TECHNICAL RESEARCH REPORT

S Y S T E M S
R E S E A R C H
C E N T E R

Supported by the
National Science Foundation
Engineering Research Center
Program (NSFD CD 8803012),
Industry and the University

An Operator Control Theory Approach to the Design and Tuning of Constrained Model Predictive Controllers

by E. Zafiriou and H.W. Chiou

TR 89-93

An Operator Control Theory Approach to the Design and Tuning of Constrained Model Predictive Controllers *

Evanghelos Zafiriou[†] and Hung-Wen Chiou

Chemical Engineering
and Systems Research Center
University of Maryland
College Park, MD 20742

Paper 21e
1989 Annual AIChE Meeting
November 5-10, 1989
San Francisco, CA

*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.

[†] To whom correspondence should be addressed

Abstract

Model Predictive Control algorithms minimize on-line and at every sampling point an appropriate objective function, subject to the satisfaction of possible hard constraints on the process outputs, inputs or other state variables. The presence of the hard constraints in the on-line optimization problem results in a nonlinear closed-loop system, even though the process dynamics are assumed linear. This paper describes a procedure for analyzing the nominal and robust stability properties of such control laws, by utilizing the Operator Control Theory framework.

1 Preliminaries

This section sets some notation for later use. An impulse response model description is used:

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

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 plant is assumed to be open-loop stable, but it may be non-square.

A quadratic objective function is used in this paper, in the lines of Quadratic Dynamic Matrix Control (Garcia and Morshedi, 1986):

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

where $\bar{k}$ is the current sample point. 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. 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:

$$u(k) = f(y(k), u(k-1), \ldots, u(k-N), r_P(k)) \tag{3}$$

where $r_P(k)$ includes all the values of the reference signal (setpoint) during the prediction horizon from $k+1$ to $k+P$.

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

$$\min_v q(v) = \frac{1}{2} v^T G v + g^T v \tag{4}$$

subject to

$$A^T v \geq b \tag{5}$$

where

$$v = [\ \Delta u(\bar{k}) \ \ \ldots \ \ \Delta u(\bar{k}+M-1)\ ]^T \tag{6}$$

and the matrices G, A, and vectors g, b are functions of the tuning parameters (weights, horizon P, 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 (Fletcher, 1981):

$$\begin{bmatrix} G & -\hat{A} \\ -\hat{A}^T & 0 \end{bmatrix} \begin{bmatrix} v^* \\ \lambda^* \end{bmatrix} = - \begin{bmatrix} g \\ \hat{b} \end{bmatrix} \tag{7}$$

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 corresponding to these constraints. The optimal $\Delta u(\bar{k})$ corresponds to the first m elements of the v^* that solves (7), where m is the dimension of u.

The special form of the LHS matrix in (7) allows the numerically efficient computation of its inverse in a partitioned form:

$$\begin{bmatrix} G & -\hat{A} \\ -\hat{A}^T & 0 \end{bmatrix}^{-1} = \begin{bmatrix} H & -T \\ -T^T & U \end{bmatrix} \tag{8}$$

Then

$$v^* = -Hg + T\hat{b} \tag{9}$$

$$\lambda^* = T^T g - U\hat{b} \tag{10}$$

and

$$u(\bar{k}) = u(\bar{k}-1) + \underbrace{\begin{bmatrix} I & 0 & \dots & 0 \end{bmatrix}}_{M} v^* \stackrel{\text{def}}{=} f(y(\bar{k}), u(\bar{k}-1), \dots, u(\bar{k}-N), r_P(k)) \tag{11}$$

2 Stability Conditions

The framework selected for the study of the properties of the overall nonlinear system is that of the Operator Control Theory (Economou, 1985). 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

$$x(k) = \begin{bmatrix} u(k) \\ \vdots \\ u(k-N+1) \end{bmatrix} \tag{12}$$

The "state" vector $x(k)$ is defined so that knowledge of it and of the external inputs (setpoints and disturbances) allows the computation of $x(k+1)$ by applying the plant and controller equations on it. Note, however, that this operator is "uncertain" since it involves the actual plant, i.e., the "true" plant impulse response coefficients H_1,..., H_N.

Convergence of the successive substitution $x(k+1) = F(x(k))$ implies stability of the overall nonlinear system; fast convergence implies good performance. Zafiriou (1989a) obtained sufficient and necessary stability conditions for the overall nonlinear system by obtaining conditions under which the mapping described by F is a contraction. The terms stability and instability of the control system are used in the global asymptotical sense over the domain of F under consideration. These conditions were shown to be able to capture the nonlinear characteristics of the constrained controller and were used to analyze a 2×2 subsystem of the Shell Heavy Oil Fractionator (model published in Prett and Garcia, 1988).

In this paper we shall concentrate on a necessary condition for the closed-loop operator to be a contraction. For this condition an interpretation has been obtained that allows us to bypass the problem of dealing with model uncertainty in the time domain, and use the tools that were developed for Robust Linear Control (e.g., the structured singular value (Doyle, 1982)) to treat any of the types of model error that can be handled by that theory. Let J_i be a set of indices for the active constraints of (4) and $J_1,\ldots, J_n$ correspond to all possible active sets of constraints when all xs in the domain of F are considered. For each J_i define a standard feedback controller $C_{J_i}(z)$:

$$C_{J_i}(z) \stackrel{\text{def}}{=} -\left[I - (\nabla_{x_1} f)_{J_i} z^{-1} - \ldots - (\nabla_{x_N} f)_{J_i} z^{-N}\right]^{-1} (\nabla_y f)_{J_i} \tag{13}$$

Also define the transfer matrices:

$$Q_{J_i}(z) \stackrel{\text{def}}{=} \left[I - (\nabla_{x_1} f)_{J_i} z^{-1} - \ldots - (\nabla_{x_N} f)_{J_i} z^{-N}\right]^{-1} \tag{14}$$

Then the following theorems hold (Zafiriou,1989a):

Theorem 1 *F can be a contraction only if all feedback controllers $C_{J_i}(z)$, $i \ni (\nabla_y f)_{J_i} \neq 0$, produce a stable system when applied to the linear unconstrained process and all transfer matrices $Q_{J_i}(z)$, $i \ni (\nabla_y f)_{J_i} = 0$, are stable.*

Theorem 2 *F can be a contraction for all plants in a set Π, only if all feedback controllers $C_{J_i}(z)$, $i \ni (\nabla_y f)_{J_i} \neq 0$, stabilize all unconstrained plants in the set Π and all transfer matrices $Q_{J_i}(z)$, $i \ni (\nabla_y f)_{J_i} = 0$, are stable.*

These theorems allow us to can handle any set Π that Robust Linear Control theory can (for a discussion of the possible Πs see Morari and Zafiriou, 1989). The above interpretation of the contraction conditions also indicates that robust performance conditions can be formulated for the same set of feedback controllers. One should note that violation of the above conditions does not necessarily imply instability. From a practical point of view, however, violation 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 stable region. 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.

3 Search for Practically Relevant Sets of Active Constraints

3.1 Practical Relevance of a J_i

The number of all possible combinations of active constraints for the on-line optimization problem is very large for any control problem of reasonable complexity. A particular J_i, however, is relevant to the stability question only if that combination of active constraints at the optimum can actually occur during the operation of the control system. Let us use the subscript i in $\hat{A}_i$, $\hat{b}_i$ to denote that they correspond to a particular J_i, $i = 1, ..., n$. Also let $\check{A}_i^T$, $\check{b}_i$ consist of the rows of A^T, b that correspond to the inactive constraints at the optimum. Then by using (9), (10) we see that in order for such a combination to be possible at the optimum we need to have

$$\check{A}_i^T(-H_i g + T_i \hat{b}_i) \geq \check{b}_i \tag{15}$$

$$T_i^T g - U_i \hat{b}_i \geq 0 \tag{16}$$

Equation (15) is the requirement that the inactive constraints should be satisfied for the solution v^* that is given by (9). Equation (16) requires that the Lagrange multipliers have the correct sign so that v^* is indeed optimal. Since g, b are linear combinations of the past manipulated variables and the current measurement, (15), (16) can be combined with the hard constraints on the past us, the past Δus and the current output measurement y 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 "past" in this context refers to the QP parameters that correspond to points $\bar{k} - 1, ..., \bar{k} - N$, and current to $\bar{k}$. Note that the constraint on $y(\bar{k})$ is really an estimate, since any constraints that may have been placed on it in the QP may not have been satisfied due to modeling error and unmeasured disturbances. Also, it may be better to use a constraint on the estimated possible difference between the plant and model output $(y(\bar{k}) - \tilde{y}(\bar{k}))$. If the problem has no feasible solution, then that particular J_i 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.

3.2 Search over i

The procedure of section 3.1 determines whether one particular $C_{J_i}(z)$ needs to be checked for stability. Applying this test however to all possible J_i could require a tremendous amount of computational effort for even moderately complex control problems. A search method is needed that would allow us, when a particular J_i fails the test, to discard not only that J_i but also a whole class of other J_is without having to test them. A reasonable candidate for this "class" is all J_is with the same plus more active constraints than the one that failed the test. If the nature of the relevance test is such that it allows this inference, then a tree-like search procedure could implemented. This is described schematically in Table 3.2, for the case of four one-sided constraints.

0 0 0 0	0 0 0 0	
1 0 0 0	1 0 0 0	
1 1 0 0	1 1 0 0	<- not relevant
1 1 1 0	1 1 1 0	<- skip
1 1 1 1	1 1 1 1	<- skip
1 0 1 1	1 0 1 1	
1 0 0 1	1 0 0 1	
0 1 0 0	0 1 0 0	
0 1 1 0	0 1 1 0	<- not relevant
0 1 1 1	0 1 1 1	<- skip
0 1 0 1	0 1 0 1	<- skip
0 0 1 0	0 0 1 0	
0 0 1 1	0 0 1 1	
0 0 0 1	0 0 0 1	

Table 1: Search over i; 0:inactive, 1:active

Unfortunately, the test described in section 3.1 does not allow us to make the inference shown in Table 3.2. The reason is the presence of (16) in the test, which assures that a "past" exists such that the particular combination of active QP constraints is not only feasible but also optimal. It is possible that, say, two specific constraints cannot be active at the optimal solution of the QP, unless a third also becomes active. If however for a J_i, (15) has no solution over the constrained past us, Δus, and the y measurement values, then one can indeed infer that all J_is with the same plus more active constraints are not relevant either. As a result of this observation, we decided to subject every J_i that is tested during the search procedure, to the following three tests.

Test I: Check whether rank$[\hat{A}_i^T]$ =#rows$[\hat{A}_i^T]$. If not, discard this J_i and all with the same plus more active constraints.

Test II: Check whether (15) has a solution over the variables (subject to the u, Δu and y constraints on these variables): $u(\bar{k}-N), ..., u(\bar{k}-1), y(\bar{k})$. If not, discard this J_i and all with the same plus more active constraints.

Test III: Check whether the system of (15), (16) has a solution over the same variables as Test II. If not, discard this J_i only.

Tests II and III were discussed in the previous paragraph. Let us explain Test I. Clearly, before one can even consider Tests II or III, one has to make sure that (7) has indeed a solution, or equivalently that the LHS matrix is invertible. Hence one would to add a rank test on this matrix. However, it is sometimes possible to find a "past" such that the number of active constraints in (5) (equal to the number of rows of $\hat{A}_i^T$), is larger than than the available variables of the on-line QP $(u(\bar{k}), ..., u(\bar{k}+M-1))$. This is something that requires certain exact values for the "past" that one would not expect to take place during the actual implementation. Hence the rank test on the LHS of (7) is substituted with the stronger requirement described by Test I.

The three tests are applied in the above sequence. Test I involves a rank computation, but not always. When the number of rows of $\hat{A}_i^T$ is larger than the that of the columns, the J_i under consideration fails Test I. Tests II and III involve solving Phase I of a standard Linear Program, i.e., finding a feasible solution for an LP. When a J_i fails Test I or Test II, then one can skip other J_is as shown in Table 3.2. If it fails Test III, one can discard that J_i, but cannot skip other J_is.

4 Illustration

In this section we shall study the application of the search procedure of section 3 on the Shell Standard Control Problem (SSCP) (Prett and Garcia, 1988). Let us consider the top 2×2 part of the Heavy Oil Fractionator of the SSCP. This system has as outputs 1 and 2, the Top End Point and the Side End Point correspondingly. The inputs are the Top Draw and the Side Draw. The transfer function of this subsystem is

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{(4.05+2.11\epsilon_1)e^{-27s}}{50s+1} & \frac{(1.77+0.39\epsilon_2)e^{-28s}}{60s+1} \\ \frac{(5.39+3.29\epsilon_1)e^{-18s}}{50s+1} & \frac{(5.72+0.57\epsilon_2)e^{-14s}}{60s+1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{17}$$

where ϵ_1, ϵ_2 represent the model uncertainty and they can take any value between -1 and $+1$, 0 corresponding to the nominal model. A sampling time of $T = 6\ min$ is selected which results in lower and upper constraints of -0.3 and 0.3 for the changes in the inputs from one sampling point to the next. Lower and upper constraints of -0.5 and 0.5 exist for all the inputs and outputs.

In the objective function of (2) we initially select $P = 6$, $M = 2$, $B = D = 0$. The minimization is carried out subject to the above described hard constraints. The Constraint Window for the outputs includes future points 5-6 for the Top End Point and 3-4 for the Side End Point. Beginning the windows at earlier times may result in infeasibilities because of the longer time delays.

The on-line QP for this example has 12 two-sided constraints. The results of the application of the search over i for this system are shown in Table 4. Note that the tree structure shown in Table 3.2 is modified to allow for two-sided constraints (2: lower side active, 1: upper side active). This produces a smaller number of total J_is than if 24 one-sided constraints were used, since $3^{12} < 2^{24}$. Still, the total number is huge. The search procedure, however, reduces that number to only 68 J_is that need to be examined. One should note though, that this number could be 3-4 times as large, depending on how one defines the problem. In this case, we limited ourselves to starting from steady-state situations, and in Tests II and III, we looked for output disturbances that could result in the J_i under consideration. The bounds used for these disturbances, however, were quite large, equal to -1 and +1, which is twice the size of the output constraints in the on-line optimization.

Let us now apply the theorems of section 2 on the J_is that were judged as relevant to the stable operation of the controller. Note that for this example, if one changes all the 1s to 2s and all the 2s to 1s in the J_i descriptions, the feedback controller $C_{J_i}(z)$ remains the same,

Total possible J_is	=	**531,441**
Enumerated J_is	=	**4,985**
J_is that failed the 1st criterion	=	**2,300**
J_is that failed the 2nd criterion	=	**585**
J_is that failed the 3rd criterion	=	**2,032**
Relevant J_is	=	**68**

Table 2: Search results for the 2×2 example

and so there is no need to check symmetric cases. The first 12 entries in Table 3 describe the status of the 12 constraints for the J_is under examination. The second to last column lists the largest magnitude of the closed-loop poles for the corresponding $C_{J_i}(z)$, when the plant is the same as the model. Hence for nominal stability of the corresponding controller, we need that value to be inside the Unit Circle. This is not the cacse for quite a few of these J_is. However, simulations for no model-plant mismatch (Zafiriou, 1989b), showed no instability. The explanation is that the closed-loop nominal operator is not a contraction, but it is stable nevertheless. This is something that is expected and which we have observed before. One would expect however, the simulations to show some bad performance at regions where the corresponding J_is take hold. This does not happen and the explanation lies in the nature of the J_is that correspond to nominal closed-loop poles outside the Unit Circle. All of these have at least one of the constraints on $\Delta u_1(\bar{k})$ and $\Delta u_2(\bar{k})$ active at the optimum. Since these values are actually implemented (see section 1), it is essentially impossible to have such J_is occur again and again during the perfect model simulations, because of the constraints on u (at -0.5 and +0.5; the Δu constraints are at -0.3 and +0.3). Hence, the second to last column in Table 3 should raise no concerns if the model was accurate. If however, model-plant mismatch is present, the J_is that correspond to active Δu constraints might occur often enough to cause instability.

The last column in Table 3 gives the Structured Singular Value (Doyle, 1982) for each $C_{J_i}(z)$. Note that the fact that the uncertain parameters ϵ_1 and ϵ_2 are real-valued had been taken into account. The value of μ is equal to the inverse of the smallest value of $max\{|\epsilon_1|, |\epsilon_2|\}$ that is needed to have the Robust Stability condition violated for the corresponding $C_{J_i}(z)$. By looking at these values, we see that the smallest needed ϵ has magnitude of approximately 1.2. Hence one would not expect any problems when the ϵs are within the -1 and +1 bounds. If they can take values up to 1.2 though, the controller will become unstable. This is exactly the behavior that was observed in the simulations of Zafiriou (1989b).

Table 4 repeats the information of Table 3 in the case where a penalty has been introduced

No.	$\Delta u_1(k)$	$\Delta u_2(k)$	$\Delta u_1(k+1)$	$\Delta u_2(k+1)$	$u_1(k)$	$u_2(k)$	$u_1(k+1)$	$u_2(k+1)$	$y_2(k+3)$	$y_2(k+4)$	$y_1(k+5)$	$y_1(k+6)$	max|PQ|*	μ
(1)	0	0	0	0	0	0	0	0	0	0	0	0	0.9068	0.8002
(2)	2	0	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(3)	2	1	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(4)	2	1	1	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(5)	2	1	1	2	0	0	0	0	0	0	0	0	1.0000	0.0000
(6)	2	1	1	0	0	0	0	1	0	0	0	0	1.0000	0.0000
(7)	2	1	0	2	0	0	0	0	0	0	0	0	1.0000	0.0000
(8)	2	1	0	0	0	0	2	0	0	0	0	0	1.0000	0.0000
(9)	2	1	0	0	0	0	2	1	0	0	0	0	1.0000	0.0000
(10)	2	1	0	0	0	0	0	1	0	0	0	0	1.0000	0.0000
(11)	2	1	0	0	0	0	0	1	0	0	0	1	1.0000	0.0000
(12)	2	2	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(13)	2	2	1	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(14)	2	2	1	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(15)	2	2	0	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(16)	2	0	1	0	0	0	0	0	0	0	0	0	1.0000	0.0413
(17)	2	0	1	1	0	0	0	0	0	0	0	0	1.0000	0.0087
(18)	2	0	1	1	0	0	0	0	0	0	1	0	1.9027	0.0000
(19)	2	0	1	2	0	0	0	0	0	0	0	0	1.0000	0.0087
(20)	2	0	1	0	0	0	0	1	0	0	1	0	1.9027	0.0000
(21)	2	0	0	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(22)	2	0	0	1	0	0	2	0	0	0	0	0	1.0000	0.0087
(23)	2	0	0	1	0	0	2	1	0	0	0	0	1.0000	0.0000
(24)	2	0	0	1	0	0	2	0	0	0	1	0	1.9027	0.0000
(25)	2	0	0	1	0	0	0	1	0	0	0	0	1.0000	0.0000
(26)	2	0	0	1	0	0	0	0	0	0	1	0	1.9027	0.0000
(27)	2	0	0	2	0	0	0	0	0	0	0	0	1.0000	0.0000
(28)	2	0	0	0	0	0	2	0	0	0	0	0	1.0000	0.0413
(29)	2	0	0	0	0	0	2	1	0	0	0	0	1.0000	0.0418
(30)	2	0	0	0	0	0	2	1	0	0	1	0	1.9027	0.0000
(31)	2	0	0	0	0	0	0	1	0	0	0	0	1.0000	0.0476
(32)	2	0	0	0	0	0	0	1	0	0	1	0	1.9027	0.0000
(33)	2	0	0	0	0	0	0	1	0	0	1	1	1.9027	0.0000
(34)	0	2	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(35)	0	2	1	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(36)	0	2	1	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(37)	0	2	1	0	0	0	1	0	0	0	0	0	1.0000	0.0000
(38)	0	2	1	0	0	0	1	2	0	0	0	0	1.0000	0.0000
(39)	0	2	1	0	0	0	0	2	0	0	0	0	1.0000	0.0000
(40)	0	2	2	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(41)	0	2	2	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(42)	0	2	2	0	0	0	0	2	0	0	0	0	1.0000	0.0000
(43)	0	2	2	0	0	0	0	2	0	0	2	0	1.0000	0.0000
(44)	0	2	2	0	0	0	0	2	0	0	0	2	1.0000	0.0000
(45)	0	2	0	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(46)	0	2	0	0	0	0	1	0	0	0	0	0	1.0000	0.0000
(47)	0	2	0	0	0	0	1	2	0	0	0	0	1.0000	0.0000
(48)	0	2	0	0	0	0	0	2	0	0	0	0	1.0000	0.0000
(49)	0	2	0	0	0	0	0	2	0	1	0	2	1.0000	0.0000
(50)	0	2	0	0	0	0	0	2	0	0	2	0	1.0000	0.0000
(51)	0	2	0	0	0	0	0	2	0	0	2	2	1.0000	0.0000
(52)	0	2	0	0	0	0	0	2	0	0	0	2	1.0000	0.0000
(53)	0	0	2	0	0	0	0	0	0	0	0	0	0.9075	0.6786
(54)	0	0	2	1	0	0	0	0	0	0	0	0	0.9067	0.7728
(55)	0	0	2	1	0	0	2	0	0	0	0	0	0.9038	0.0087
(56)	0	0	2	1	0	0	2	1	0	0	0	0	0.0000	0.0000
(57)	0	0	2	1	0	0	0	1	0	0	0	0	0.8863	0.0000
(58)	0	0	2	2	0	0	0	0	0	0	0	0	0.9067	0.7728
(59)	0	0	2	0	0	0	2	0	0	0	0	0	0.9058	0.0413
(60)	0	0	2	0	0	0	2	1	0	0	0	0	0.9046	0.0418
(61)	0	0	2	0	0	0	0	1	0	0	0	0	0.8996	0.3382
(62)	0	0	0	2	0	0	0	0	0	0	0	0	0.9055	0.6260
(63)	0	0	0	2	0	0	1	0	0	0	0	0	0.9012	0.3891
(64)	0	0	0	2	0	0	1	2	0	0	0	0	0.8868	0.0000
(65)	0	0	0	2	0	0	0	2	0	0	0	0	0.8869	0.0000
(66)	0	0	0	0	0	0	2	0	0	0	0	0	0.9075	0.8570
(67)	0	0	0	0	0	0	2	1	0	0	0	0	0.9071	0.7189
(68)	0	0	0	0	0	0	0	2	0	0	0	0	0.9064	0.6918

Table 3: B=0

on the inputs in the on-line optimization ($B = 0.2I$ in (2)). The values of μ are reduced and as result, ϵs of magnitude 1.2 do not cause a violation of the robust stability conditions. Note that the location of the nominal closed-loop poles is not significantly affected for the J_is corresponding to active Δu constraints. This however is not expected to cause any problems because μ is not large enough for the other J_is. Again, this is exactly the behavior previously observed in simulations (Zafiriou, 1989b).

References

[] J. C. Doyle, "Analysis of Feedback Systems with Structured Uncertainty", *I.E.E. Proc.,* Part D, **129**, pp. 242-250, 1982.

[] C. G. Economou, *An Operator Theory Approach to Nonlinear Controller Design*, Ph.D. Thesis, California Insitute of Technology, 1985.

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

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

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

[] M. Morari and E. Zafiriou, *Robust Process Control*, Prentice–Hall, Englewood Cliffs, NJ, 1989.

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

[] N. L. Ricker, T. Subrahmanian and T. Sim, "Case Studies of Model-Predictive Control in Pulp and Paper Production", *Proc. IFAC Workshop on Model Based Process Control,* Pergammon Press, Oxford, 1989.

[] E. Zafiriou, "Robust Model Predictive Control of Processes with Hard Constraints", *accepted for publication in Comp. and Chem. Eng.,* 1989a.

[] E. Zafiriou, "Robustness of Model Predictive Control Algorithms for Systems with Hard Constraints," *Amer. Control Conf.,* Pittsburgh, PA, pp. 2500-2505 of the Proceedings, June 1989b.

No.	$\Delta u_1(k)$	$\Delta u_2(k)$	$\Delta u_1(k+1)$	$\Delta u_2(k+1)$	$u_1(k)$	$u_2(k)$	$u_1(k+1)$	$u_2(k+1)$	$y_2(k+3)$	$y_2(k+4)$	$y_1(k+5)$	$y_1(k+6)$	max|PQ|*	μ
(1)	0	0	0	0	0	0	0	0	0	0	0	0	0.9054	0.5377
(2)	2	0	0	0	0	0	0	0	0	0	0	0	1.0000	0.0267
(3)	2	1	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(4)	2	1	1	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(5)	2	1	1	2	0	0	0	0	0	0	0	0	1.0000	0.0000
(6)	2	1	1	0	0	0	0	1	0	0	0	0	1.0000	0.0000
(7)	2	1	0	2	0	0	0	0	0	0	0	0	1.0000	0.0000
(8)	2	1	0	0	0	0	2	0	0	0	0	0	1.0000	0.0000
(9)	2	1	0	0	0	0	2	1	0	0	0	0	1.0000	0.0000
(10)	2	1	0	0	0	0	0	1	0	0	0	0	1.0000	0.0000
(11)	2	1	0	0	0	0	0	1	0	0	0	1	1.0000	0.0000
(12)	2	2	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(13)	2	2	1	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(14)	2	2	1	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(15)	2	2	0	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(16)	2	0	1	0	0	0	0	0	0	0	0	0	1.0000	0.0318
(17)	2	0	1	1	0	0	0	0	0	0	0	0	1.0000	0.0085
(18)	2	0	1	1	0	0	0	0	0	0	1	0	1.9027	0.0000
(19)	2	0	1	2	0	0	0	0	0	0	0	0	1.0000	0.0085
(20)	2	0	1	0	0	0	0	1	0	0	1	0	1.9027	0.0000
(21)	2	0	0	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(22)	2	0	0	1	0	0	2	0	0	0	0	0	1.0000	0.0085
(23)	2	0	0	1	0	0	2	1	0	0	0	0	1.0000	0.0000
(24)	2	0	0	1	0	0	2	0	0	0	1	0	1.9027	0.0000
(25)	2	0	0	1	0	0	0	1	0	0	0	0	1.0000	0.0000
(26)	2	0	0	1	0	0	0	0	0	0	1	0	1.9027	0.0000
(27)	2	0	0	2	0	0	0	0	0	0	0	0	1.0000	0.0000
(28)	2	0	0	0	0	0	2	0	0	0	0	0	1.0000	0.0318
(29)	2	0	0	0	0	0	2	1	0	0	0	0	1.0000	0.0380
(30)	2	0	0	0	0	0	2	1	0	0	1	0	1.9027	0.0000
(31)	2	0	0	0	0	0	0	1	0	0	0	0	1.0000	0.0373
(32)	2	0	0	0	0	0	0	1	0	0	1	0	1.9027	0.0000
(33)	2	0	0	0	0	0	0	1	0	0	1	1	1.9027	0.0000
(34)	0	2	0	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(35)	0	2	1	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(36)	0	2	1	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(37)	0	2	1	0	0	0	1	0	0	0	0	0	1.0000	0.0000
(38)	0	2	1	0	0	0	1	2	0	0	0	0	1.0000	0.0000
(39)	0	2	1	0	0	0	0	2	0	0	0	0	1.0000	0.0000
(40)	0	2	2	0	0	0	0	0	0	0	0	0	1.0000	0.0000
(41)	0	2	2	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(42)	0	2	2	0	0	0	0	2	0	0	0	0	1.0000	0.0000
(43)	0	2	2	0	0	0	0	2	0	0	2	0	1.0000	0.0000
(44)	0	2	2	0	0	0	0	2	0	0	0	2	1.0000	0.0000
(45)	0	2	0	1	0	0	0	0	0	0	0	0	1.0000	0.0000
(46)	0	2	0	0	0	0	1	0	0	0	0	0	1.0000	0.0000
(47)	0	2	0	0	0	0	1	2	0	0	0	0	1.0000	0.0000
(48)	0	2	0	0	0	0	0	2	0	0	0	0	1.0000	0.0000
(49)	0	2	0	0	0	0	0	2	0	1	0	2	1.0000	0.0000
(50)	0	2	0	0	0	0	0	2	0	0	2	0	1.0000	0.0000
(51)	0	2	0	0	0	0	0	2	0	0	2	2	1.0000	0.0000
(52)	0	2	0	0	0	0	0	2	0	0	0	2	1.0000	0.0000
(53)	0	0	2	0	0	0	0	0	0	0	0	0	0.9055	0.5707
(54)	0	0	2	1	0	0	0	0	0	0	0	0	0.9054	0.5897
(55)	0	0	2	1	0	0	2	0	0	0	0	0	0.9038	0.0085
(56)	0	0	2	1	0	0	2	1	0	0	0	0	0.0000	0.0000
(57)	0	0	2	1	0	0	0	1	0	0	0	0	0.8863	0.0000
(58)	0	0	2	2	0	0	0	0	0	0	0	0	0.9054	0.5897
(59)	0	0	2	0	0	0	2	0	0	0	0	0	0.9051	0.0318
(60)	0	0	2	0	0	0	2	1	0	0	0	0	0.9046	0.0380
(61)	0	0	2	0	0	0	0	1	0	0	0	0	0.8969	0.2896
(62)	0	0	0	2	0	0	0	0	0	0	0	0	0.9048	0.4355
(63)	0	0	0	2	0	0	1	0	0	0	0	0	0.9006	0.8342
(64)	0	0	0	2	0	0	1	2	0	0	0	0	0.8868	0.0000
(65)	0	0	0	2	0	0	0	2	0	0	0	0	0.8869	0.0000
(66)	0	0	0	0	0	0	2	0	0	0	0	0	0.9058	0.5465
(67)	0	0	0	0	0	0	2	1	0	0	0	0	0.9059	0.5473
(68)	0	0	0	0	0	0	0	2	0	0	0	0	0.9053	0.5333

Table 4: B=0.2 I