MODEL PREDICTIVE CONTROL WITH GENERALIZED INPUT PARAMETRIZATION

Edwin T. van Donkelaar, Okko H. Bosgra, Paul M.J. Van den Hof

Mechanical Engineerging Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 DG Delft, The Netherlands
Fax : +31-15-2784717 and e-mail : e.t.vandonkelaar@wbmt.tudelft.nl

Keywords : model predictive control, parametrization.

Abstract

In this article it is investigated how, alternative to e.g. the standard pulse or blocking mechanisms, other input parametrizations can be used in model predictive control to improve the trade-off between performance and complexity. An efficient parametrization is obtained using the observation that the class of all solutions to a finite or infinite horizon LQ control problem can be parametrized with a number of free parameters that is equal to the model order, without loss of performance. The inifinite horizon controller with this parametriztion is shown to provide a stable closed-loop system, also if constraints are active. The complexity of the parametrization can be systematically reduced using standard reduction techniques, such as e.g. LQG-balanced reduction, which provide an indication of the performance loss. Constrained stability of the closed-loop system is preserved with this reduction approach. The proposed algorithms are illustrated with simulation examples.

1 Introduction

For the control of systems with hard input and state constraints basically one technology is available: model predictive control [7]. The computational requirement for this control strategy is high due to the on-line optimization. This has restricted the use of this technology to relatively slow sampled systems with limited dynamic performance specifications such as the ones encountered in the petro-chemical and chemical industry. With the current generation model predictive control it is difficult if not impossible to achieve high performance constrained control of fast dynamical systems such as fast and accurate robots and electro-mechanical positioning systems such as wafer steppers or consumer electronics. It is also difficult to design high performance model predictive controllers for large-scale systems which have a relatively small sampling time, encountered in e.g. steel and aluminium production processes. To apply this technology to these systems a large increase in efficiency is required.

A way to improve efficiency in MPC is to decrease the number of degrees of freedom in the optimization problem to obtain a lower order optimization problem while attempting to keep the loss in performance as small as possible. Examples are the reduction of the free variables in the input parametrization by introduction of a control horizon, e.g. used in [2] in a finite horizon MPC and in [8] in an infinite horizon MPC, the use of blocking ([10],[5]), the use of functions in predictive functional control (PFC, [9]) and the more recent work of [11] where the input trajectory is parameterized in terms of only one variable.

In this article an approach is discussed to determine an input parametrization for model predictive controllers such that only a small number of parameters is consumed to obtain a closed-loop system that is equivalent to finite or infinite horizon LQ optimal control in the unconstrained case. The approach is able to deal with constaints but in a suboptimal way. The input trajectory is parametrized in terms of an expansion in basis functions which can be calculated a priori. The number of applied basis functions, and hence the optimization complexity, can be reduced systematically with a standard model reduction tool, namely LQG balanced reduction. This reduction method provides the user with valuable information about how much the number of basis functions can be reduced, such that the decrease in unconstrained performance level is small. This provides a tool to make the trade-off between unconstrained performance and complexity in a more systematic way.

In section 2 the problem is specified in mathematical terms. In section 3 an approach is presented to obtain an input parametrization which consumes the smallest number of parameters to obtain finite and infinite LQ optimal control in the unconstrained case. In section 4 this procedure is applied to an infinite horizon LQ criterion with constraints. The proposed procedure has a number of free parameters in the optimization that is equal to the model order. In section 5 it is shown how the number of free variables can be further reduced systematically. In this reduction the loss of performance can be assessed. In section

6 the proposed model predictive control algorithm is applied to a simulation example and the reduction method is illustrated on a multivariable nonlinear simulation model of a complex petro-chemical process. Finally, section 7 concludes the paper.

2 Model predictive control

Linear model predictive control or receding horizon control provides a solution to the problem of constrained control of systems. Let the system be given by the state-space description

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), x(0) = x_0 \\ y(t) &= Cx(t) \end{aligned}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^{n_u}$ the input vector at time t. The matrix A has eigenvalues strictly inside the unit disc, $|\lambda\{A\}| < 1$ and the pair (A, B) is controllable and (C, A) is observable. The aim is to control this system while satisfying constraints on the input and state variables

$$\begin{aligned} K_u u(t) &< k_u \text{ for all } t \\ K_x x(t, x_0) &< k_x \text{ for all } t \end{aligned} \quad (1)$$

where $K_u \in \mathbb{R}^{n_u \times n_u}, K_u \in \mathbb{R}^{n \times n}$ are the matrices that specify the input and state constraints. These matrices can have a different row dimension than specified and can be time-varying but are specified in this way for convenience of notation.
The most commonly used cost function in model predictive control is the quadratic cost function given by

$$J(u(\cdot)) = \sum_{t=0}^{P-1} \{x^T(t)Q_1x(t)+u^T(t)Q_2u(t)\}+x^T(P)Q_0x(P). \quad (2)$$

with the weighting matrices $Q_1, Q_2 \geq 0$ and the pair $\{Q_1, A\}$ is detectable. The optimal input trajectory can be found by solving an optimization problem in terms of the vector $U^T = [u^T(0)\ u^T(1) \ldots u^T(P-1)]$, which is given by the quadratic programming problem

$$\min_U \{U^T HU + 2x_0^T g^T U\} \quad (3)$$

subject to $KU \leq k$. The variables in these expressions are defined in Appendix A. At each time instant the value x_0 in the quadratic program is updated with the measured value of the state (full information case) or a prediction of the current state (partial information case). In this way at each time instant an updated quadratic program is specified where possibly another set of constraints is active.
The input trajectory U can be parametrized in various ways. A general way to describe the parametrization of U is such that the class of obtainable input trajectories is a subspace $\mathcal{U}_r \subset \mathbb{R}^{Pn_u}$ which has a (considerably) lower dimension than the full space. This can be described by

$$U(\theta) = \phi\theta \text{ with } \phi \in \mathbb{R}^{Pn_u \times n_\theta}, \theta \in \mathbb{R}^{n_\theta} \quad (4)$$

where θ is the free parameter and ϕ is a user-chosen matrix of which the columns form a basis for the space of all input trajectories that can be achieved. The input parametrization above includes the conventional such as input parametrization with a control horizon and parametrization using blocking. With the input space parametrized by the column span of ϕ, the optimization problem that has to be solved is, similar to conventional MPC: a quadratic programming problem. This is given by $\theta^* = arg\min_\theta J(\theta)$ with

$$J(\theta) = \theta^T \phi^T H \phi\theta + 2x_0^T g^T \phi\theta \quad (5)$$

subject to the constraints $K\phi\theta \leq k$ which are linear in the free parameter θ.
The matrix ϕ is clearly a design variable that is able to influence the controlled behaviour in a crucial way. The main question is, how this freedom can be used in a structured way to obtain a clear trade-off between closed-loop performance and optimization complexity.

3 Finding a suitable input parametrization

It is difficult to find a fixed low dimensional subspace $\mathcal{U}_r$ that contains the optimal input trajectories for every possible active set of constraints or good approximations of these optimal trajectories. Therefore we restrict ourselves to find a low dimensional subspace that contains all unconstrained input trajectories and provides some freedom to deal with constraints.
From standard LQ theory [1] it is known that the finite horizon LQ problem with a free parametrization is generated by a time-varying state feedback which can be calculated a priori. The only thing that cannot be computed a priori is the state with which the optimization problem is initialized. This simple observation provides a low dimensional subset that contains all unconstrained optimal profiles. This is reflected in the following lemma for the linear quadratic cost function from (3).

Lemma 3.1 *Consider the quadratic cost function given by* (3) *with finite horizon P and let there be no constraints. Then, the subspace*

$$\mathcal{U}_r = im\{H^{-1}g\}$$

with $H \in \mathbb{R}^{Pn_u \times Pn_u}, g \in \mathbb{R}^{Pn_u \times n}$ given in (3) *contains all unconstrained optimal solutions.*

Proof: The lemma follows from the well known result that the unconstrained solution to the quadratic programming problem (3) is given by $U(x_0) = H^{-1}gx_0$. ∎

Apparently if one wants to find the unconstrained optimum for all possible initialisations x_0, a search over an n-dimensional subspace is sufficient instead of a Pn_u-dimensional one.
Theorem 3.1 holds for a finite horizon criterion but a similar result also holds for the infinite horizon case.

Lemma 3.2 *Consider the quadratic cost function be given by* (3) *with infinite horizon* $P = \infty$ *and let there be no constraints. Then, the subspace*

$$\mathcal{U}_r = im\{\begin{bmatrix} F \\ F(A-BF) \\ F(A-BF)^2 \\ \vdots \end{bmatrix}\} \subset l_2^n[0,\infty), \quad (6)$$

with $\{A, B\}$ *state space matrices given in* (1) *and* F *the LQ-optimal state feedback given by* $F = (B^T XB + Q_2)^{-1}B^T XA$ *with* X *the unique nonnegative definite solution of the Algebraic Riccati Equation*

$$X = A^T[X + XB(B^T XB + Q_2)^{-1}B^T X]A + Q_1,$$

contains all unconstrained optimal solutions.

Proof: The unconstrained solution to the problem (3) with infinite horizons is the LQ optimal control profile given by

$$u(t, x_0) = F(A - BF)^t x_0, \;\; t = 0, 1, 2, \ldots$$

which directly shows the result. ■

The lemma above indicates that an efficient input parametrization for infinite horizon model predictive control is generated by a dynamical system $\{F, A - BF\}$. With this parametrization the infinite dimensional optimization problem is reduced to a finite dimensional optimization problem with a number of free variables that is equal to the model order.

4 Model predictive control with input parametrization

In this section the parametrization of lemma 3.1 is applied to obtain an alternative parametrization of model predictive control. This controller and some of its properties is given in the following proposition.

Proposition 4.1 *Let the receding horizon controller cost function be given by* (2) *with* $P = \infty$ *and let* x_0 *be either the measured state vector (full information case) or a prediction thereof (partial information case). Let the input over the infinite horizon be parametrized as* $u(t, \theta) = F(A - BF)^t\theta$ *and let*
$\tilde{A} = \begin{bmatrix} A-BF & 0 \\ BF & A \end{bmatrix}$ *and* $\tilde{C} = \begin{bmatrix} \sqrt{R}F & 0 \\ 0 & \sqrt{Q}C_z \end{bmatrix}$ *and let* N_c *be the constraint horizon be such that after this time instant no constraints are active.*
Then

1. *the optimal control input* $u^*(0)$ *is given by* $u^*(0) = F\theta^*$ *with* θ^* *the solution to the finite dimensional quadratic programming problem*

$$\min_{\theta\in\mathbb{R}^n} \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} Y \begin{bmatrix} \theta \\ x_0 \end{bmatrix} \quad (7)$$

subject to: $K_u F(A - BF)^t\theta < k_u$

$$\begin{bmatrix} 0 & K_x \end{bmatrix} \tilde{A}^t \begin{bmatrix} \theta \\ x_0 \end{bmatrix} < k_x, t = 0, 1, \ldots, N_c$$

where Y *is the solution of the Lyapunov equation*

$$\tilde{A}^T Y \tilde{A} + \tilde{C}^T \tilde{C} = Y$$

2. *if no constraints are active this controller is equivalent to LQ control with state feedback* F.

Proof: The input and state trajectories over the infinite horizon are given by

$$\begin{bmatrix} u(t,\theta) \\ x(t,\theta,x_0) \end{bmatrix} = \tilde{A}^t \begin{bmatrix} \theta \\ x_0 \end{bmatrix} \quad (8)$$

Substituting this in the cost function yields

$$J(\theta, x_0) = \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} \left(\sum_{t=0}^{\infty} \tilde{A}^{Tt}\tilde{C}^T\tilde{C}\tilde{A}t\right) \begin{bmatrix} \theta \\ x_0 \end{bmatrix}$$

The matrix in this expression can be calculated with the Lyapunov equation in statement 1. Statement 2 can be proven by the fact that the unconstrained optimal solution is given by $\theta = -x_0$. ■

The closed-loop behaviour of the resulting controller is equivalent to LQ optimal state feedback control if no contraints are active. This is obtained with only n degrees of freedom in the optimization problem, where n is the model order. The optimization problem can be built up by solving one Riccati equation for the solution of the LQ control problem and a Lyapunov equation to specify the cost function. This can be done fast because good software tools are available for solving Riccati and Lyapunov equations, also for large scale problems. Therefore, the procedure is flexible for on-line changes in the internal model and the controller cost function which is a favouralbe properety for e.g. constrained control of nonlinear systems with switching linear models such as nonlinear quadratic dynamic matrix control (NLQDMC, [3]). Tuning of the algorithm is equal to standard LQG control design. The only additional tuning parameter is the constraint horizon N_c. This parameter is no tuning variable for nominal unconstrained performance as it has no influence on the closed-loop performance. This makes tuning of this controller more simple than predictive controllers where both the prediction horizon P and the control horizon M influence the unconstrained performance. The parameter N_c can also be chosen automatically as discussed in [8].

The controller presented in this section provides a stable closed-loop, also if constraints are active. This will be proven along similar lines as the stability proofs of [8] for the full information case and [13] for the partial information case.

Proposition 4.2 *The predictive control strategy given in theorem 4.1 is globally asymptotically stable if and only if the optimization problem* (7) *is feasible*

Proof: First consider the full information case. Global time index is denoted with t and the local time index within the optimization is denoted with k. Let the input trajectory $u_t^*(k) = -F(A-BF)^k\theta_t^*$ be a feasible but possibly not optimal solution at time t. Let the corresponding cost be given by $J(t)$. The first sample of this trajectory is applied as current input $u(t) = F\theta^*$. This yields a state $x(t)$ which is equal to the predicted state if no disturbances are present and the model and plant are equal. Then a feasible trajectory for $t+1$ is given by $u_{t+1}^*(k) = -F(A-BF)^k\theta_{t+1}^*$ with $\theta_{t+1}^* = (A-BF)\theta_t^*$ as this is equivalent with the previous trajectory without the first sample. Denote the corresponding cost function with $J(t+1)$. This performance cost level need not be optimal therefore it holds that

$$J(t+1) \leq J(t) - x^T(t)Q_1x(t) - u^T(t)Q_2u(t)$$

Because $Q_1, Q_2 \geq 0$ the sequence $J(t)$ is non increasing. It is bounded from below by zero and therefore $J(t)$ converges to zero, hence $x(t), u(t)$ also converge to zero. Therefore the nonlinear state feedback is stabilizing. Due to the separation principle this stabilizing state feedback combined with a stable observer yields a stabilizing dynamic output feedback [13]. ■

Closed loop stability can only be lost if the optimization problem is infeasible which can only be lost if hard state (output) constraints are used. In the literature it is described how the problem of feasibility can be avoided.

5 Systematic reduction of the complexity

In this section it is described how the complexity of the model predictive control algorithm discussed in the previous section can be reduced while keeping track of the performance loss. This is done by choosing the parametrization of input in terms of a linear combination of profiles that have the largest contribution to the cost function. Because the parametrization is generated by a linear model it is possible to base the system-based input parametrization on a reduced order model i.e.

$$u(t,\theta) = F_r(A_r - B_rF_r)^t\theta \tag{9}$$

where $\{A_r, B_r\}$ are state-space matrices for the reduced order system and F_r is the LQ-optimal state feedback for this reduced order model.

The model reduction algorithm that is applied should be such that the reduced order basis functions have the largest contribution to the cost function. For LQG control this can be done e.g. with LQG-balanced reduction [4]. With this reduction technique first a similarity transformation is applied that forces the solution of the control discrete algebraic Riccati equation (CDARE) and the filter discrete algebraic Riccati equation (FDARE),

$$X = A^TXA - A^TXB(B^TXB+I)^{-1}B^TXA + C^TC \tag{10}$$
$$Y = AYA^T - AYC^T(CXC^T+I)^{-1}CYA^T + B^TB. \tag{11}$$

respectively, to be equal. In [4] it is proven that the similarity transformation $\{\bar{A}, \bar{B}, \bar{C}\} = \{TAT^{-1}, TB, CT^{-1}\}$ renders the solution of the riccati equations equal to Σ if the tranformation matrix is $T^{-1} = R^TU\Sigma^{-\frac{1}{2}}$. Here R is a Cholesky factor $Y = R^*R$ and U comes from the eigenvalue decomposition $RXR^* = U\Sigma^2U^*$ where $U^*U = I$. The states that have a "large" corresponding value on the diagonal of Σ are both "difficult" to filter and are essential states to control and must certainly be accounted for in a reduced order controller. The diagonal elements are invariants for linear systems and can be used to decide on the reduction order in a similar manner as Hankel singular values are used in balanced reduction. Let the reduction order be given by n_r then the resulting reduced order model can simply be obtained by

$$A_r = [I_{n_r}\ 0]\bar{A}[I_{n_r}\ 0]^T, B_r = [I_{n_r}\ 0]\bar{B}, C_r = \bar{C}[I_{n_r}\ 0]^T.$$

The reduced order input parametrization can be constructed with (9). The input parametrization based on the reduced order model is again generated by a stable dynamic system because the LQ-optimal state feedback is guaranteed to be stabilizing. Due to this fact and stability of the system it can be proven that for any reduction order the receding horizon controller is stable also in the constrained case. This is given in the next corollary.

Corollary 5.1 *The predictive control strategy given in proposition 4.1 with input parametrization generated by a stable dynamic system* $\{C_p, A_p\}$ *following*

$$u(t,\theta) = C_pA_p^t\theta$$

is globally asymptotically stable if and only if the optimization problem (7) *is feasible.*

Proof: Along identical lines as the proof of proposition (4.1) only with $\{C_p, A_p\}$ instead of $\{F, A-BF\}$. ■

Note that this theorem implies that constrained closed-loop stability with the proposed controller is preserved if the input is parametrized with any stable system. However, the model that is applied for the prediction is still equal to the plant.

6 Simulation examples

In this section two simulation examples are given to demonstrate the properties of the proposed approach, denoted with MPC_{ip}, compared to model predictive control, denoted with MPC.
The first system that is considered is a highly oscillatory nonminimum-phase system given by the transfer function

$$G(z) = \frac{-5.7980z^3 + 19.5128z^2 - 21.6452z + 7.9547}{z^4 - 3.0228z^3 3.8630z^2 - 2.6426z + 0.8084}$$

The open-loop step response is given in figure 1.

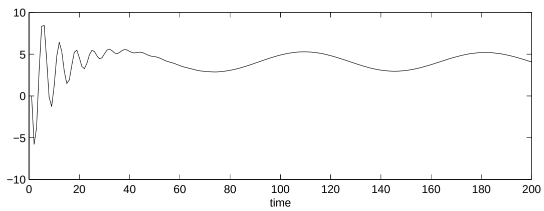

Fig. 1: *Step response of the plant*

If conventional MPC is applied to this system it is necessary to take a prediction horizon that is long enough to incorporate at least one full period, i.e. P=100. Also the choice of the control horizon is critical for this system. A control horizon which is equal to the prediction horizon gives good performance. Reduction of the control horizon upto M=75 is possible without considerable loss of performance, further reduction gives bad performance due to the slow oscillation.
Due to the long prediction and control horizon the computational burden is large for MPC. With the approach presented in this article, MPC_{ip}, the number of degrees of freedom is equal to the model order, i.e. n=4. This yields a controlled performance, depicted in figure 6, that is practically identical to the fully parametrized controller only at a much lower computational cost. Also if constraints are active MPC_{ip} performs better than conventional MPC with the same number of degrees of freedom as can be seen from figure 6. It performs slightly less than the fully parametrized MPC but at a much lower computational cost. To give an indication, on a Pentium 233 MHz computer the simulation of 200 time samples cost 511.83 seconds for fully parametrized conventional MPC and only 10.885 seconds for MPC_{ip}.
The second system that is considered is a 4 input 4 output subsystem of the nonlinear simulation model of a fluidized bed catalytic cracking unit (FCCU) given in [6]. The inputs are F_3, F_4, ΔP and V_{lift}. The outputs are T_{reg}, T_r, C_{O_2} and l_{sp}. The linear model on which the model predictive controllers are based, is of 8th order. A detailed flowsheet of the process with an explanation of the variables can be found in [6]. The system has large interaction, a combination of fast and slow dynamical phenomena and it is nonlinear. The linear model on which the model predictive controllers are based, is of 8th order.

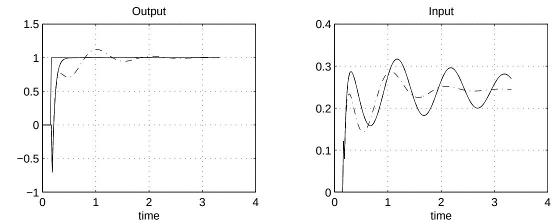

Fig 2. *Closed-loop step response conventional MPC with M=P=100 (solid) and P=100,M=4 (dash dotted) and MPC_{ip} (dashed) with 4 degrees of freedom.*

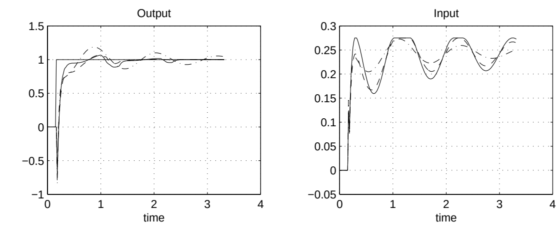

Fig. 3. *Constrained closed-loop step response conventional MPC with M=P=100 (solid) and P=100,M=4 (dash dotted) and MPC_{ip} (dashed) with 4 degrees of freedom.*

To compare MPC and MPC_{ip}, the tuning is chosen such that the complexity of the optimization problems is equal. The finite horizon MPC has tuning variables $M = 2$ for all four inputs and $P = 100$ to obtain enough preview. MPC_{ip} has tuning $N_c = 100$ and also 8 free variables in the optimization. Both controllers are tuned with tuning parameters $Q = I, R = I$.

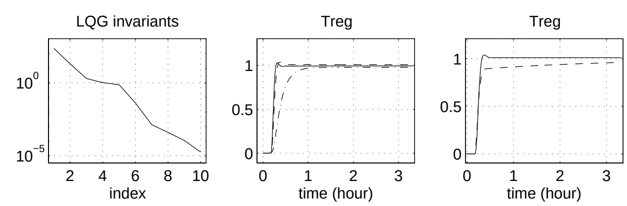

Fig. 4. *Left: LQG invariants of the model. Middle: the controlled output T_{reg} on a step on the reference signal of $2°C$; unreduced MPC_{ip} of order 10 (solid), reduced to order 5 (dashed) and reduced to order 4 (dash dotted). Right: MPC with the input parametrization proposed in this paper, reduced to 5 degrees of freedom (solid) and MPC with conventional parametrization with 5 degrees of freedom (dashed).*

The procedure of section 5 is applied to assess whether the complexity can be reduced without significant loss of performance. For this purpose the LQG invariants of the model are determined. These are plotted in figure 6.
From this figure it can be seen that upto order 5 the LQG invariants are larger or equal than one and are significantly smaller for higher orders. Therefore the order can be reduced to 5 without considerable loss in unconstrained performance. The closed-loop step response for

the full order and reduced order is also given in figure 6. This figure indicates that the loss in performance is indeed small. Further reduction to order 4 shows a large loss in performance which is in accordance with the LQG invariants. Hence, these invariants give a good indication of the smallest number of free variables that is needed to obtain good unconstrained performance.

7 Conclusions

In this article a systematic approach is described to choose the degrees of freedom in the parametrization of the input trajectory in finite and infinite horizon model predictive control algorithms. The input is parametrized as a linear combination of basis functions which are chosen such that only n free variables is needed, where n is the model order, to obtain closed-loop control which is equivalent to finite or infinite horizon LQ optimal control if no constraints are active. It is proven that the infinite receding horizon controller with the proposed parametrization provides closed-loop stability, also if constraints are active. For an infinite horizon LQ cost function a systematic way is discussed to further reduce the number of free variables with standard model reduction tools such as LQG-balanced reduction. This reduction method provides an indication of the performance loss in terms of the LQG invariants of the system which helps to decide on a suitable reduction order. The proposed approach is illustrated on simulation examples. In the present algorithm the choice of the basis is independent of the constraints. Hence, the more the constraints play a role in the control problem, the more performance is lost compared to a free parametrization. It is therefore a topic of current research to choose the basis functions such that more robustness of the parametrization for active constraints is obtained.

References

[1] Anderson, B. and J. Moore (1989). *Optimal Control, Linear Quadratic Methods.* Prentice-Hall Inc., Englewood Cliffs, NJ.

[2] Cutler, C. and B. Ramaker (1980). Dynamic matrix control, a computer algorithm. *Proc. Joint Automatic Control Conf.*, paper WP5-B.

[3] Gattu, G. and E. Zafirou (1992). Nonlinear Quadratic Dynamic Matrix Control with state-estimation. *Ind. Eng. Chem. Res.*, Vol.31, no.4, pp.1098-1104.

[4] Jonckheere, E. and L. Silverman (1983). A new set of invariants for linear systems, application to reduced order compensator design. *IEEE Trans. Autom. Control*, Vol, 28, no. 10, pp. 953-964.

[5] Lee, J., Y. Chikkula and Z. Yu (1995). Improving computational efficiency of model predictive control algorithms using wavelet transformation. *Int. J. Control*, Vol. 61, no. 4, pp. 859-883.

[6] Mc Farlane, R., R. Reineman, J. Bartee and C. Georgakis (1993). Dynamic simulator for a model IV fluid catalytic cracking unit. *Computers Chem. Engng.*, Vol. 17, no. 3, pp. 275-300.

[7] Morari, M. and J, Lee (1997). Model predictive control: past, present and future. *Proc. PSE'97 ESCAPE Symposium, Trondheim, Norway*, pp. 1-12.

[8] Rawlings, J. and K. Muske (1993). The stability of constrained receding horizon control. *IEEE Trans. Autom. Control*, Vol. 38, no. 10, pp. 1512-1516.

[9] Richalet, J.. S.E. Ata-Doss, C. Aber, H. Kuntze, A. Jacubasch and W. Schill (1987). Predictive functional control, application to fast and accurate robots. *Proc. 10th IFAC World Congress, Munich, Germany.*

[10] Ricker, N., T. Subrahmanian and T. Sim (1988). Case studies of model predictive control in pulp and paper production. In: McAvoy, T.J. *et al.* (Eds.): *Proc. IFAC Workshop on Model Based Process Control*, p. 13, Pergamon Press, Oxford.

[11] Rossiter,T.A., M.J. Rice, J. Schuurmans and B. Kouvaritakis (1998). A computationally efficient constrained predictive control law. *Proceedings 1998 American Contr. Conf., Philadelphia, PA.*

[12] Scokaert, P. and J. Rawlings (1996). Infinite horizon linear quadratic control with constraints. *Proc. 13th IFAC World Congress*, San Francisco, CA, USA, pp. 109-114.

[13] Zheng, A. and M. Morari (1995). Stability of model predictive control with mixed constraints. *IEEE Trans. Autom. Control*, Vol. 40, no. 10, pp. 1818-1823.

Appendix

The Hessian is $H = R + G^T QG$ and $G_x^T = H_x^T QG$ where

$$G = \begin{bmatrix} 0 & & \cdots & & 0 \\ B & 0 & \ddots & & \\ AB & B & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & \\ & & & & 0 \\ A^{P-1}B & A^{P-2}B & \cdots & AB & B \end{bmatrix}, H_x = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^P \end{bmatrix} \tag{12}$$

where $G \in \mathbb{R}^{Pn \times Pn_u}$ and $H_x \in \mathbb{R}^{Pn \times n}$,

$$Q = diag(Q_1, \ldots, Q_1, Q_0) \in \mathbb{R}^{(P+1)n \times (P+1)n} \text{ and}$$
$$R = diag(Q_2, \ldots, Q_2) \in \mathbb{R}^{Pn_u \times Pn_u} \tag{13}$$

$$\begin{aligned} K_U &= diag(K_u, \ldots, K_u) \in \mathbb{R}^{Pn_u \times Pn_u} \text{ and} \\ k_U &= \begin{bmatrix} k_u^T & \ldots & k_u^T \end{bmatrix}^T \mathbb{R}^{Pn_u} \\ K_X &= diag(K_x, \ldots, K_x)G \in \mathbb{R}^{Pn \times Pn_u} \text{ and} \\ k_X(x_0) &= \begin{bmatrix} k_x^T & \ldots & k_x^T \end{bmatrix}^T - K_X H_x x_0 \in \mathbb{R}^{Pn} \\ K^T &= [K_U^T \; K_X^T] \in \mathbb{R}^{n_c \times Pn_u}, \; k^T = [k_U^T \; k_X^T] \in \mathbb{R}^{n_c} \end{aligned}$$