spectral radius of A. Note that if the optimization (3) is not subject to (4), then $n = 1$ and (21) becomes sufficient as well, because, given a matrix one can always find a consistent norm arbitrarily close to its spectral radius. The reason that (21) is not sufficient in general is that such a consistent norm is in general a different one for two different matrices (different J_is), while (17) requires the same norm for all i. In the case of $n = 1$, (21) translates to the requirement that the closed-loop poles of the system are located inside the Unit Circle.

If (21) is not true, then F is not a contraction. This however does not necessarily imply that the control system is unstable. The following theorem provides a condition that is sufficient for instability.

Theorem 4 *The control system is unstable if*

$$\rho(F'_{J_i}) > 1, \quad i = 1, \ldots, n \tag{22}$$

Theorem 4 can be used to predict instability of the overall nonlinear system. Theorem 3 on the other hand does not seem at a first glance to be of much use, since violation of (21) does not necessarily imply instability. From a practical point of view, however, violation of that condition for some i, should be taken as a very serious warning that the control system parameters should be modified. The reason is that when in the region of the domain of F that corresponds to that i, the system will behave as a virtually unstable system, the only hope for stability being to move to a region with $\rho(F'_{J_i}) < 1$. It might be the case that for a particular system in question this will always happen, making this system a stable one. But even in this case, a temporary unstable-like behavior might occur, thus making the control algorithm practically unacceptable.

From (15) we see that F'_{J_i} depends on the impulse response coefficient matrices $H_1, \ldots, H_N$ of the actual plant. These matrices are never known exactly and so in order to guarantee stability for the actual plant, one has to compute the conditions of Section 3 not just for the model, but for all possible plants. To do so, one needs to have some informat n on the possible modeling error associated with the H_is. Let $\mathcal{H}$ be the set of possible values for these coefficients. Then

Theorem 5 *The control system is asymptotically stable for all plants with coefficients in* $\mathcal{H}$ *if*

$$\sup_{\mathcal{H}} \| \; (\nabla_{x_1}\Psi)_{J_i} \quad (\nabla_{x_2}\Psi)_{J_i} \quad \ldots \quad (\nabla_{x_N}\Psi)_{J_i} \; \|_\infty < 1, \quad i = 1, \ldots, n \tag{23}$$

Theorem 6 *F can be a contraction for all plants with coefficients in* $\mathcal{H}$ *only if*

$$\sup_{\mathcal{H}} \rho(F'_{J_i}) < 1, \quad i = 1, \ldots, n \tag{24}$$

4 A Robust Linear Control Stabilization Interpretation of the Necessary Stability Conditions

In order to carry out the maximizations over $\mathcal{H}$ described by (24), (23), one needs to parametrize the "uncertain" $H_1, \ldots, H_N$, in terms of a fewer "uncertain" parameters. For example, in the simple case where the linear plant dynamics are described by the transfer function $\frac{K}{\tau s+1}$, where K, τ, are within some ranges, we can write $H_1, \ldots, H_N$, as functions of K, τ, and compute $\sup_{\mathcal{H}}$ as $\sup_{K,\tau}$. However, the situation is usually more complex, a fact that makes the efficient parametrization of the modeling error in $H_1, \ldots, H_N$, a very important research topic.

The following re-formulation of the necessary conditions of the previous section, allows us to bypass the problem of dealing with uncertainty in the Hs directly, and use the tools that were developed for Robust Linear Control (e.g., the structured singular value) to treat any of the types of model error that can be handled by that theory. Consider a standard feedback controller $C(z)$. Then

$$u(z) = C(z)(r(z) - y(z)) \tag{25}$$

where r is the setpoint vector. Define

$$C_{J_i}(z) \stackrel{\text{def}}{=} -\left[I - (\nabla_{x_1}f)_{J_i} z^{-1} - \ldots - (\nabla_{x_N}f)_{J_i} z^{-N}\right]^{-1} (\nabla_y f)_{J_i} \tag{26}$$

Since the plant is assumed to be open-loop stable, for stability of this linear control system we need that the closed-loop transfer function between u and r or d (disturbance) be stable. From (25), (26) we get by using (1)

$$u(z) = -\left[I - (\nabla_{x_1}\Psi)_{J_i} z^{-1} - \ldots - (\nabla_{x_N}\Psi)_{J_i} z^{-N}\right]^{-1} (\nabla_y f)_{J_i} r(z) \tag{27}$$

where $(\nabla_{x_j}\Psi)_{J_i}$ is given by (15). Hence, provided that the truncation number N is chosen large enough so that the effect of further terms is negligible for both the model and the plant, stability of the system under feedback control $C_{J_i}(z)$ is equivalent to stability of the transfer matrix in (27), which is equivalent to (21) since F'_{J_i} is the companion matrix of the denominator of (27). Hence we have

Theorem 7 *F can be a contraction only if all feedback controllers $C_{J_i}(z)$, $i = 1, \ldots, n$, produce a stable system when applied to the unconstrained process.*

Theorem 8 *F can be a contraction for all plants in a set Π, only if all feedback controllers $C_{J_i}(z)$, $i = 1, \ldots, n$, stabilize all plants in the set Π.*

The advantage of Thm. 8 over Thm. 6 lies in the fact that through Thm. 8 we can handle any set Π that Robust Linear Control theory can. This new interpretation of the conditions also indicates that robust performance conditions can be formulated for the same set of feedback controllers. For the sufficient conditions a similar formulation may be possible but it would probably involve some conservativeness.

5 Practical Interpretation of a Condition Violation

Conditions (21), (18) can be used to examine the stability of the system for a particular selection of tuning parameters. An important question however is what are the implications if for a particular $\hat{A}_i$ the conditions are not satisfied. This would only be relevant if the particular combination of active constraints at the optimum can actually occur during the operation of the control system. The following is a procedure that can decide if a certain set of active constraints at the optimum is relevant.

Let $\check{A}^T$, $\check{b}$ consist of the rows of A^T, b that correspond to the inactive constraints at the optimum. Then by using (8), (9) we see that in order for such a combination to be possible at the optimum we need to have

$$\check{A}^T(-Hg + T\hat{b}) > \check{b} \tag{28}$$

$$T^T g - U\hat{b} \geq 0 \tag{29}$$

Since g, b are linear combinations of the past manipulated variables and the current measurement, (28), (29) can be combined with the hard constraints on the past us, the past Δus and the output $y(\bar{k})$ to constitute a system of linear inequalities that have to have a feasible solution over the values of the past inputs and the current measurement. Note that depending on the estimate of expected disturbances, one may wish to modify the bounds on $y(\bar{k})$ that are used in the above problem. If the problem has no feasible solution, then the fact that for that particular $\hat{A}$ the stability conditions are not satisfied, is of no practical importance.

Note that the above procedure can also serve to construct a sequence of possible past inputs that can lead to a situation during the operation of the control system where the stability conditions are not satisfied.

6 Illustrations

In this section two examples are given, to demonstrate the effectiveness of the nominal and robust stability conditions and the nonlinear behavior of the control system. The examples are simple so that the effect of incuding hard constraints in the on-line optimization problem is clear.

6.1 Nominal Stability of a 2×2 process

Let us consider a system with the following transfer function:

$$P(s) = \begin{bmatrix} \frac{1}{s+1} & 0 \\ \frac{-2e^{-5s}}{s+1} & \frac{-s+2}{(s+2)(s+1)} \end{bmatrix} \tag{30}$$

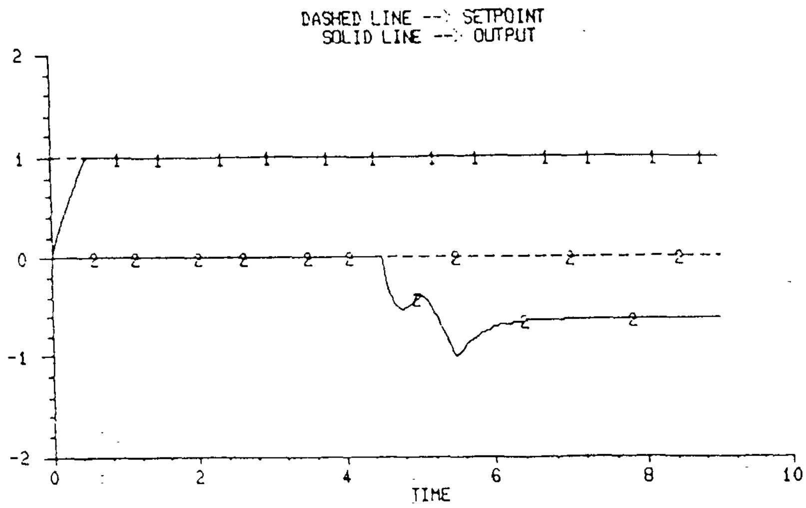

Figure 1: Unconstrained minimization.

A sampling time $T = 0.5$ is used and the following objective function is minimized on-line:

$$\min_{u(\bar{k}),\ldots,u(\bar{k}+M-1)} \sum_{l=1}^{P} \left[e(\bar{k}+l)^T \Gamma^2 e(\bar{k}+l) + u(\bar{k}+l-1)^T B^2 u(\bar{k}+l-1) \right] \tag{31}$$

where $\bar{k}$ is the current sample point, e is the predicted difference between the setpoints and the plant outputs and Γ, B, are weights.

$$\Gamma = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix} \tag{32}$$

is selected signifying that the first output is more important than the second.

Let us first consider the unconstrained problem. First we select $P = M = 2$, which is a selection that is expected [6,7] to produce an unstable control system if $B = 0$. The reason is the right-half plane (RHP) zero of $P(s)$. Indeed, one can easily check that for these values of the tuning parameters, we have $\rho(F'_{J_1}) > 1$, where J_1 corresponds to the case where no constraints are active at the optimum. Hence the necessary condition (21) predicts the instability. From theory [7] we know that by making B sufficiently large, we can stabilize the system. Indeed by making

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0.1 \end{bmatrix} \tag{33}$$

the system is stabilized ($\rho(F'_{J_1}) < 1$, which is sufficient for $n = 1$). The fact that the RHP zero is pinned to the second plant output, made it unnecessary to increase the 11 element of B. The response to a unit step change in setpoint 1 is shown in Fig. 1. The steady-state offset in output 2 is expected from theory and can be avoided by modifying the control algorithm, but we will not do so to avoid the unnecessary complication of the example.

Let us now assume that after looking at the response, the designer decides that a slight tightening of the specifications is in order, namely the addition in the optimization problem of a lower bound on output 2 at the value -0.9. Since output 2 only slightly violated this bound when the unconstrained algorithm was used, one might think that the response for the constrained algorithm should be almost the same as that in Fig. 1. This is not so, however. The response for the same setpoint change is shown in Fig. 2. The system is unstable. An instability warning was issued by the necessary condition for F to be a contraction (21), since $\rho(F'_{J_2}) > 1$, where J_2 corresponds to the case where the low constraint on output 2 is active at the optimum. Indeed by looking at a close-up of Fig. 2 in Fig. 3, we see that the system went unstable as soon as output

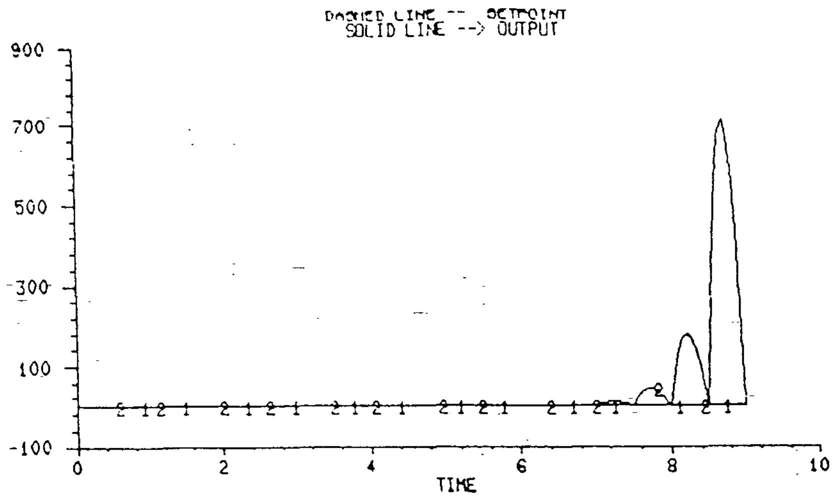

Figure 2: Minimization subject to lower bound constraint on output 2.

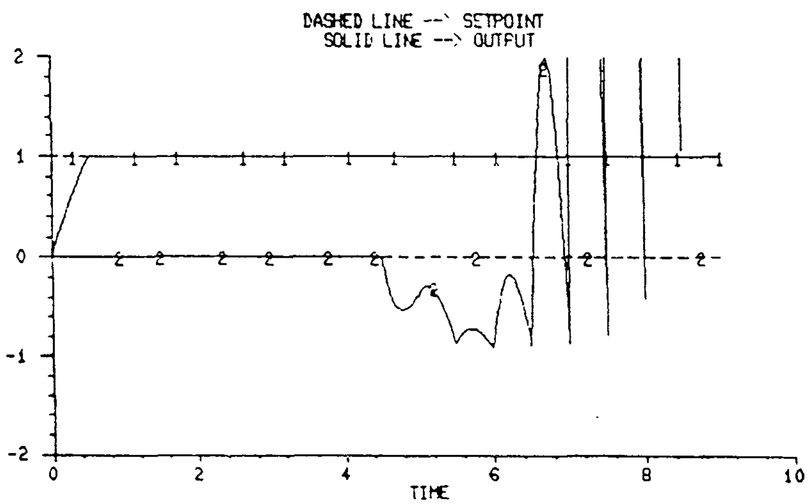

Figure 3: Close-up of Fig. 2.

2 reached the low bound to which the on-line minimization was subject. The constraint remained active at the subsequent sample points and the system never stabilized.

A question that one may ask at this point is whether the use of a

$$B = \begin{bmatrix} 0 & 0 \\ 0 & \beta \end{bmatrix} \tag{34}$$

with a β larger than the previously used value of 0.1, will stabilize the system. We know that this would be the case for the unconstrained problem; however, for the constrained case that does not happen. By examining the analytic expression for F'_{J_2} one sees that β does not even appear in it and can therefore in no way influence the stability of the system when the constraint becomes active. When the constraint is reached, the algorithm puts as its higher priority keeping output 2 above the lower bound and to do so it inverts the 22 element of $P(s)$ and causes instability.

6.2 Robust Stability of a SISO process

Consider the process model

$$\tilde{p}(s) = \frac{1}{s+1} \tag{35}$$

A sampling time $T = 0.1$ will be used and the control algorithm will minimize on-line the objective function

$$\min_{u(\bar{k}),\ldots,u(\bar{k}+M-1)} \sum_{l=1}^{P} \left[e(\bar{k}+l)^T \Gamma^2 e(\bar{k}+l) + \Delta u(\bar{k}+l-1)^T D^2 \Delta u(\bar{k}+l-1) \right] \tag{36}$$

To allow the analytic study of the properties of the control system we shall choose the parameters to be $P = M = \Gamma = 1$. A choice of $D = 0$, when there are no hard constraints, will result in an IMC controller that inverts the model [6].

Let us now consider a model-plant mismatch caused by a delay term in the plant:

$$p(s) = \frac{e^{-0.15s}}{s+1} \tag{37}$$

For this plant, robust linear control theory can easily show that the control system will be unstable for $D = 0$. D has to ber increased over $D = 0.2$ to stabilize it. The choice $D = 0.4$ results in reasonable performance.

Our interest in this example has to do with the effect of hard constraints on its output. Let us specify a lower bound of -1 and an upper bound of $+1$ for y and include these constraints in the on-line optimization problem. Since the horizon $P = 1$, it is not possible for both to be active at the optimum. In this case $n = 3$, corresponding to (i) no active constraints, (ii) upper constraint active, (iii) lower constraint active. Analytic computaion of $c_{J_i}(z)$, $i = 1, 2, 3$, results in the expressions

$$c_{J_1}(z) = \tag{38}$$

$$H_1 / \left[(D^2 + H_1^2) + (H_1 H_2 - H_1^2 - D^2) z^{-1} + H_1(H_2 - H_1) z^{-2} + \ldots + H_1(H_N - H_{N-1}) z^{-N+1} - H_1 H_N z^{-N} \right] \tag{39}$$

$$c_{J_2}(z) = c_{J_3}(z) = \tag{40}$$

$$1 / \left[H_1 + (H_2 - H_1) z^{-1} + \ldots + (H_N - H_{N-1}) z^{N+1} - H_N z^{-N} \right] \tag{41}$$

One can easily see from these exressions that c_{J_2} and c_{J_3} correspond to an IMC controller that inverts the process model, the same as c_{J_1} for $D = 0$. The difference is that D does not appear in (41) and therefore this controller will be unstable when the model-plant mismatch is present. The question that arises now, is the one discussed in Section 5. For the case of the upper constraint and for a setpoint equal to zero, (29) predicts that if the system is at equilibrium, a disturbance of magnitude greater than 1.6 will result in an on-line optimization where the upper constraint is active. The system could however manage to return to the contraction region of no active constraints. Indeed for a disturbance of 1.7, as Fig. 4 shows, the system is still stable, although at the edge of instability. An increase of the disturbance to 1.75 however results in an unstable system as Fig. 5 shows. Note that $D = 0.4$ is being used; although D does not appear in (41).

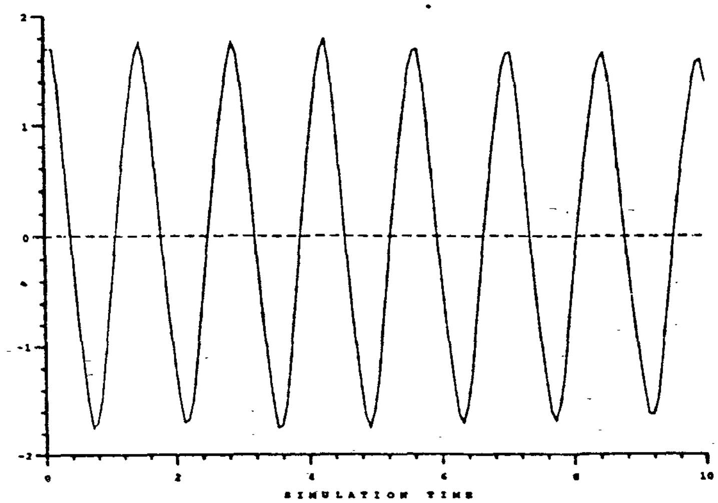

Figure 4: Constrained; D=0.4 and d=1.70

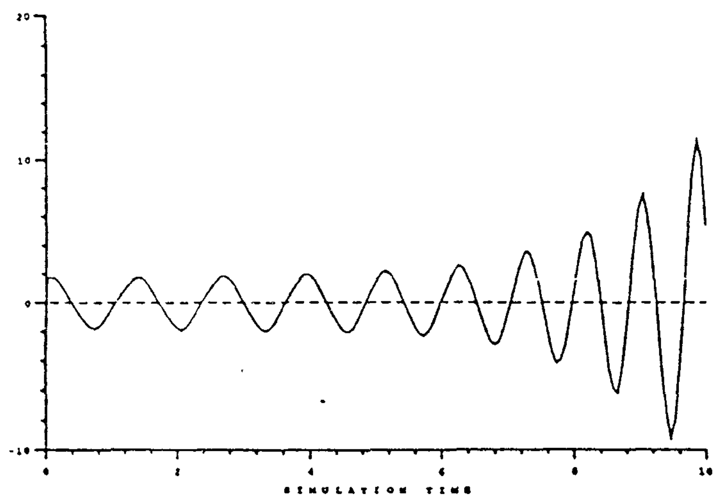

Figure 5: Constrained; D=0.4 and d=1.75

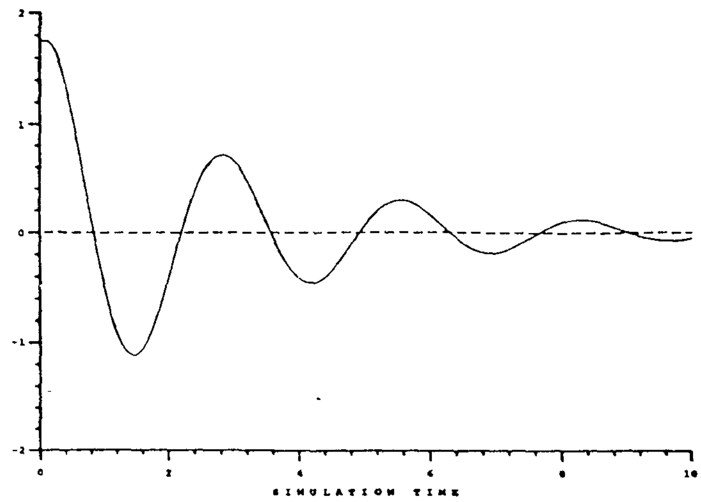

Figure 6: Unconstrained; D=0.4 and d=1.75

it does play a role on whether the constraints are active at the optimum. Both simulations use the plant of (37).

Let us now remove the constraints from the optimization problem and repeat the simulation for the same $d = 1.75$ and $D = 0.4$. The result is shown in Fig. 6. The response is reasonable and the constraints are virtually satisfied, although they were removed from the optimization problem. This example is not meant to suggest that output constraints should not be included in the optimization, but merely to point out that their effect should be studied carefully before their inclusion and to demonstrate that the stability conditions that were provided in this paper can predict this effect successfuly.

7 Conclusions

This paper has provided a theoretical framework for the study of the properties of control algorithms that are based on the on-line minimization of some objective function, subject to certain hard constraints. The selected framework seems to be very promising since it allowed the derivation of necessary and/or sufficient conditions for nominal and robust stability of the overall nonlinear system. These conditions can be formulated in a way that allows the treatment of the kinds of model-plant mismatch that robust linear control theory can handle.

The simple examples that were used demonstrated in a clear way that one cannot afford to neglect the nonlinear phenomena caused by the hard constraints to which the on-line optimization is subject. This example also indicates that inclusion of hard constraints on the plant outputs in the specifications can cause serious problemsa and that their effct should be carefully studied before they are used. The stability conditions of this paper can be used in this study.

Acknowledgements

The control software package CONSYD, developed at Caltech (Dr. M. Morari' group) and the University of Wisconsin (Dr. W. H. Ray's group), was used in the simulations.

References

[1] P. J. Campo and M. Morari, "Robust Model Predictive Control", Proc. Amer. Control Conf., p.1021,

Minneapolis MN, 1987.

[2] C. G. Economou, *An Operator Theory Approach to Nonlinear Controller Design*, Ph.D. Thesis, Caltech, 1985.

[3] R. Fletcher, *Practical Methods of Optimization; vol. 2: Constrained Optimization*, John Wiley and Sons, 1981.

[4] C. E. Garcia, "Quadratic Dynamic Matrix Control of Nonlinear Processes: An Application to a Batch Reaction Process", AIChE Ann. Mtg., San Francisco CA, 1984.

[5] C. E. Garcia and A. M. Morshedi, "Quadratic Programming Solution of Dynamic Matrix Control (QDMC)", Chem. Eng. Commun., **46**, pp. 73-87, 1986.

[6] C. E. Garcia and M. Morari, "Internal Model Control. 1. A Unifying Review and Some New Results", Ind. Eng. Chem. Process Des. Dev., **21**, 308-323, 1982.

[7] C. E. Garcia and M. Morari, "Internal Model Control. 3. Multivariable Control Law Computation and Tuning Guidelines", Ind. Eng. Chem. Process Des. Dev., **24**, 484-494, 1985.

[8] C. E. Garcia and D. M. Prett, "Advances in Industrial Model Predictive Control", Chemical Process Control Conf. III, Asilomar CA, 1986.

[9] M. Morari, C. E. Garcia and D. M. Prett, "Model Predictive Control: Theory and Practice", IFAC Workshop on Model–Based Process Control, Atlanta GA, 1988.

[10] D. M. Prett and C. E. Garcia, *Fundamental Process Control*, Butterworth Publishers, 1988.

[11] E. Zafiriou, "Robustness and Tuning of On-Line Optimizing Control Algorithms", Proc. of the IFAC Workshop on Model Based Process Control (Atlanta, GA), Pergammon Press, 1988.

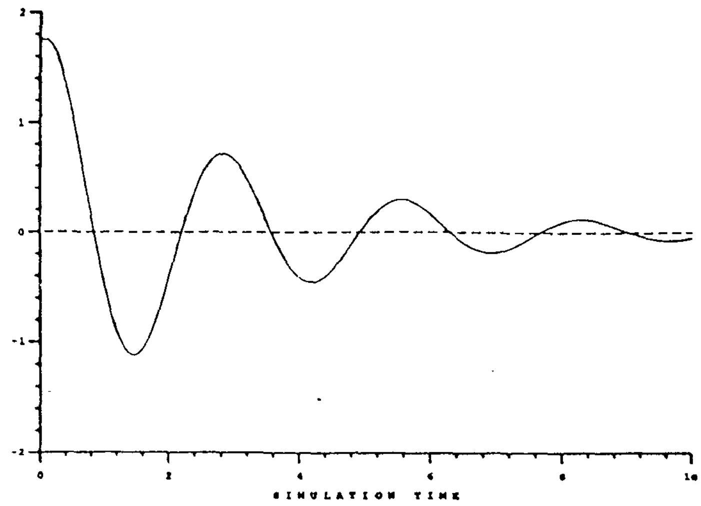

Figure 6: Unconstrained; D=0.4 and d=1.75

it does play a role on whether the constraints are active at the optimum. Both simulations use the plant of (37).

Let us now remove the constraints from the optimization problem and repeat the simulation for the same $d = 1.75$ and $D = 0.4$. The result is shown in Fig. 6. The response is reasonable and the constraints are virtually satisfied, although they were removed from the optimization problem. This example is not meant to suggest that output constraints should not be included in the optimization, but merely to point out that their effect should be studied carefully before their inclusion and to demonstrate that the stability conditions that were provided in this paper can predict this effect successfuly.

7 Conclusions

This paper has provided a theoretical framework for the study of the properties of control algorithms that are based on the on-line minimization of some objective function, subject to certain hard constraints. The selected framework seems to be very promising since it allowed the derivation of necessary and/or sufficient conditions for nominal and robust stability of the overall nonlinear system. These conditions can be formulated in a way that allows the treatment of the kinds of model-plant mismatch that robust linear control theory can handle.

The simple examples that were used demonstrated in a clear way that one cannot afford to neglect the nonlinear phenomena caused by the hard constraints to which the on-line optimization is subject. This example also indicates that inclusion of hard constraints on the plant outputs in the specifications can cause serious problemsa and that their effct should be carefully studied before they are used. The stability conditions of this paper can be used in this study.

Acknowledgements

The control software package CONSYD, developed at Caltech (Dr. M. Morari' group) and the University of Wisconsin (Dr. W. H. Ray's group), was used in the simulations.

References

[1] P. J. Campo and M. Morari, "Robust Model Predictive Control", Proc. Amer. Control Conf., p.1021,