Model Predictive Control Tuning by Controller Matching

Stefano Di Cairano, *Member, IEEE*, and
Alberto Bemporad, *Senior Member, IEEE*

***Abstract*—The effectiveness of model predictive control (MPC) in dealing with input and state constraints during transient operations is well known. However, in contrast with several linear control techniques, closed-loop frequency-domain properties such as sensitivities and robustness to small perturbations are usually not taken into account in the MPC design. This technical note considers the problem of tuning an MPC controller that behaves as a given linear controller when the constraints are not active (e.g., for perturbations around the equilibrium that remain within the given input and state bounds), therefore inheriting the small-signal properties of the linear control design, and that still optimally deals with constraints during transients. We provide two methods for selecting the MPC weight matrices so that the resulting MPC controller behaves as the given linear controller, therefore solving the posed inverse problem of controller matching, and is globally asymptotically stable.**

***Index Terms*—Constrained linear systems, controller tuning, inverse optimality, model predictive control.**

I. Introduction

Classical methods for controller synthesis usually provide closed-loop stability and a certain degree of robustness and performance, but in general they do not easily account for constraints. Some modifications can be introduced to properly handle constraints, such as anti-windup schemes for input saturation [1]. However these usually work only for a restricted class of constraints, are complicated to design (especially for multivariable systems), and may yield to reduced closed-loop performance. A more systematic way of handling constraints is to resort to model predictive control (MPC) strategies [2], [3]. At every control cycle, MPC uses the current state information to predict the evolution of the system over a given future horizon. Accordingly, MPC selects the input sequence that results in the best performance among the ones that satisfy the constraints. However, as discussed in [2], the stability, robustness, and frequency-domain properties of MPC are sensibly more difficult to characterize with respect to linear feedback controllers. This reduces the transfer of the MPC technology to applications [3].

In this technical note we address the following inverse problem: How to select the performance index (in particular, the weighting matrices) of a linear MPC controller so that it behaves as a given favorite linear controller when the constraints are not active. Hence, for perturbations around an equilibrium point in the interior of the admissible ranges for input and state variables, the closed-loop properties of the MPC controller match those of the original linear controller. The exact region of the state space where the matching occurs (i.e., where the constraints are inactive) can be easily computed from the Karush-Kuhn-Tucker conditions of optimality of the quadratic programming problem associated with MPC, see e.g., [4]. The advantage is that, contrary to the linear controller, the resulting MPC controller is able to properly handle the constraints during transient operations, and that global stability of the closed-loop system in the presence of constraints can be enforced.

A related problem has been recently studied in [5], where the authors prove constructively, yet not computationally, that every continuous nonlinear control system can be obtained by parametric convex programming, and identify MPC as a possible beneficiary. The results presented in the next sections provide computationally feasible procedures to locally solve such a problem for linear control laws within the domain of linear MPC, hence enabling an existing linear control design to handle constraints. We also briefly analyze the inverse optimality problem in the MPC framework [5].

The technical note is structured as follows. In Section II we formulate the MPC matching problem, where the weight matrices in the cost function must be tuned so that, when the constraints are not active, the synthesized MPC feedback law is equivalent to a given linear state-feedback controller. In Sections III and IV we propose the general solution, based on a bilinear matrix inequality (BMI), and introduce a parameterization of the problem that leads to a linear matrix inequality (LMI) formulation. The method introduced in Section III is less conservative, but generates an LMI whose size is proportional to the length of the prediction horizon, while the method in Section IV is more conservative, but the size of the LMI is independent on the horizon length. In Section V we extend the design to dynamic compensators. Some examples are provided in Section VI.

Notation

Relational operators between non-symmetric matrices and vectors are intended componentwise, while for a symmetric matrix $Q = Q' \in \mathbb{R}^{n\times n}$, the notation $Q > 0$ ($Q \geq 0$) denotes positive (semi)definiteness. $\mathbb{R}$, $\mathbb{R}_+$, and $\mathbb{R}_{0+}$ are the set of real, positive real, and nonnegative real numbers, respectively, $\mathbb{Z}$, $\mathbb{Z}_+$ and $\mathbb{Z}_{0+}$, the set of integers, positive integers, and nonnegative integers, respectively. $\mathbb{Z}_{[a,b]}$ denotes the set $\{r \in \mathbb{Z} : a \leq r \leq b\}$. For a given vector v, we indicate by $[v]_i$ its ith component. I_n denotes the identity matrix of order n, and $0_{n,m} \in \mathbb{R}^{n\times m}$ denotes a matrix entirely composed of zeros (subscripts will be dropped when clear from the context). We denote the interior of a set $\mathcal{X}$ by $\text{int}(\mathcal{X})$, and the origin of a vector space by $\mathbf{0}$. Given the dynamical system $x(k+1) = \phi(x(k))$, a set $\mathcal{X}$ is positively invariant (PI) for $\phi(\cdot)$ if for all $x \in \mathcal{X}$, $\phi(x) \in \mathcal{X}$.

II. Controller Matching Problem

Model predictive control is based on solving at every control cycle $k = 0, 1, \ldots,$ the finite-horizon optimal control problem

$$\mathcal{V}(x(k)) = \min_{U(k)} x'(N|k)Px(N|k) + \sum_{i=0}^{N-1} x'(i|k)Qx(i|k) + u'(i|k)Ru(i|k) \tag{1a}$$

$$\text{s.t. } x(i+1|k) = Ax(i|k) + Bu(i|k), \quad i = 0, \ldots, N-1 \tag{1b}$$

$$x_{\min} \leq x(i|k) \leq x_{\max}, \quad i = 0, \ldots, N \tag{1c}$$

$$u_{\min} \leq u(i|k) \leq u_{\max}, \quad i = 0, \ldots, N-1 \tag{1d}$$

$$x(0|k) = x(k) \tag{1e}$$

Manuscript received January 08, 2009; revised May 31, 2009. First published December 01, 2009; current version published January 13, 2010. This work was supported in part by the European Commission under the HYCON Network of Excellence under contract FP6-IST-511368, and under project WIDE, contract P7-IST-224168. Recommended by Associate Editor R. D. Braatz.

S. Di Cairano was with the Information Engineering Department, University of Siena, Siena 53100, Italy. He is now with the Ford Research and Advanced Engineering, Dearborn, MI 48124 USA (e-mail: dicairano@ieee.org).

A. Bemporad is with the Information Engineering Department, University of Siena, Siena 53100, Italy (e-mail: bemporad@dii.unisi.it).

Color versions of one or more of the figures in this technical note are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2009.2033838

0018-9286/$26.00 © 2009 IEEE

Authorized licensed use limited to: Alberto Bemporad. Downloaded on January 18, 2010 at 15:55 from IEEE Xplore. Restrictions apply.

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector, N is the prediction horizon, $U(k) = [u'(0|k) \dots u'(N-1|k)]' \in \mathbb{R}^{Nm}$ is the vector to be optimized [1], and $\mathcal{V} : \mathbb{R}^n \to \mathbb{R}$ is the *value function*. The performance criterion to be optimized is defined by (1a). Since now on, even if not stated explicitly, we assume that matrices $Q, P \in \mathbb{R}^{n \times n}$, $R \in \mathbb{R}^{m \times m}$ satisfy the conditions $Q = Q' \geq 0$, $P = P' \geq 0$, $R = R' > 0$. Equation (1b) defines the prediction model, which is assumed to be completely reachable, and (1c), (1d) define state and input constraints, respectively.

Given the current state $x(k)$, the finite horizon optimal control problem (1) can be reformulated as the following quadratic program (QP) with respect to $U(k)$

$$\min_{U(k)} \quad U'(k)HU(k) + 2x'(k)FU(k) \tag{2a}$$

$$\text{s.t.} \quad GU(k) \leq W + Mx(k). \tag{2b}$$

In (2), $G \in \mathbb{R}^{q \times Nm}$, $M \in \mathbb{R}^{q \times n}$, and $W \in \mathbb{R}^q$ define the problem constraints, while the cost function is defined by

$$H = (\mathcal{R} + \mathcal{S}'\mathcal{Q}\mathcal{S}), \quad F = \mathcal{T}'\mathcal{Q}\mathcal{S} \tag{3}$$

where $H > 0$, $\mathcal{S}$ is the N-steps state reachability matrix, $\mathcal{T}$ is the N-steps free evolution matrix

$$\mathcal{S} = \begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \dots & B \end{bmatrix}, \quad \mathcal{T} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}$$

and $\mathcal{Q} \in \mathbb{R}^{Nn \times Nn}$, $\mathcal{R} \in \mathbb{R}^{Nm \times Nm}$ are block-diagonal matrices

$$\mathcal{Q} = \begin{bmatrix} Q & 0 & 0 & \dots & 0 \\ 0 & Q & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & Q & 0 \\ 0 & 0 & \dots & 0 & P \end{bmatrix}, \quad \mathcal{R} = \begin{bmatrix} R & 0 & \dots & 0 \\ 0 & R & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & R \end{bmatrix}.$$

We denote by $U^*(k)$ the optimizer of problem (2)[2]. Given $U^*(k)$, the ith constraint is called *active* at optimality if $[GU^*(k)]_i = [W + Mx(k)]_i$, $i \in \mathbb{Z}_{[1,q]}$. If no constraint is active, the optimizer of (2) is the unconstrained solution

$$U^*(k) = \begin{bmatrix} u^*(0|k) \\ \vdots \\ u^*(N-1|k)cr \end{bmatrix} = -H^{-1}F'x(k). \tag{4}$$

In this case, the MPC command at step k is

$$u_{\text{MPC}}(x(k)) = u^*(0|k) = -\Psi H^{-1}F'x(k) \tag{5}$$

where matrix $\Psi = [I_m \quad 0 \quad \dots \quad 0]$ extracts the first move actually applied to the process from the optimal sequence $U^*(k)$. For given system dynamics and prediction horizon (i.e., for fixed $\mathcal{S}$ and $\mathcal{T}$), the matrix $-H^{-1}F'$ obtained by $\bar{Q}$, $\bar{R}$, and $\bar{P}$ is the same as the one obtained by $\sigma\bar{Q}$, $\sigma\bar{R}$, and $\sigma\bar{P}$, where $\sigma \in \mathbb{R}_+$ is an arbitrary positive scaling factor. Thus, for numerical reasons and since $R > 0$, it is not restrictive to require $R \geq \sigma I$, where $\sigma \in \mathbb{R}_+$ is a (small) positive constant.

[1]The results can be extended to the case of a shorter control horizon $N_u < N$, $u(i|k) = K_f x(N_u - 1|k)$, $i = N_u, \dots, N-1$, where K_f is a given control law.

[2]The reformulation of (1a) results also in a constant term $Y = x(k)'(Q + \mathcal{T}'Q\mathcal{T})x(k)$ which is not included in (2a), since it does not affect $U^*(k)$.

In industrial practice MPC strategies are often used more for their capability to handle constraints than for performance optimization, meaning that there are several unexploited degrees of freedom in choosing the weight matrices Q, R, P. On the other hand, these affect the robustness properties and the frequency-domain response for small signals of MPC [2], [3], which are very difficult to shape by design. In this technical note we propose to choose Q, R, and P, that is to tune the MPC cost function, by solving the following problem:

Problem 1 (MPC Matching): For a pre-assigned "favorite controller"

$$u_{\text{fv}}(k) = Kx(k), \; K \in \mathbb{R}^{m \times n} \tag{6}$$

define the cost function (1a) such that the *unconstrained* MPC controller (5) based on (1) is equal to the favorite controller (6), that is $u_{\text{fv}}(k) = -\Psi H^{-1}F'x(k)$. ■

In general the MPC behavior will be different from the favorite controller during transients, when active constraints are dealt with, but when the constraints are inactive, the MPC will behave as the favorite controller, hence it will inherit properties such as robustness, sensitivity, and stability. The set of states $x(k)$ where the matching occurs is the polyhedron $\mathcal{P} = \{x \in \mathbb{R}^n : -(GH^{-1}F + M)x \leq W\}$, where the unconstrained optimizer $-H^{-1}Fx$ satisfies the constraints of the QP problem (2).

A direct approach to solve the posed controller matching problem based on the reformulation of MPC problem (1) as a tracking problem is given in [6]. Even though the approach in [6] results in a simple design, global stability is hard to guarantee a priori, due to the tracking formulation with time-varying references. In the next sections we solve Problem 1 by inverse matching, namely by selecting appropriate weight matrices in (1a) so that the unconstrained behavior of the resulting MPC controller "matches" the one of the favorite controller. Global asymptotic stability in the set of feasible initial conditions can be enforced for the proposed approaches.

III. Inverse Matching Based on QP Matrices

Problem 1 is immediately solved if one can find weight matrices Q, R, P in (1) such that

$$-\Psi H^{-1}F'x(k) = Kx(k). \tag{7}$$

Unfortunately (7) is not trivial to solve, due to the non-invertibility of matrix Ψ and the way H^{-1} depends on Q, R, P. To solve Problem 1, we remove Ψ in (7) setting

$$H^{-1}F'x(k) = -\begin{bmatrix} \kappa_0 \\ \kappa_1 \\ \vdots \\ \kappa_{N-1} \end{bmatrix} x(k) \tag{8}$$

where we fix $\kappa_0 = K$, while $\kappa_i \in \mathbb{R}^{m \times n}$, $i \in \mathbb{Z}_{[1,N-1]}$, are free matrices. In (8) we account for the whole optimal input sequence of (1), but we enforce the match with the favorite controller only for the first control action, according to the receding horizon mechanism of MPC.

Lemma 1: Let $(\tilde{\mathcal{K}}, \tilde{Q}, \tilde{R}, \tilde{P})$ be any feasible solution of the following problem

$$\min_{\mathcal{K},Q,R,P} \quad J(\mathcal{K}, Q, R, P) \tag{9a}$$

$$\text{s.t.} \quad Q \geq 0, \quad P \geq 0, \quad R \geq \sigma I \tag{9b}$$

$$(\mathcal{R} + \mathcal{S}'\mathcal{Q}\mathcal{S})\mathcal{K} + \mathcal{S}'\mathcal{Q}\mathcal{T} = 0 \tag{9c}$$

$$\kappa_0 = K \tag{9d}$$

Authorized licensed use limited to: Alberto Bemporad. Downloaded on January 18, 2010 at 15:55 from IEEE Xplore. Restrictions apply.

where $\mathcal{K} = [\kappa_0' \, \ldots \, \kappa_{N-1}']'$, and $J : \mathbb{R}^{Nm\times n} \times \mathbb{R}^{n\times n} \times \mathbb{R}^{m\times m} \times \mathbb{R}^{n\times n} \to \mathbb{R}$ is an (arbitrary) objective function. Then, the MPC strategy based on the optimal control problem (1) where we set $Q = \tilde{Q}, P = \tilde{P}$, $R = \tilde{R}$, solves Problem 1.

Proof: Equality (9c) represents (8) after multiplying both sides by H, which is invertible. Constraint (9d) enforces the equality between the MPC command (5) and the favorite controller (6). Finally, (9b) ensures that the obtained matrices define a valid cost function for the MPC problem (1). Any solution satisfying (9b), (9c), (9d) results in (7) when constraints in (1) are not active. ■

Due to the bilinear constraint (9c), (9) is a nonconvex mathematical program. Regarding the choice of function J, although Problem 1 is solved by any feasible solution of (9) one should notice that the resulting optimal triplet (Q, R, P) affects the behavior of the MPC controller when the constraints are active. A possible choice for J is to specify a triplet $(\bar{Q}, \bar{R}, \bar{P})$ of desired weights and set

$$J(\mathcal{K}, Q, R, P) = \|Q - \bar{Q}\| + w_R\|R - \bar{R}\| + w_P\|P - \bar{P}\| \quad (10)$$

where $w_R, w_P \in \mathbb{R}_{0+}$ and $\|\cdot\|$ is any matrix norm.

The following lemma is immediate to prove and covers the case in which it is not possible to find matrices Q, R and P that exactly solve Problem 1.

Lemma 2: Consider the problem

$$\min_{\mathcal{K},Q,R,P,V} \quad \|\mathcal{K} + V\mathcal{S}'\mathcal{Q}\mathcal{T}\| \quad (11a)$$

$$\text{s.t.} \quad Q \geq 0, \quad P \geq 0,$$

$$R \geq \sigma I, \quad \kappa_0 = K \quad (11b)$$

$$V\mathcal{R} + V\mathcal{S}'\mathcal{Q}\mathcal{S} = I \quad (11c)$$

let J^* be the optimum and $(\mathcal{K}^*, Q^*, R^*, P^*, V^*)$ be any optimizer.[3] If $J^* = 0$, Problem 1 is exactly solved. Otherwise, Q^*, R^*, P^* provide the closest approximation $K^* = -\Psi V^* F'$ of K in the matrix norm $\|\cdot\|$. ■

Note that because of (11a), $V = H^{-1}$ and (11a) equals $\|\mathcal{K} - (-H^{-1}F')\|$.

Due to the bilinear terms in (11c), (11) is nonconvex. Even though nowadays solvers exist for bilinear problems, the convergence to a solution is not guaranteed. In order to formulate (9) as a convex problem with linear matrix inequality (LMI) constraints, one can fix the whole vector

$$\mathcal{K} = \bar{\mathcal{K}} \triangleq \begin{bmatrix} K \\ \bar{\kappa}_1 \\ \vdots \\ \bar{\kappa}_{N-1} \end{bmatrix} \quad (12)$$

where $\bar{\kappa}_i, i \in \mathbb{Z}_{[1,N-1]}$, are pre-assigned gains. This obviously further constrains the design problem. In order to recover additional degrees of freedom in solving Problem 1 we may allow a time-varying cost function in (9) by setting

$$\mathcal{Q} = \begin{bmatrix} Q_1 & 0 & 0 & \ldots & 0 \\ 0 & Q_2 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & Q_{N-1} & 0 \\ 0 & 0 & \ldots & 0 & P \end{bmatrix},$$

$$\mathcal{R} = \begin{bmatrix} R_0 & 0 & \ldots & 0 \\ 0 & R_1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & R_{N-1} \end{bmatrix}$$

where $Q_i = Q_i' \geq 0$, $R_i = R_i' \geq \sigma I$, for all $i \in \mathbb{Z}_{[0,N-1]}$. Since $x(0|k)$ is fixed in (1), the optimizer does not depend on the weight Q_0, which is therefore ignored since now on.

Lemma 3: Let

$$\bar{\mathcal{K}} = \begin{bmatrix} K \\ K(A+BK) \\ \vdots \\ K(A+BK)^{N-1} \end{bmatrix} \quad (13)$$

and consider the following *convex* problem with LMI constraints

$$J^* = \min_{\mathcal{Q},\mathcal{R}} \|(\mathcal{R} + \mathcal{S}'\mathcal{Q}\mathcal{S})\bar{\mathcal{K}} + \mathcal{S}'\mathcal{Q}\mathcal{T}\| \quad (14a)$$

$$\text{s.t. } P \geq 0, \quad R_i \geq \sigma I, \quad i = 0, \ldots, N-1 \quad (14b)$$

$$Q_i \geq 0, \quad i = 1, \ldots, N-1. \quad (14c)$$

in the variables Q_i, $i \in \mathbb{Z}_{[1,N-1]}$, R_i, $i \in \mathbb{Z}_{[0,N-1]}$, and P. If the optimum of (14) is $J^* = 0$, the MPC controller with cost function $x'(N|k)P^*x(N|k) + \sum_{i=0}^{N-1} x'(i|k)Q_i^*x(i|k) + u'(i|k)R_i^*u(i|k)$, where P^*, Q_i^*, R_i^*, $i \in \mathbb{Z}_{[0,N-1]}$ are the optimizer of (14), behaves as the favorite controller (6) when the constraints are not active. ■

By imposing (13) one requires that the controller (6) is applied along the entire prediction horizon. This is more stringent than what is required to solve Problem 1.

If the optimum of (14) is not 0, an exact solution to Problem 1 is not found. Instead, when constraints are not active the MPC controller synthesized from (14) when $J^* \neq 0$ best approximates (6) according to the selected norm $\|\cdot\|$ weighted by $H > 0$. This in turn implies that the desired closed-loop state evolution is also matched in an approximated way. In fact, let $K^* = -\Psi V^* \mathcal{S}'\mathcal{Q}^*\mathcal{T}$ be the approximating controller obtained by solving (14), and the norm in (14a) be an induced matrix norm. When the constraints are not active the difference between the desired and attained state update is $\|(Ax + Bu_{\text{fv}}) - (Ax + Bu_{\text{MPC}})\| \leq \|B\|\,\|K - K^*\|\,\|x\|$ for any $x \in \mathbb{R}^n$. In this technical note we focus on the study of the exact solutions of Problem 1, and we defer the complete analysis of the approximated solutions to future research.

A. Global Closed-Loop Stability

If perfect matching occurs and the favorite controller is stabilizing, *local* stability of the MPC closed-loop also follows immediately. However, as for general MPC [2], *global* stability is more complicated to guarantee, especially because here we want to maintain local equivalence with the favorite controller (6). The approach described in [2] based on terminal cost and terminal set can be specialized for this purpose.

Theorem 1: Let $\mathcal{X}_T \subseteq \mathbb{R}^n$ be a polyhedral positively invariant set for (1a) in closed loop with (6) such that $\mathbf{0} \in \text{int}(\mathcal{X}_T)$ and $\mathcal{X}_T \subseteq \{x \in \mathbb{R}^n : x_{\min} \leq x \leq x_{\max}, u_{\min} \leq Kx \leq u_{\max}\}$, and add constraint $x(N|k) \in \mathcal{X}_T$ to (1). Let Q, R, P be computed either by (9), or by (14) under the restriction $Q_i = Q$, $R_i = R$, for all $i \in \mathbb{Z}_{[0,N-1]}$. In (9), or (14), add the LMI constraint

$$(A+BK)'P(A+BK) + K'RK + Q - P \leq 0. \quad (15)$$

Denote by $\mathcal{X}_{\text{feas}} \subseteq \mathbb{R}^n$ the set of states $x \in \mathbb{R}^n$ such that (1) is feasible when $x(k) = x$, then, *(i)* the resulting closed-loop MPC dynamics are asymptotically stable and remain in $\mathcal{X}_{\text{feas}}$, for all $x(0) \in \mathcal{X}_{\text{feas}}$; *(ii)* if either (9) is solved, or (14) is solved with zero optimum $(J^* = 0)$,

[3]Due to the non-convexity of (11) multiple global optima and optimizers may exist.

Authorized licensed use limited to: Alberto Bemporad. Downloaded on January 18, 2010 at 15:55 from IEEE Xplore. Restrictions apply.

there exists a set $\mathcal{X}_{\rm fv} \supseteq \mathcal{X}_T$ such that the MPC behaves as the favorite controller (6) for all $x \in \mathcal{X}_{\rm fv}$; *(iii)* if $Q > 0$, $\mathcal{X}_{\rm fv}$ is reached in a finite time $k(x(0))$, for all $x(0) \in \mathcal{X}_{\rm feas}$ and, if in addition $\mathcal{X}_{\rm feas}$ is bounded, there exists a finite $\bar{k} = \max_{x(0)\in\mathcal{X}_{\rm feas}} k(x_0)$.

Proof:

(i) By using the terminal set and terminal constraint approach of [2], it is possible to show that $\mathcal{V}(x)$ is a Lyapunov function in $\mathcal{X}_{\rm feas}$, by taking $\mathcal{X}_T$ as the terminal set, P as the terminal cost matrix, K as the auxiliary controller that is feasible in $\mathcal{X}_T$, and the cost function as in (1a) (details are omitted here for brevity and can be found in [2]).

(ii) Consider $\bar{k}$ such that $x(\bar{k}) \in \mathcal{X}_T$. Since $\mathcal{X}_T$ is positively invariant for $(A+BK)$ and $u = Kx$ satisfies the constraints in $\mathcal{X}_T$, $\tilde{U}(x(\bar{k})) = [(Kx(\bar{k}))' \ldots (K(A+BK)^{N-1}x(\bar{k}))']'$ is feasible for (1). Since the optimum of (1) with Q, R, P computed by (9) (or by (14) with $Q_i = Q$, $R_i = R$, for all $i \in \mathbb{Z}_{[0,N-1]}$, resulting in $J^* = 0$) is achieved for $u_{\rm MPC} = u_{\rm fv}$, $\tilde{U}(x(\bar{k}))$ is optimal. Hence, for all $x \in \mathcal{X}_T$ the MPC behaves as the favorite controller. Since $\mathcal{X}_T$ is positively invariant for dynamics $x(k+1) = (A+BK)x(k)$, for all $k \geq \bar{k}$, $x(k) \in \mathcal{X}_T$, and hence for all $k \geq \bar{k}$, $u_{\rm MPC}(x(k)) = u_{\rm fv}(x(k))$.

(iii) By the results of [2], $\mathcal{V}(x(k)) - \mathcal{V}(x(k+1)) \geq x(k)'Qx(k)$, $Q > 0$. Since $\mathcal{X}_T$ is compact and $\mathbf{0} \in {\rm int}(\mathcal{X}_T)$, there exists $\gamma \in \mathbb{R}_+$ such that $x'Qx > \gamma$ for all $x \notin \mathcal{X}_T$. Assume by contradiction that $x(k) \notin \mathcal{X}_T$, for all $k \in \mathbb{Z}$. Then $\mathcal{V}(x(k)) \leq \mathcal{V}(x(0)) - k\gamma$, and hence $\lim_{k\to\infty} \mathcal{V}((x(k)) = -\infty$, which contradicts $\mathcal{V}(x) \geq 0$, for all $x \in \mathbb{R}^n$. Hence, there must exist a finite $k(x(0))$ such that $x(k(x(0))) \in \mathcal{X}_T$. By *(ii)*, it follows that $x(k) \in \mathcal{X}_T$, for all $k \geq k(x(0))$. If in addition $\mathcal{X}_{\rm feas}$ is bounded, then there exists $V_M = \max_{x(0)\in\mathcal{X}_{\rm feas}} \mathcal{V}(x(0))$, and hence $\mathcal{V}(x(k)) \leq V_M - k\gamma$, for all $x(0) \in \mathcal{X}_{\rm feas}$. By setting $\bar{k} \geq V_M/\gamma$ the result follows. ■

The set $\mathcal{X}_T$ can be for instance the maximum positively invariant set, computed as in [7], while the set $\mathcal{X}_{\rm feas}$ can be analyzed by the techniques in [4].

IV. Matching Based on Inverse LQR

In this section we propose a computationally simpler alternative to solve Problem 1. Instead of solving (7), we use the following theorem (see [4]).

Theorem 2: Given $\bar{Q} \in \mathbb{R}^{n\times n}$, $\bar{Q} \geq 0$, and $\bar{R} \in \mathbb{R}^{m\times m}$, $\bar{R} > 0$, let $\bar{P} \in \mathbb{R}^{n\times n}$, $\bar{P} > 0$ be the solution of the Riccati equation

$$\bar{P} = A'\bar{P}A - A'\bar{P}B(B'\bar{P}B + \bar{R})^{-1}B'\bar{P}A + \bar{Q}. \quad (16)$$

Set $Q = \bar{Q}$, $R = \bar{R}$, and $P = \bar{P}$ in (1a). For any prediction horizon $N = 1, 2, \ldots$, when the constraints are not active, the MPC command (5) obtained by solving (1) is $u_{\rm MPC}(x(k)) = K_{\rm LQR}x(k)$, where $K_{\rm LQR} = -(B'\bar{P}B+\bar{R})^{-1}B'\bar{P}A$ is the LQR gain obtained from (16). ■

Under the hypothesis of Theorem 2, the MPC controller that has no (active) constraints behaves as the linear state feedback gain that optimizes $\min_{u(\cdot)} \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k)$ for the linear dynamics $x(k+1) = Ax(k) + Bu(k)$. For solving Problem 1, we look for weights Q, R and P such that the favorite controller (6) is the corresponding LQR gain. In this way when constraints are not active, by Theorem 2 the MPC behaves as the LQR for any horizon $N = 1, 2, \ldots$, hence also as the favorite controller (6).

Corollary 1: Consider the convex optimization problem with LMI constraints

$$\min_{Q,R,P} \quad J(Q,R,P) \quad (17a)$$

$$\text{s.t.} \quad P \geq 0, \quad R \geq \sigma I, \quad Q \geq 0 \quad (17b)$$

$$P = A'PA + A'PBK + Q \quad (17c)$$

$$B'PA = -(B'PB + R)K \quad (17d)$$

where $J : \mathbb{R}^{n\times n} \times \mathbb{R}^{m\times m} \times \mathbb{R}^{n\times n} \to \mathbb{R}$ is convex (e.g., as in (10)) and (17b), (17c), (17d) are linear matrix (in)equalities. Let $\tilde{Q}$, $\tilde{R}$, $\tilde{P}$ be any feasible solution (not necessarily the optimal one) of (17). Then the MPC strategy based on the optimal control problem (1) where we set $Q = \tilde{Q}$, $P = \tilde{P}$, $R = \tilde{R}$, solves Problem 1.

Proof: Equalities (17c), (17d), and constraint (17b) enforce P and K to be the solution of the Riccati equation and the corresponding LQR gain, respectively, and Q, R, to be the corresponding weights. Thus, given any $\tilde{Q}$, $\tilde{R}$, $\tilde{P}$ that are a feasible solution of (17), K is the LQR gain that optimizes the LQR cost, where $Q = \tilde{Q}$, $R = \tilde{R}$. Theorem 2 guarantees that the optimal control problem (1) where we set $Q = \tilde{Q}$, $R = \tilde{R}$, $P = \tilde{P}$ results in an MPC command (5) equal to the one that would be issued by the LQR, and hence also by the favorite controller (6) whenever the constraints in the MPC quadratic program (2) are not active. ■

Working with LQR gains introduces additional constraints, since an infinite-horizon inverse-optimal cost function is sought. In fact, Theorem 2 and Lemma 3 ensure that whenever a solution of (17) exists, a solution of (14) with $J^* = 0$ exists. The opposite is not guaranteed. However, (17) involves a simpler LMI than (14), especially when the horizon N is large, because the number of variables in the LMI is independent of N.

The additional requirement of (6) being an LQR gain implies that the feasibility of (17) cannot be guaranteed, because there exists $K \in \mathbb{R}^{n\times m}$ that are not LQR gain for any quadratic cost function. The problem of checking whether a given controller is optimal with respect to some performance criterion (inverse optimality problem) was first introduced by Kalman [8]. The inverse LQ design problem has been studied in [9], where the conditions for a linear state-feedback controller to be an LQR gain were analyzed. The works [8]–[10] focused on continuous-time systems, and provided algebraic conditions for a given linear state feedback law to be an LQR. The inverse LQR design for discrete-time systems can be solved by convex optimization using (17). Formulations based on (17) that provide approximate solutions of Problem 1 when K is not an LQR gain can be simply obtained, and will not be discussed here.

Global stability in the set of feasible initial conditions in the presence of constraints can be achieved by choosing N large enough [2], [4], which does not increase the complexity of (17), since (17) is independent of N. Also, Theorem 1 can be applied using the Riccati matrix P as the terminal weight and the associated $K_{\rm LQR} = K$ as the auxiliary controller, since P computed from (17) satisfies (15).

V. Extension to Dynamic Controllers

Although the MPC tuning techniques presented in the previous sections were developed for static state-feedback favorite controllers, they can be easily extended to dynamic output-feedback favorite controllers. For the simplicity of notation, we consider a SISO favorite dynamic controller $U(z) = C(z)Y(z)$ with strictly proper transfer function $C(z) = A(z)/B(z)$ that can be expressed by the difference equation

$$u(k) = \sum_{i=0}^{n_a^c} a_i y(k-i) + \sum_{i=1}^{n_b^c} b_i u(k-i). \quad (18)$$

The state-space model (1b) of the process, together with an output equation $y(k) = Cx(k)$, can be formulated as the transfer function $Y(z) = G(z)U(z)$, where $G(z) = E(z)/D(z) = C(zI-A)^{-1}B$, and hence as the difference equation $y(k) = \sum_{i=1}^{n_e} e_i u(k-i) + \sum_{i=1}^{n_d} d_i y(k-i)$. Let $n_a = \max\{n_a^c, n_d\}$,

Authorized licensed use limited to: Alberto Bemporad. Downloaded on January 18, 2010 at 15:55 from IEEE Xplore. Restrictions apply.

$n_b = \max\{n_b^c, n_e\}$, and define the state vector $x(k) = [\,y(k-1)\ldots y(k-n_a)\; u(k-1)\ldots u(k-n_b)\,]'$. The process dynamics are described by the (possibly non-minimal [4]) state-space realization

$$\begin{aligned} x(k+1) &= \begin{bmatrix} d_1 \ldots d_{n_a} & e_1 \ldots e_{n_b} \\ L_{n_a} & 0 \\ 0 \ldots 0 & 0 \ldots 0 \\ 0 & L_{n_b} \end{bmatrix} x(k) \\ &\quad + \begin{bmatrix} 0 \\ 0_{n_a-1,1} \\ 1 \\ 0_{n_b-1,1} \end{bmatrix} u(k) \\ y(k) &= [\, d_1 \ldots d_{n_a} \quad e_1 \ldots e_{n_b} \,]\, x(k) \end{aligned} \tag{19}$$

where $d_i = 0$ for all $i \in \mathbb{Z}_{[n_d+1,n_a]}$, $e_i = 0$ for all $i \in \mathbb{Z}_{[n_e+1,n_b]}$, and $L_n = [\,I_{n-1} \quad 0_{n-1,1}\,] \in \mathbb{R}^{n-1\times n}$. For this choice of the state vector, the favorite controller is $u(k) = Kx(k)$, $K = [\,a_1 + a_0 d_1 \cdots a_{n_a} + a_0 d_{n_a} \quad b_1 + a_0 e_1 \cdots b_{n_b} + a_0 e_{n_b}\,]$, where $a_i = 0$ for all $i \in \mathbb{Z}_{[n_a^c+1,n_a]}$, and $b_i = 0$ for all $i \in \mathbb{Z}_{[n_b^c+1,n_b]}$. Thus, the techniques of the previous sections can be applied immediately to the case of dynamic output-feedback.

When the constraints are active, the MPC command can be different from the one of the favorite controller. Thus the dynamics of (18) will be different from the desired one. However, as soon as the constraints become inactive, the functional form (18) of the controller is enforced, even though the sequence of previous inputs and outputs was different from the one that would have been generated by the system in closed-loop with the favorite controller.

VI. Examples

Example 1 (Matching Based on QP Matrices): Consider the unstable linear system $x(k+1) = Ax(k) + Bu(k)$, where

$$A = \begin{bmatrix} 0.675 & 0.923 & 0.014 \\ -0.315 & 0.215 & -0.750 \\ 1.05 & 0.00 & 1.50 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.5 \\ 0 \\ 1 \end{bmatrix}, \quad x \in \mathbb{R}^3, \;\; u \in \mathbb{R}$$

and the favorite controller $u_{\rm fv} = Kx$, where $K = [\,-0.918 \quad 0.347 \quad -0.806\,]$ is designed by pole-placement so that $A + BK$ has eigenvalues $\{0.75, 0.375, 0.40\}$. Assume that the input constraints $-2.5 \le u \le 2.5$ are present. We solve (14) where we have set $N = 3$, $\sigma = 10^{-3}$, $\bar{\mathcal{K}}$ as in (13), and we have imposed $Q_i = Q \ge 0$, $R_i = R > \sigma I$, for all $i \in \mathbb{Z}_{[1,N-1]}$. The solution has optimal cost $J^* = 0$, and optimizer matrices

$$Q^* = \begin{bmatrix} 3.813 & -0.884 & -1.199 \\ -0.884 & 0.528 & 0.539 \\ -1.199 & 0.539 & 0.619 \end{bmatrix}, \quad R^* = 0.376,$$

$$P^* = \begin{bmatrix} 1.808 & 0.501 & 0.028 \\ 0.501 & 0.343 & 0.393 \\ 0.028 & 0.393 & 0.835 \end{bmatrix}.$$

Thus, by setting $Q = Q^*$, $R = R^*$, $P = P^*$ in (1), whenever the constraints are not active, $u_{\rm MPC} = u_{\rm fv}$. Let $\{x(k)\}_{k=0}^{20}$ be the closed-loop state trajectory obtained by MPC along 20 steps, where $x(0) =$

[4]In general, the process state-space realization (19) is not minimal, but as discussed in [11] it is completely reachable if the polynomials $N(z)$, $D(z)$ are coprime.

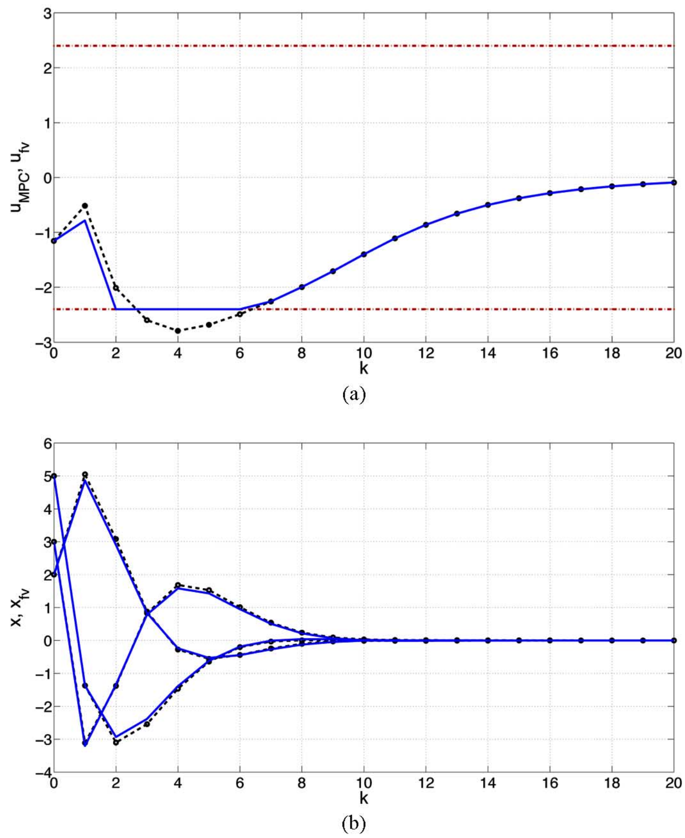

Fig. 1. Example 1, MPC matching based on QP matrices. (a), $K = [-0.918\; 0.347\; 0.806]$, $u_{\rm MPC}$ (solid) and $u_{\rm fv}$ (dashed with circle markers). (b), $K = [-0.8\; 0.6\; 0.5]$. Closed-loop trajectory generated by MPC (solid) and by favorite controller (dashed with circle markers).

$[3\;5\;2]'$. The applied MPC command $u_{\rm MPC}(x(k))$ together with the hypothetical (not applied) favorite controller command $u_{\rm fv}(x(k))$ are shown in Fig. 1(a). For this example the LMI (17) is infeasible. In fact the approach of Section III requires matching (and optimality) along a finite horizon, and is less conservative than the one in Section IV, requiring matching along an infinite horizon.

If the favorite controller is changed to $K = [\,-0.8 \quad 0.6 \quad -0.5\,]$ an exact solution to the matching problem is not found. The optimal solution to (14) has cost $J^* = 0.032$ and the MPC whose cost is based on the new resulting matrices Q^*, R^*, P^* is $u_{\rm MPC} = [\,-0.825 \quad 0.485 \quad -0.618\,]\,x$ in the absence of constraints. However, as shown in Fig. 1(b), the closed-loop trajectories $\{x(k)\}_{k=0}^{20}$ with MPC and $\{x_{\rm fv}(k)\}_{k=0}^{20}$ with the favorite controller, with same initial condition $x_{\rm fv}(0) = x(0)$, are very close (note that in this simulation the input constraints are never active).

Example 2 (Matching Based on Inverse LQR): Consider the linear system $y(k) = 1.8y(k-1) + 1.2y(k-2) + u(k-1)$ with sampling time $T_s = 2$, constrained input $-24 \le u \le 24$, and output constraint $y \ge -5$. Consider the favorite discrete-time PID controller

$$\begin{aligned} u_{\rm fv}(x(k)) &= -\Bigg(K_I \mathcal{I}(k) + K_P y(k) \\ &\qquad + \frac{K_D}{T_s}(y(k) - y(k-1)) \Bigg), \\ \mathcal{I}(k) &= \mathcal{I}(k-1) + T_s y(k) \end{aligned} \tag{20}$$

Authorized licensed use limited to: Alberto Bemporad. Downloaded on January 18, 2010 at 15:55 from IEEE Xplore. Restrictions apply.

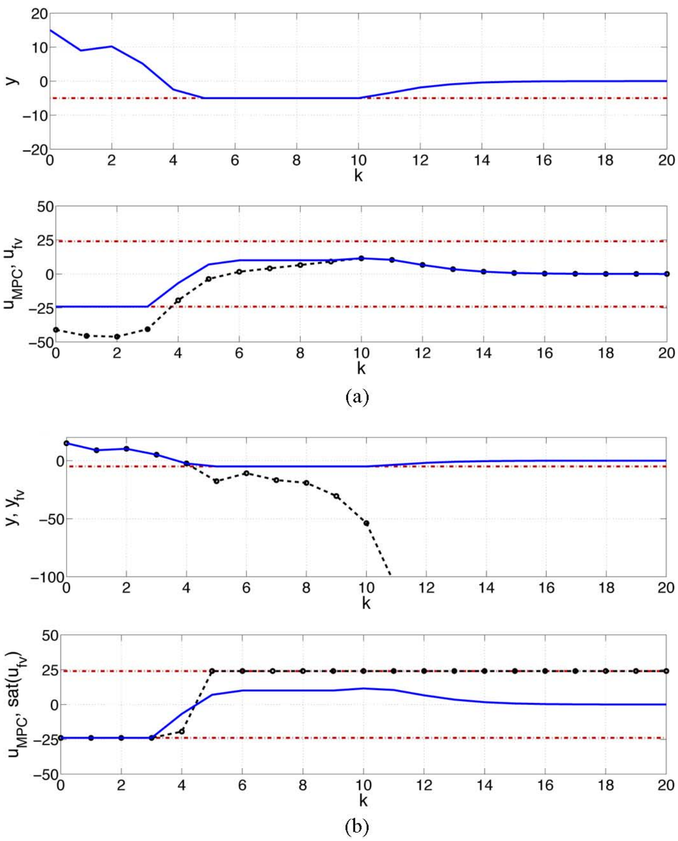

Fig. 2. Example 2, MPC matching of a PID controller based on inverse LQR. (a) upper plot, output trajectory generated by MPC, lower plot, $u_{\rm mpc}$ (solid) and $u_{\rm fv}$ (dashed with circle markers). (b), output trajectory (upper plot) and control input (lower plot) of MPC (solid) and saturated favorite controller (dashed with circle markers).

where $K_I = 0.248$, $K_P = 0.752$, $K_D = 2.237$. As shown in Section V, we obtain the state-space representation $x(k+1) = Ax(k) + Bu(k)$, where

$$x(k) = \begin{bmatrix} y(k-1) \\ y(k-2) \\ \mathcal{I}(k-1) \\ u(k-1) \end{bmatrix},$$

$$A = \begin{bmatrix} 1.8 & 1.2 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 3.6 & 2.4 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

and the PID (20) is expressed as $u_{\rm fv} = Kx$, where $K = -[\,K_D/T_s + 1.8K_{pid}\ 1.2K_{pid}\ \ K_I\ \ K_{pid}\,]$, $K_{pid} = K_P + K_I T_s + K_D/T_S$. The inverse LQR-based matching problem (17) where $\sigma = 10^{-3}$, and with objective function (10), where $\bar{Q} = 6.5I$, $\bar{R} = 1$, $w_R = 1$, $w_P = 0$ returns the optimal matrices

$$Q^* = \begin{bmatrix} 6.401 & 0.064 & -0.001 & 0.019 \\ 0.064 & 6.605 & 0.006 & 0.080 \\ -0.001 & 0.006 & 6.647 & -0.020 \\ 0.019 & 0.080 & -0.020 & 6.378 \end{bmatrix}, \quad R^* = 1,$$

$$P^* = \begin{bmatrix} 422.7 & 241.7 & 50.39 & 201.4 \\ 241.7 & 151.0 & 32.13 & 120.4 \\ 50.39 & 32.13 & 19.85 & 26.75 \\ 201.4 & 120.4 & 26.75 & 106.6 \end{bmatrix}.$$

The MPC strategy (1) with $N = 3$ and $Q = Q^*$, $R = R^*$, $P = P^*$ is implemented so that $u_{\rm MPC} = u_{\rm fv}$ whenever the constraints are not active. Note that the approach of Section IV is independent of N, hence the prediction horizon can be changed without recomputing the weight matrices. The upper plot of Fig. 2(a) shows the MPC closed-loop output trajectory $\{y(k)\}_{k=0}^{20}$ starting from $x(0) = [5 \quad 5 \quad 0 \quad 0]'$ and the lower plot shows the input commands issued by the MPC and the (hypothetical) ones issued by the favorite controller from the same state. When the input and output constraints become inactive, $u_{\rm MPC} = u_{\rm fv}$. Fig. 2(b) compares the closed-loop output trajectory of the MPC controller $\{y(k)\}_{k=0}^{20}$ with the closed-loop output trajectory $\{y_{\rm fv}(k)\}_{k=0}^{20}$ resulting by applying the favorite controller subject to input saturation. The saturated favorite controller cannot enforce the output constraint, and it becomes unstable due to input constraints.

VII. Conclusion

In this technical note we have provided constructive techniques for designing an MPC controller that, for signals not activating the constraints, inherits the frequency-domain and other linear properties of a given linear controller and that, at the same time, is able to optimally handle constraints on (possibly multiple) inputs and outputs, and to guarantee global closed-loop stability. The design procedures are based on tuning the MPC cost function using the solution of convex optimization problems. The approach can also be interpreted as technique to automatically synthesize an anti-windup scheme, which has a piecewise-affine form [4].

References

[1] K. Åstrom and L. Rundqwist, "Integrator windup and how to avoid it," in *Proc. Amer. Control Conf.*, 1989, pp. 1693–1698.
[2] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.
[3] S. Qin and T. Badgwell, "A survey of industrial model predictive control technology," *Control Eng. Practice*, vol. 93, no. 316, pp. 733–764, 2003.
[4] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.
[5] M. Bares, M. Diehl, and I. Necoara, "Every continuous nonlinear control system can be obtained by parametric convex programming," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1963–1967, Aug. 2008.
[6] S. Di Cairano and A. Bemporad, "Model predictive controller matching: Can MPC enjoy small signal properties of my favorite linear controller?," in *Proc. Eur. Control Conf.*, 2009, pp. 2217–2222.
[7] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Math. Problems Eng.*, vol. 4, pp. 317–367, 1998.
[8] R. Kalman, "When is a linear control system optimal?," *Trans. ASME, J. Basic Eng.*, pp. 51–60, 1964.
[9] T. Fujii, "A new approach to the LQ design from the viewpoint of the inverse regulator problem," *IEEE Trans. Autom. Control*, vol. AC-32, no. 11, pp. 995–1004, Nov. 1987.
[10] K. Sugimoto, "Partial pole placement by LQ regulators: an inverse problem approach," *IEEE Trans. Autom. Control*, vol. 43, no. 5, pp. 706–708, May 1998.
[11] E. Mosca, *Optimal, Predictive, and Adaptive Control*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

Authorized licensed use limited to: Alberto Bemporad. Downloaded on January 18, 2010 at 15:55 from IEEE Xplore. Restrictions apply.