Model Predictive Controller Matching: Can MPC Enjoy Small Signal Properties of My Favorite Linear Controller ?

S. Di Cairano, A. Bemporad

Abstract— **Model predictive control (MPC) strategies can efficiently deal with constraints on system states, inputs, and outputs. However, in contrast with linear control techniques, closed-loop frequency-domain properties of MPC such as sensitivity and robustness to small perturbations are difficult to enforce a priori. This paper considers the problem of transforming a given linear feedback control design, referred to as "favorite controller", into a model predictive control one. In this way, the MPC controller inherits all the stability, robustness and frequency properties of the given favorite controller in the region around the equilibrium where the constraints are not active. The added value is that the constructed MPC controller is able to properly handle constraints that may be activated during the transient, and that global stability in the set of feasible initial conditions can be guaranteed.**

I. Introduction

Modern control systems are required to satisfy a large number of specifications: asymptotic closed-loop stability, robustness with respect to external disturbances and modeling errors, closed-loop performance, and the capability of handling constraints on input and state variables. Industrial control systems are mainly designed to provide stability, and a certain degree of robustness and performance, but in general they do not easily account for constraints. Some modifications have been introduced in standard control systems to properly handle constraints, such as anti-windup schemes for input saturation [1], but these often work for a restricted class of constraints, are complicated to design, and may yield to reduced closed-loop performance.

Model predictive control (MPC) [2] is a control strategy that naturally deals with constraints on system states, inputs, and outputs by solving at every control cycle an optimization problem. Stability of MPC was surveyed in [3] where it is shown that MPC stability, robustness, and frequency-domain behavior are sensibly more difficult to characterize with respect to linear feedback controllers. This has generated reluctance in engineers as regards the use of MPC [4] in industry.

In this paper we propose a set of techniques for solving the following inverse problem: how to select the performance index so that the resulting MPC controller behaves as a given favorite linear controller in a region around the equilibrium where the constraints are not active. The advantage is that, contrary to the linear controller, the resulting MPC is able to properly handle the constraints during transient operations, and that stability of the constrained system can be enforced. We also provide some answers to the question posed in [5] as regards the inverse optimality problem in the MPC framework.

The paper is structured as follows. In Section II we formulate the MPC control matching problem and briefly discuss *direct methods* for solving it in Section III. In Sections IV, V we introduce two *inverse methods*, where the MPC weights are designed so that the resulting MPC law equals the favorite control law in the absence of constraints. Global stability of the inverse methods is guaranteed. The proposed techniques are exemplified in Section VI, and the results summarized in Section VII.

A. Notation

$Q > 0$ ($Q \geq 0$) indicates that a symmetric matrix Q is positive definite (positive semidefinite). Relational operators between vectors are intended componentwise. $\mathbb{R}$ and $\mathbb{R}_{0+}$ are the set of 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 i^{th} component; similarly for a matrix A, $[A]_j$ is its j^{th} column, $[A]^i$ is its i^{th} row, and $[A]_{ij}$ is the element at the i^{th} row, j^{th} column. I_n denotes the identity matrix of order n, $\mathrm{O}_{n,m} \in \mathbb{R}^{n\times m}$ denotes a matrix entirely composed of zeros, where subscripts will be dropped when clear from the context. We denote the interior of a set $\mathcal{X}$ by $\mathrm{int}(\mathcal{X})$, and the origin of a vector space by $\mathbf{0}$. Given dynamics $x(k+1) = \phi(x(k))$, a set $\mathcal{X}$ is positively invariant (PI) for $\phi(\cdot)$ if for all $x \in \mathcal{X}$, $\phi(x) \in \mathcal{X}$.

II. The Controller Matching Problem

Model predictive control is based on solving at every control cycle $k \in \mathbb{Z}_{0+}$ the finite horizon optimal control problem

$$\begin{aligned}\mathcal{V}(x(k)) = \min_{U(k)} \sum_{i=0}^{N-1} & x'(i|k)Qx(i|k) + u'(i|k)Ru(i|k) \\ & +x'(N|k)Px(N|k) && \text{(1a)}\\ \text{s.t.}\quad & x(i+1|k) = Ax(i|k) + Bu(i|k),\ {\scriptstyle i\in\mathbb{Z}_{[0,N-1]}} && \text{(1b)}\\ & x_{\min} \leq x(i|k) \leq x_{\max},\ {\scriptstyle i\in\mathbb{Z}_{[0,N]}} && \text{(1c)}\\ & u_{\min} \leq u(i|k) \leq u_{\max},\ {\scriptstyle i\in\mathbb{Z}_{[0,N-1]}} && \text{(1d)}\\ & x(0|k) = x(k), && \text{(1e)}\end{aligned}$$

A. Bemporad is with the Information Engineering Department, University of Siena, Italy, bemporad@dii.unisi.it. S. Di Cairano was with the Information Engineering Department, University of Siena, Italy, at the time of this work. He has recently joined Powertain Control R&A, Ford Motor Company. This work was partially supported by the European Commission under the HYCON Network of Excellence, contract number FP6-IST-511368, and under project WIDE, contract number FP7-IST-224168.

ISBN 978-963-311-369-1

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, 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 $\mathcal{V} : \mathbb{R}^n \to \mathbb{R}_{0+}$ is the *value function*. The performance criterion to be optimized is defined by (1a), where $Q, P \in \mathbb{R}^{n\times n}$, $R \in \mathbb{R}^{m\times m}$, $Q = Q' \geq 0$, $P = P' \geq 0$, $R = R' > 0$. From now on, we will always assume that Q, R, P are symmetric matrices, even if not stated explicitly. Equation (1b) defines the prediction model, a linear model with state x and input u which is assumed to be completely reachable, and (1c), (1d) define the constraints.

Given the current state $x(k)$, the finite horizon optimal control problem (1) can be reformulated as a quadratic program[2] (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

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

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

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

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

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

We denote by $U^*(k)$ the optimizer of problem (2).

Definition 1: Given $U^*(k)$, the i^{th} constraint, $i \in \mathbb{Z}_{[1,q]}$, is *active* at optimality if $[GU(k)^*]_i = [W + Mx(k)]_i$.

If no constraint is active, the optimizer of problem (2) is the unconstrained solution

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

In this case, the MPC command at step $k \in \mathbb{Z}_{0+}$ is

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

where matrix $\Psi = [\, I_m \; 0 \cdots 0 \,]$ 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 $\bar{P}$, $\bar{Q}$, and $\bar{R}$ is the same as the one obtained by $\sigma\bar{P}$, $\sigma\bar{Q}$, and $\sigma\bar{R}$, where $\sigma \in \mathbb{R}$, $\sigma > 0$, is an arbitrary positive scaling factor. Thus, since $R > 0$, it is not restrictive to require $R \geq \sigma I$, where $\sigma > 0$ is a (small) positive constant. Such a constraint avoids numerically ill-conditioned solutions.

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 P, Q, R. On the other hand, these affect the robustness properties and the frequency-domain response for small signals of MPC [3], [4], which are very difficult to shape by design. This often results in the reluctance of engineers to use MPC as a substitute for simpler control schemes, such as PID or linear state feedback, that do not allow for constraint handling but are well suited for enforcing robust stability and frequency response specifications.

In this paper we want to obtain by design an MPC controller that behaves as a pre-assigned "favorite" controller when the constraints are not active:

Problem 1 (MPC matching): Given a *favorite controller*

$$u_{\text{fv}} = Kx \tag{6}$$

where $K \in \mathbb{R}^{m\times n}$, define the cost function (1a) such that *when the constraints are not active* the MPC control (5) based on (1) is equal to the favorite control (6). The problem is *approximately solved* if, when the constraints are inactive, (5) is as close as possible to (6) in a given criterion.

Of course the MPC behavior in general will be different from the linear one during transients to properly deal with active constraints and possibly achieve global stability.

Before discussing techniques that synthesize MPC feedback laws that solve Problem 1, we briefly discuss techniques based on tracking adequately generated references, called *direct techniques*.

III. DIRECT MATCHING TECHNIQUES

A direct approach to solve Problem 1 is to use the behavior of (1b) in closed loop with (6) to generate a reference signal for the state and input vectors, and to modify (1) to be in the form of state and input tracking [2]. This amounts to consider the state reference dynamics $r_x(i+1|k) = (A + BK)r_x(i|k)$, $r_x(0|k) = x(0|k)$, the reference input $r_u(i|k) = Kr_x(i|k)$, and to modify (1a) to weight $x(i|k)-r_x(i|k)$ and $u(i|k)-r_u(i|k)$. However, this approach has some drawbacks. It results in a time-varying reference tracking formulation, while most of the stabilization results apply to regulation. As the reference generation model is linear, such formulation can be transformed into a state-regulation problem by extending the state vector, but in this case the overall extended system would not be controllable, although stabilizable. This, combined with the initialization, $r_x(0|k) = x(0|k)$ makes the use of standard stabilization approaches [3] more difficult.

Remark 1: Most of the approaches for stabilization by MPC [3] are based on the existence of a feasible solution for (1) at time k obtained by extending the solution computed at time $k-1$. However, whenever $u_{\text{MPC}}(x(k)) \neq u_{\text{fv}}(x(k))$,

[1]The results of this paper can be extended to the case of a control horizon $N_u < N$, $u(i|k) = u(N_u - 1|k)$, $i = N_u, \dots, N-1$.

[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 the optimizer.

$r_x(0|k) = x(0|k) \neq r_x(1|k-1)$, hence the input sequence obtained at time $k-1$ cannot be extended, due to a different initial state. A solution in which the reference closed-loop dynamics evolves autonomously, i.e., $r_x(k) = r_x(1|k-1)$, cannot guarantee that whenever (6) is feasible $u_{\rm MPC}(x(k)) = u_{\rm fv}(x(k))$, due to cumulated errors.

An approach for which stability can be analyzed is based on input tracking only. In particular, one can use

$$J(x(k), U(k)) = (u(0|k) - Kx(k))' Q_K (u(0|k) - Kx(k)), \tag{7}$$

as the cost function in (1a), where $Q_K > 0$ has to be determined. Cost function (7) provides $u(0|k) = Kx(k)$ for any $Q_K > 0$ whenever the constraints (1c)–(1d) are not active. The minimum value of the cost function is achieved when the MPC algorithm produces the same input as the favorite controller. Thus, whenever the favorite controller is feasible, the MPC controller behaves as such. However, by this approach we almost completely cancel the performance optimization capabilities of MPC, since only the first input is accounted for in the cost, and the objective is only related to the favorite controller local approximation, even though constraints are still enforced along the whole prediction horizon. Furthermore, cost function (7) is not strongly related with closed-loop stability, but it shall be adjusted for this purpose. Let $P, Q \in \mathbb{R}^{n\times n}$ be matrices such that

$$(A+BK)'P(A+BK) - P \leq -(1+\alpha)Q,\ P>0,\ Q>0 \tag{8}$$

where $\alpha > 0$ is a fixed (small) scalar, so that $x(k)'Px(k)$ is a Lyapunov function for the system in closed-loop with the favorite controller. Modify cost function (7) into

$$\begin{aligned} J(x(k), U(k)) = 2x(k)'(A+BK)PB(u(0|k) - Kx(k)) + \\ (u(0|k) - Kx(k))'B'PB(u(0|k) - Kx(k)) \end{aligned} \tag{9}$$

Proposition 1: Let $U^*(k)$ be the solution at time $k \in \mathbb{Z}_{0+}$ of (1), where (1a) is replaced by (9). If there exists $\bar{k} \in \mathbb{Z}_{0+}$ such that for all $k \geq \bar{k}$ $J(U^*(k), x(k)) \leq x(k)'Qx(k)$, the system is asymptotically stable.

Proof: Let $d(k) = u^*(0|k) - Kx(k)$ and $A_{cl} = A + BK$. The evolution of the system in closed-loop with the MPC controller is $x(k+1) = A_{cl}x(k) + Bd(k)$. Since $x'(k+1)'Px(k+1) - x(k)'Px(k) = -x(k)'Px(k) + x(k)'A_{cl}'PA_{cl}x(k) + d(k)'B'PBd(k) + 2x(k)'A_{cl}'PBd(k)$, and by (8) $x'(k+1)'Px(k+1) - x(k)'Px(k) \leq -(1+\alpha)x(k)'Qx(k) + d(k)'B'PBd(k) + 2x(k)'A_{cl}PBd(k)$, hence for all $k \geq \bar{k}$, $x'(k+1)'Px(k+1) \leq -x(k)'Px(k) - (1+\alpha)x(k)'Qx(k) + J(U^*(k), x(k)) \leq -\alpha x(k)'Qx(k)$. Thus, P is a Lyapunov function for all $k \geq \bar{k}$, which proves asymptotic Lyapunov stability of the closed-loop system. ■

The condition in Proposition 1 is sufficient, but is not necessary. The global minimum of the cost function (9) can be smaller than zero, and it can be achieved for $u^*(0|k) \neq Kx(k)$, which means that when the constraints are not active the MPC controller does not behave as the favorite controller. In order to force the behavior required by Problem 1 we need to add to the MPC optimal control problem (1) the linear constraint[3]

$$x(k)'A_{cl}'PB(u(0|k) - Kx(k)) \geq 0. \tag{10}$$

It is immediate to prove that a lower bound to the optimum is $J(U^*(k), x(k)) = 0$, achieved when $d(k) = 0$, i.e., when $u^*(0|k) = Kx(k)$ is a feasible input. Thus, when all the original constraints of (1) are not active (while constraint (10) is in general active), the MPC behaves as (6).

The MPC-based controller obtained by (9) does not provide global stability guarantee. However, Proposition 1 ensures that even when constraints are active, the MPC controller tries to optimize the decrease of the Lyapunov function, and thus it tries at its best to stabilize the system.

The direct approaches for controller matching are based on the reformulation of MPC problem (1) as a tracking problem. Even though they are simple, global stability is hard to guarantee and they reduce optimization capabilities of MPC. Also, direct approaches only force MPC to track a reference signal, but they do not give provide any relation between MPC feedback law and (6). To address these limitations we investigate the capability of MPC to be locally functionally equivalent to the favorite controller. We aim at solving Problem 1 by selecting appropriate weight matrices in (1a).

IV. Matching Based on QP Matrices

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

$$-\Psi H^{-1}F'x(k) = Kx(k). \tag{11}$$

Unfortunately (11) is not trivial to solve, due to the non-invertibility of matrix Ψ and the way the inverse of H depends on Q, R, P.

To solve Problem 1, we start removing Ψ in (11) by setting

$$H^{-1}F'x(k) = -\begin{bmatrix} \kappa_0 \\ \kappa_1 \\ \vdots \\ \kappa_{N-1} \end{bmatrix} x(k), \tag{12}$$

where $\kappa_0 = K$, while $\kappa_i \in \mathbb{R}^{m\times n}$, $i \in \mathbb{Z}_{[1,N-1]}$ are free matrices. In (12) we account for the whole optimal input sequence of problem (1), but we enforce the equality with the favorite control action only for the first control action. As the remaining optimal control functions κ_i, $i = 1, \ldots, N-1$ are left as free variables, they do not introduce further constraints on the original problem.

Proposition 2: Let $(\tilde{\mathcal{K}}, \tilde{Q}, \tilde{R}, \tilde{P})$ be any feasible solution of the following problem

$$\begin{aligned} \min_{\mathcal{K},Q,R,P} \quad & J(\mathcal{K}, Q, R, P) && \text{(13a)} \\ \text{s.t.} \quad & Q \geq 0,\ P \geq 0,\ R \geq \sigma I && \text{(13b)} \\ & (\mathcal{R} + \mathcal{S}'\mathcal{Q}\mathcal{S})\mathcal{K} + \mathcal{S}'\mathcal{Q}\mathcal{T} = 0 && \text{(13c)} \\ & \kappa_0 = K && \text{(13d)} \end{aligned}$$

where $\mathcal{K} = [\kappa_0' \cdots \kappa_{N-1}']'$, and $J(\cdot)$ is any convex objective function. Then the MPC strategy based on the optimal

[3] At any control cycle in the MPC algorithm $x(k)$ is fixed. Hence constraint (10) is linear in the optimization variables.

control problem (1) where we set $Q = \tilde{Q}$, $P = \tilde{P}$, $R = \tilde{R}$, solves Problem 1.

Proof: Equality (13c) represents (12) after multiplying both sides by H, which is invertible. Constraint (13d) enforces the equality between the MPC command (5) and the favorite controller (6). Finally, (13b) ensures that the obtained matrices define a valid cost function for the MPC problem (1). Any solution satisfying (13b), (13c), (13d) results in (11) when the constraints in (1) are not active. ■

Problem (13) is nonconvex due to bilinear constraint (13c). Different cost functions (13a) can be used to define an optimal triplet of matrices (Q, R, P) among those which solves Problem 1. As an example, for a given reference triplet $(\bar{Q}, \bar{R}, \bar{P})$ set

$$J(\mathcal{K}, Q, R, P) = \|Q - \bar{Q}\| + w_R\|R - \bar{R}\| + w_P\|P - \bar{P}\| \quad (14)$$

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

In case it is impossible to find matrices Q, R and P that satisfy (13b), (13c), (13d), Problem 1 does not have an exact solution. Hence, we must accept a different behavior $u = \tilde{K}x$, where $\tilde{K} \neq K$. Problem (13) can be modified to return matrices Q^*, R^*, P^* so that $\tilde{K}$ approximates K. Consider

$$\min_{\mathcal{K},Q,R,P,V} \quad \|\mathcal{K} + V\mathcal{S}'\mathcal{Q}\mathcal{T}\| \quad (15a)$$

$$\text{s.t.} \quad Q \geq 0,\ P \geq 0,\ R \geq \sigma I,\ \kappa_0 = K \quad (15b)$$

$$V\mathcal{R} + V\mathcal{S}'\mathcal{Q}\mathcal{S} = I \quad (15c)$$

and let $\mathcal{K}^*$, Q^*, R^*, P^* be the optimizer of problem (15), and J^* be the corresponding optimal value. Problem 1 is exactly solved only if $J^* = 0$. Otherwise, the solution matrices Q^*, R^* and P^* define the MPC controller that, when constraints are nor active, corresponds to the static linear state-feedback controller that is closest to (6) in the matrix norm $\|\cdot\|$.

Problem (15) is also nonconvex due to the bilinear constraints (15c). Even though nowadays solvers exist for bilinear problems [6], the convergence to a solution is not guaranteed. In problems (13), (15), the main cause of bilinearity are the free variables κ_i, $i \in \mathbb{Z}_{[1,N-1]}$. In fact, these variables are free because only the first step of the optimal input sequence is applied by the MPC and must be equal to the command of the favorite controller (6).

In order to formulate problem (13) as a linear matrix inequality (LMI), one can fix the whole vector

$$\mathcal{K} = \bar{\mathcal{K}} \triangleq \begin{bmatrix} K \\ \bar{\kappa}_1 \\ \vdots \\ \bar{\kappa}_{N-1} \end{bmatrix}. \quad (16)$$

This obviously further constrains the design problem, because constraints are imposed on the whole sequence of command laws. In order to recover some degrees of freedom, the cost function in problem (13) can be made time-varying,

$$\mathcal{Q} = \begin{bmatrix} Q_1 & 0 & 0 & \dots & 0 \\ 0 & Q_2 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & Q_{N-1} & 0 \\ 0 & 0 & \dots & 0 & P \end{bmatrix}, \ \mathcal{R} = \begin{bmatrix} R_0 & 0 & \dots & 0 \\ 0 & R_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & R_{N-1} \end{bmatrix},$$

where $Q_i = Q_i' \geq 0$, for all $i \in \mathbb{Z}_{[1,N-1]}$, $R_i = R_i' \geq \sigma I$, for all $i \in \mathbb{Z}_{[0,N-1]}$. When $\mathcal{K}$ is fixed by (16), problem (13) straightforwardly becomes a problem with LMI constraints in the variables Q_i, $i \in \mathbb{Z}_{[1,N-1]}$, R_i, $i \in \mathbb{Z}_{[0,N-1]}$, and P. Also, a problem similar to (15) can be formulated that returns an MPC controller that approximates the favorite controller (6) when its precise behavior cannot be obtained.

Consider the problem

$$\min_{\mathcal{Q},\mathcal{R}} \quad \|(\mathcal{R} + \mathcal{S}'\mathcal{Q}\mathcal{S})\bar{\mathcal{K}} + \mathcal{S}'\mathcal{Q}\mathcal{T}\| \quad (17a)$$

$$\text{s.t.} \quad P \geq 0, R_i \geq \sigma I,\ i \in \mathbb{Z}_{[0,N-1]} \quad (17b)$$

$$Q_i \geq 0,\ i \in \mathbb{Z}_{[1,N-1]} \quad (17c)$$

which is a convex problem subject to LMI constraints, where we have set

$$\bar{\mathcal{K}} = \begin{bmatrix} K \\ K(A+BK) \\ \vdots \\ K(A+BK)^{N-1} \end{bmatrix}. \quad (18)$$

If the optimum is 0 for the optimizer Q_i^*, $i \in \mathbb{Z}_{[1,N-1]}$, R_i^*, $i \in \mathbb{Z}_{[0,N-1]}$, P^*, the MPC controller with time-varying cost function (1a) defined by $Q_i = Q_i^*$, $i \in \mathbb{Z}_{[1,N-1]}$, $R_i = R_i^*$, $i \in \mathbb{Z}_{[0,N-1]}$, $P = P^*$ and for any[4] $Q_0 \geq 0$ behaves as the favorite controller (6) when the constraints are not active. If the optimum is not 0, when constraints are not active the MPC behaves as the best approximation of the controller in the norm $\|\cdot\|$, weighted by the positive definite matrix H. Note that by (16) we require that the MPC behaves as the favorite controller along the whole prediction horizon, which is a tighter requirement than what is needed by Problem 1.

When the constraints are inactive, the MPC controller obtained as the solution of Problem 1 behaves as the favorite controller (6), which is reasonably assumed to yield asymptotically stable closed-loop dynamics. Thus, if the constraints are not active in a neighborhood of the origin, the closed-loop dynamics are locally asymptotically stable. The exact region of the state space where the matching occurs (i.e., where the constraints are inactive) 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) [7].

As for general MPC [3], global stability is more complicated to guarantee, especially when the local equivalence with the favorite controller has to be maintained. The approach described in [3] 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 PI set for (1b) in closed loop with (6) such that $\mathbf{0} \in \text{int}(\mathcal{X}_T)$ and $\mathcal{X}_T \subseteq \{x \in \mathbb{R}^n : x \in [x_{\min}, x_{\max}], K_{\text{fv}}x \in [u_{\min}, u_{\max}]\}$. Add constraint $x(N|k) \in \mathcal{X}_T$ to (1) with Q, R, P are computed by (13) (or by (17), with $Q_i = Q$, $R_i = R$, $i \in \mathbb{Z}_{[0,N-1]}$, and the optimum is $J^* = 0$), and 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 $x(k) = x$. Let $P \geq 0$ be such that

$$(A + BK)'P(A + BK) + K'RK + Q - P \leq 0. \quad (19)$$

[4]Since $x(0|k)$ is fixed in (1), the optimizer does not depend on Q_0.

Then (i) the closed-loop dynamics are asymptotically stable and remain in $\mathcal{X}_{\rm feas}$, for all $x(0) \in \mathcal{X}_{\rm feas}$; (ii) there exists a set $\mathcal{X}_{\rm fv} \supseteq \mathcal{X}_T$ such that the MPC behaves as the favorite controller; (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: Due to space limitations, the proof is omitted here and can be found in [8]. ■

In order to satisfy the assumption of Theorem 1, the LMI (19) is to be added to the cost design problems, e.g., (13) or (17), while $\mathcal{X}_T$ can be computed for instance as the maximum positively invariant set as described in [9].

V. Matching Based on Inverse LQR

In this section we propose an alternative to solve Problem 1, which is computationally simpler. Instead of solving (11), we use the following theorem [10].

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

$$\bar{P} = A'\bar{P}A - A'\bar{P}B(B'\bar{P}B + \bar{R})^{-1}B'\bar{P}A + \bar{Q}. \qquad (20)$$

Set $Q = \bar{Q}$, $R = \bar{R}$, and $P = \bar{P}$ in (1a). For any prediction horizon $N \in \mathbb{Z}_+$, when 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.

Proof: See e.g. [10] and the references therein. ■

Theorem 2 establishes a relation between the linear quadratic regulator (LQR) and the model predictive controller. In fact, under the hypothesis of Theorem 2, the MPC controller that has no (active) constraints behaves as the linear state feedback gain that optimizes the LQR cost $\min_{u(\cdot)} \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k)$ for the linear dynamics $x(k+1) = Ax(k) + Bu(k)$.

We look for weights Q, R and P such that the favorite controller (6) is the LQR gain with respect to them. In this way by exploiting Theorem 2, the MPC controller, which behaves as the LQR for any horizon $N \in \mathbb{Z}_+$, will be also equal to the favorite controller (6).

Corollary 1: Consider the optimization problem

$$\begin{aligned} \min_{Q,R,P} \quad & J(Q,R,P) && (21a)\\ \text{s.t.} \quad & P \geq 0,\ R \geq \sigma I,\ Q \geq 0 && (21b)\\ & P = A'PA + A'PBK + Q && (21c)\\ & B'PA = -(B'PB + R)K && (21d)\end{aligned}$$

where $J(\cdot)$ is a convex cost function of its arguments (e.g., as in (14)) and (21b), (21c), (21d) are linear matrix (in)equalities. Let $\tilde{Q}$, $\tilde{R}$, $\tilde{P}$ be any feasible solution (i.e., not necessarily the optimal one) of (21). 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 (21c), (21d), and constraint (21b) 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 cost function matrices. Thus, given any $\tilde{Q}$, $\tilde{R}$, $\tilde{P}$ that are a feasible solution of (21), 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 $Q = \tilde{Q}$, $P = \tilde{P}$, $R = \tilde{R}$ results in an MPC command (5) that is equal to the one of the favorite controller (6) whenever the constraints in the MPC quadratic program (2) are not active. ■

Cost function (21) can be used to define the reference performance criterion of the MPC problem that comes into play when the constraints are active, for instance as in (14). Also, Q can be removed from (21), by formulating (21c) as an inequality and evaluating Q from the obtained P and R.

Problem (21) is not guaranteed to be feasible, because it is not true that any $K \in \mathbb{R}^{n\times m}$ is the LQR gain for some choice of matrices $P \in \mathbb{R}^{n\times m}$, $Q \in \mathbb{R}^{n\times m}$, $R \in \mathbb{R}^{n\times m}$ that satisfies (21b).

The problem of checking whether a given controller is optimal with respect to some performance criterion (inverse optimality problem) was first introduced by Kalman [11], in the sixties. In the late eighties the inverse LQ design problem has been studied [12], where the conditions for a linear state-feedback controller to be an LQR gain were analyzed. The works [11]–[13] focused on continuous-time systems, and provided algebraic conditions for a given linear state feedback law to be an LQR. Equation (21) provides a way to solve the inverse LQR design for discrete-time systems by convex optimization.

Working with LQR gains introduces additional constraints but reduces problem complexity, since the size of (21), is not related to the MPC horizon N. As a result, problem (21) is significantly less complex than (15) when N is large. On the other hand because of Theorem 2, whenever (21) is feasible, a solution of (17) exists with zero cost, while the opposite is not guaranteed.

MPC cost based on (21) ensures that when the constraints are not active the MPC is equivalent to an LQR, and hence yields (locally) asymptotically stable closed-loop dynamics. Global stability in the set of feasible initial conditions in the presence of constraints can be achieved by choosing N large enough [7], [14], [15], since Corollary 1 is independent on the value of N. Also, Theorem 1 can be applied using the Riccati matrix P as terminal weight and $K_{\rm LQR}$ as auxiliary controller, since P computed from (21) satisfies (19).

VI. Examples

We present some examples of inverse matching techniques.

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

$$A = \begin{bmatrix} 0.675 & 0.923 & 0.014 \\ -0.315 & 0.215 & -0.750 \\ 1.05 & 0.00 & 1.50 \end{bmatrix},\ B = \begin{bmatrix} 0.5 \\ 0 \\ 1 \end{bmatrix},\ \begin{array}{l} x \in \mathbb{R}^3 \\ u \in \mathbb{R}, \end{array}$$

and favorite controller (6), where $K = \begin{bmatrix} -0.918 & 0.347 & -0.806 \end{bmatrix}$ is designed through pole-placement so that the closed-loop matrix $A + BK$ has eigenvalues 0.35, 0.375, 0.40. Assume that the input constraints $-1.5 \leq u \leq 1.5$ must be enforced.

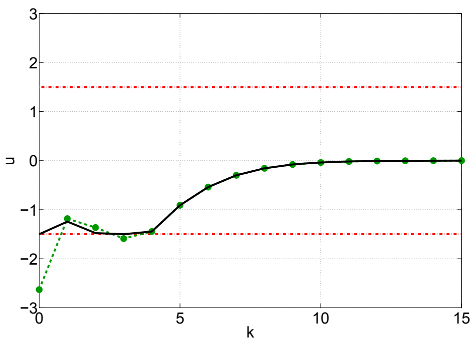

Fig. 1. Example 1, matching based on QP matrices. MPC command u_{MPC} compared to favorite controller command, $u_{\text{fv}} = Kx$, $K = [-0.918 \;\; 0.347 \;\; -0.806]$

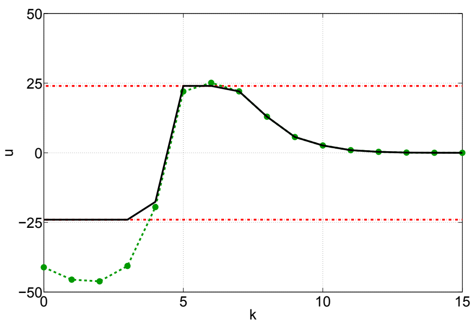

Fig. 2. Example 2, matching based on inverse LQR. MPC command u_{MPC} compared $u_{\text{fv}} = [-5.38 \;\; -2.84 \;\; -0.25 \;\; -2.37]x(k)$.

We solve problem (17), where we have set $N = 3$, $\sigma = 10^{-3}$, $\bar{\mathcal{K}}$ as in (18), and we have imposed $Q_i = Q \geq 0$, $\forall i \in \mathbb{Z}_{[1,N-1]}$, $R_i = R > \sigma I$, $\forall i \in \mathbb{Z}_{[0,N-1]}$. Problem (17) is solved with an optimal cost of 0, resulting in matrices Q^*, R^*, P^*. Thus, when $Q = Q^*$, $R = R^*$, $P = P^*$ in (1), the MPC command u_{MPC} equals u_{fv} whenever the constraints are not active. The simulation of the closed-loop system from initial state $x(0)' = [\,0\;5\;5\,]$ is shown in Figure 1, where the solid line is the MPC command applied to the system and the dashed line with circle markers is the hypothetical command that would be issued by the favorite controller for the same state and that does not account for constraints. Note that in order to have $u_{\text{MPC}} = u_{\text{fv}}$ the constraints need not to be active along the whole MPC horizon $N = 3$.

Example 2 (Matching based on inverse LQR): Consider the linear system $x(k+1) = Ax(k) + Bu(k)$, where

$$A = \begin{bmatrix} 1.8 & 1.2 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 3.6 & 2.4 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad \begin{array}{l} x \in \mathbb{R}^4 \\ u \in \mathbb{R}, \end{array}$$

and constrained input $-24 \leq u \leq 24$, and the favorite controller (6), $K = [\,-5.38\;-2.84\;-0.25\;-2.37\,]$. The inverse LQR-based matching problem (21) where $\sigma = 10^{-3}$, and with objective function (14), where $\bar{Q} = 6.5I$, $\bar{R} = 1$, $w_R = 1$, $w_P = 0$ was solved obtaining Q^*, R^*, P^*. We have implemented the MPC strategy (1) with $N = 3$ and $Q = Q^*$, $R = R^*$, $P = P^*$ so that $u_{\text{MPC}} = u_{\text{fv}}$ whenever the constraints are not active. The simulation of the closed-loop system from initial state $x(0)' = [\,5\;5\;0\;0\,]$ is shown in Figure 2, where the solid line is the MPC command and the dashed line with circle markers is the input that would be applied by the favorite controller.

VII. Conclusions

To exploit both the frequency-domain properties of a given linear controller for signals small enough not to activate constraints, and the ability of model predictive control to handle constraints, in this paper we provided criteria to convert the existing linear control law into an MPC controller. The approach can be seen as a way of automatically generating an anti-windup scheme, which is a piecewise-affine function in case the resulting MPC controller is considered into its explicit form [7]. Finally, we have shown how to obtain global stability of the MPC scheme in the set of feasible initial conditions, by adequately adapting the MPC stabilization results to preserve the controller matching objective.

References

[1] K. Åstrom and L. Rundqwist, "Integrator windup and how to avoid it," in *American Control Conference*, 1989, pp. 1693–1698.
[2] J. Maciejowski, *Predictive control with constraints.* Englewood Cliffs, NJ: Prentice Hall., 2002.
[3] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, June 2000.
[4] S. Qin and T. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 93, no. 316, pp. 733–764, 2003.
[5] M. Bares, M. Diehl, and I. Necoara, "Every continuous nonlinear control system can be obtained by parametric convex programming," *IEEE Trans. Automatic Control*, vol. 53, no. 8, pp. 1963–1967, 2008.
[6] M. Kocvara and M. Stingl, *PENBMI User's Guide*, Tomlab Optimization, 2003, latest version available at http://www.penopt.com/doc/penbmi2_1.pdf.
[7] 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.
[8] S. Di Cairano and A. Bemporad, "Model predictive controller tuning by controller matching," *IEEE Trans. Automatic Control*, 2009, submitted.
[9] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical Problems in Engineering*, vol. 4, pp. 317–367, 1998.
[10] P. Scokaert and J. Rawlings, "Constrained linear quadratic regulation," *IEEE Trans. Automatic Control*, vol. 43, no. 8, pp. 1163–1169, 1998.
[11] R. Kalman, "When is a Linear Control System Optimal?" *Trans. ASME, Jour. of Basic En.*, pp. 51–60, 1964.
[12] T. Fujii, "A new approach to the LQ design from the viewpoint of the inverse regulator problem," *IEEE Trans. Automatic Control*, vol. 32, no. 11, pp. 995–1004, 1987.
[13] K. Sugimoto, "Partial pole placement by LQ regulators: an inverse problem approach," *IEEE Trans. Automatic Control*, vol. 43, no. 5, pp. 706–708, 1998.
[14] D. Chmielewski and V. Manousiouthakis, "On constrained infinite-time linear quadratic optimal control," *Systems & Control Letters*, vol. 29, no. 3, pp. 121–130, Nov. 1996.
[15] J. Rawlings and K. Muske, "The stability of constrained receding-horizon control," *IEEE Trans. Automatic Control*, vol. 38, pp. 1512–1516, 1993.