

Solving Constrained LQR Problems by Eliminating the Inputs from the QP

Giulio M. Mancuso and Eric C. Kerrigan

Abstract—In this paper a new approach to formulate the constrained Linear Quadratic Regulator (LQR) problem as a Quadratic Programming (QP) problem is introduced. The new approach takes advantage of the (Moore-Penrose) generalized inverse to eliminate control inputs as decision variables, hence the optimization is performed only over the states belonging to the prediction horizon. This allows one to save on computation if an interior point method is used to solve the QP problem compared to using existing formulations, where the optimization is done over the states and inputs.

I. INTRODUCTION

Model predictive control (MPC), has become a very successful control policy. This success is given by the fact that it is an optimal control policy that takes into account physical constraints of the real system to control [1]. At each sampling instant MPC solves a constrained LQR problem. After the optimization process, only the first optimal control value is updated in to the system. Although the MPC allows great performance on control real systems, it require lots of computation to calculate the opportune control action, since a considerable optimization problem has to be solved at each sampling instant.

In practice, the constrained LQR problem is commonly formulated as a quadratic programming (QP) problem. There are different ways to formulate the QP starting from the same LQR problem. Substantially all the approaches in the literature use as decision variable the control inputs (*condensed formulation*) [1]–[3] or both inputs and states (*non-condensed formulation*) [1]–[3].

There are several work that point out how different formulation are suitable for some kind of systems respect to other in terms of computational complexity and conditioning of the QP problem [2], [4].

In this work we present a new approach to formulate the QP in terms of only the states. Our formulation has been derived by removing the control inputs and taking advantage of the generalized inverse of the input matrix B , when the linear system is described by $x_{k+1} = Ax_k + Bu_k$ and B is full column rank.

For long prediction horizons, it has been shown [4] that, in terms of computational complexity, the non-condensed formulation with both inputs and states allows better performance compared to the condensed formulated with the inputs only.

G. M. Mancuso is with Scuola Superiore Sant’Anna, Italy, (e-mail: g.mancuso@sssup.it).

E. C. Kerrigan is with the Department of Aeronautics and Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: e.kerrigan@imperial.ac.uk).

978-1-61284-799-3/11/\$26.00 ©2011 IEEE

We will demonstrate that the new approach leads to a sparse and symmetric formulation of the QP similar to the non-condensed formulation. Moreover, we will show that, for long horizons, the new approach allows one to save computation when an interior point algorithm is used to solve the QP, compared to the non-condensed approach.

The paper is organized as follows. Section 2 gives a statement of the problem and discusses some existing results. Section 3 introduces some basics of the Moore-Penrose generalized inversed that will be useful for our new formulation. The new approach with only the states as decision variables is fully discussed in Sections 4 and 5. Finally, Section 6 shows the advantages from the computational point of view when using our approach in combination with an interior point method.

II. LINEAR QUADRATIC MPC PROBLEM

The optimal control problem to solve is

$$\min_{\substack{u_0, \dots, u_{N-1} \\ x_0, \dots, x_N}} V(\cdot) := \frac{1}{2} x'_N P x_N + \frac{1}{2} \sum_{k=0}^{N-1} (x'_k Q x_k + u'_k R u_k) \quad (1)$$

subject to:

$$\begin{aligned} \hat{x} &= x_0 \\ x_{k+1} &= Ax_k + Bu_k, \quad \forall k \in \mathcal{I} \\ y_k &= Cx_k, \quad \forall k \in \mathcal{I} \\ u_{\min} &\leq u_k \leq u_{\max} \quad \forall k \in \mathcal{I} \\ y_{\min} &\leq y_k \leq y_{\max} \quad \forall k \in \mathcal{I} \cup \{N\} \end{aligned}$$

where $\mathcal{I} := \{i \in \mathbb{N} | i = 0, 1, \dots, N-1\}$, $P \geq 0, Q \geq 0$, $R > 0$ and \hat{x} is an estimate or measurement of the current state. The cost to minimize is a function of the states and inputs over the horizon with length N . The system matrices are $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{p \times n}$. From now on we will assume that the matrix B is full column rank, with $n \geq m$, i.e. the number of states is greater than or equal to the number of inputs.

A. Existing Results

In practice, the optimal control problem (1) is translated in a *quadratic programming* problem of the form

$$\begin{aligned} \min_{\theta} \quad & V(\theta) := \frac{1}{2} \theta' H \theta + \theta' h \\ \text{subject to} \quad & F \theta = f \\ & G \theta \leq g \end{aligned} \quad (2)$$

There are two main approaches used to convert the optimal control problem into a QP problem. Even if they solve the

same problem they differ in many aspects, such as the size of the problem, conditioning, etc. We are going to call the two approaches *non-condensed* and *condensed*.

- 1) In the *condensed* form the decision variables are only the inputs, i.e. $\theta = [u'_0 \ u'_1 \ \dots \ u'_{N-1}]'$ and the final QP problem will have no equality constraints. The corresponding problem to solve will be of the form

$$\begin{aligned} \min_{\theta} \quad & V(\theta) = \frac{1}{2}\theta' H \theta + \theta' D \hat{x} \\ \text{subject to} \quad & G \theta \leq g(\hat{x}) \end{aligned}$$

- 2) In the *non-condensed* form the decision variables are both the inputs and the states, i.e. $\theta = [x'_0 \ u'_0 \ x'_1 \ u'_1 \ x'_2 \ \dots \ u'_{N-1} \ x'_N]'$. The corresponding problem to solve will be of the form

$$\begin{aligned} \min_{\theta} \quad & V(\theta) = \frac{1}{2}\theta' H \theta \\ \text{subject to} \quad & F \theta = f(\hat{x}) \\ & G \theta \leq g \end{aligned} \quad (3)$$

The dimensions of the matrices resulting from the two approaches are reported in Table

III. BRIEF OVERVIEW OF THE (MOORE-PENROSE) GENERALIZED INVERSE

To introduce the new approach based only on the states, we need to briefly review some basic concepts on the (Moore-Penrose) generalized inverse [5].

Definition 1: Let $Z \in \mathbb{F}^{n \times m}$ (with \mathbb{F} we denote either \mathbb{C} or \mathbb{R}). If Z is nonzero, by singular value decomposition, there exist orthogonal matrices $S_1 \in \mathbb{F}^{n \times n}$ and $S_2 \in \mathbb{F}^{m \times m}$ such that

$$Z = S_1 \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} S_2, \quad (4)$$

where $D := \text{diag}[\sigma_1(A), \sigma_2(A), \dots, \sigma_r(A)]$, $r := \text{rank}(Z)$, and $\sigma_1(A) \geq \sigma_2(A) \geq \dots \geq \sigma_r(A) \geq 0$ are the singular values of Z . The (Moore-Penrose) generalized inverse Z^+ of Z is the $m \times n$ matrix

$$Z^+ = S_2^* \begin{bmatrix} D^{-1} & 0 \\ 0 & 0 \end{bmatrix} S_1^*, \quad (5)$$

where S^* denote the conjugate transpose of S .

A property that will be useful later is that $ZZ^+Z = Z$, which can be verified directly from the definition. Note that the previous property does not imply that $ZZ^+ = Z^+Z = I$. The following statements also hold:

$$\mathcal{R}(Z) = \mathcal{N}(I - ZZ^+), \quad (6a)$$

$$\mathcal{N}(Z') = \mathcal{R}(I - ZZ^+), \quad (6b)$$

where $\mathcal{R}(Z)$ and $\mathcal{N}(Z')$ denote, respectively, the range of Z and the null space of its transpose. Given the linear system $Zx = b$ the following statements are equivalent:

- 1) There exists a vector $x \in \mathbb{F}^m$ satisfying $Zx = b$
- 2) $\text{rank } Z = \text{rank}[Z \ b]$
- 3) $ZZ^+b = b$

To prove the equivalences we can note that

$$b = Zx = \underbrace{(ZZ^+Z)}_Z x = ZZ^+b = Z \underbrace{Z^+b}_x \quad (7)$$

and the value of x will consequently be

$$x = Z^+b. \quad (8)$$

Moreover, if Z is full column rank, the solution of the linear system, if it exists, is unique and given by (8).

IV. DYNAMIC CONSTRAINT AND STATE FORMULATION

In the MPC problem the dynamic constraint has a double purpose. First, it denotes a constraint between the input u_k and the next state x_{k+1} . Second, it denotes a relation between the current state x_k and the next state state x_{k+1} . The latter relation comes up, in general, from the impossibility of the dynamic system to move the states from an initial state to any other state in the state space in only one step, even if the system is reachable. A discrete-time system needs at most n steps to reach all the space. All those observations come out from the definition of reachability for discrete-time systems. Now we want to explicitly derive both relations separately.

Proposition 1: If the triple (x_{k+1}, x_k, u_k) is feasible, i.e.

$$x_{k+1} = Ax_k + Bu_k, \quad (9)$$

and the matrix B satisfies

$$\text{rank } B = m, \text{ where } B \in \mathbb{R}^{n \times m} \text{ with } n \geq m, \quad (10)$$

then the dynamic constraint $x_{k+1} = Ax_k + Bu_k$ is equivalent to

$$u_k = B^+(x_{k+1} - Ax_k) \quad (11)$$

$$(I - BB^+)(x_{k+1} - Ax_k) = 0. \quad (12)$$

The above equations, once we have fixed x_k , represent the relations between u_k and x_{k+1} .

Note that (11) can be found by observing that the dynamic constraint can be written as

$$\overbrace{(x_{k+1} - Ax_k)}^{\mathcal{X}_k} = Bu_k, \quad (13)$$

where $\mathcal{X}_k \in \mathbb{R}^n$ is a vector. This equation can be interpreted as a linear system of equations. If the triple (x_{k+1}, x_k, u_k) is feasible, i.e. $x_{k+1} = Ax_k + Bu_k$, and B is full column rank, the input u_k exists, is unique, and given by (11). If x_k is fixed, the last equation gives us an injective relation between the input u_k and the next state x_{k+1} .

Under the assumption stated in Proposition 1, (12) can be derived by replacing (11) in the dynamic equation (9).

If B is full rank and $n = m$, then (12) is always true ($BB^+ = I$), which means that it can be removed as a constraint, since the system is reachable in one step, i.e. there always exists an input u_k that drives the system from an initial point x_k to any point in \mathbb{R}^n .

It has to be noted that if B is full row rank and $n < m$ (this means that we have more inputs than states), the input is not necessary uniquely determined by fixing x_k and x_{k+1} . In this case (11) will represent the minimum-norm input that drives the system from x_k to x_{k+1} . If B is full row rank the current approach cannot be applied, because of the non-uniqueness of the solution to (13).

V. FROM (1) TO A QUADRATIC PROGRAM: A PSEUDOINVERSE APPROACH

In this section we are going to show how the use of the generalized inverse allows us to formulate the constrained LQR problem as a function of only the states over the prediction horizon.

A. Cost function

The quadratic objective function (1) can be expressed as the sum of N terms, i.e. $\ell_k := x_k' Q x_k + u_k' R u_k$.

Each term can be rewritten, taking advantage of (11), as

$$\begin{aligned} \ell_k &= x_k' Q x_k + u_k' R u_k = \\ &= x_k' Q x_k + (x_{k+1} - A x_k)' \overbrace{(B^+)' R B^+}^{V_R} (x_{k+1} - A x_k) = \\ &= \begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix}' \begin{bmatrix} Q + A' V_R A & (-A' V_R) \\ (-A' V_R)' & V_R \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix} \end{aligned} \quad (14)$$

For easy readability we define

$$\begin{aligned} V_R &:= (B^+)' R B^+, \quad V_Q := Q + A' V_R A, \quad V_M := -A' V_R, \\ \ell_k &= \begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix}' \begin{bmatrix} V_Q & V_M \\ V_M' & V_R \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix}. \end{aligned}$$

If we define the vector

$$\mathbf{x} := [x_0' \quad x_1' \quad \dots \quad x_{N-1}' \quad x_N']', \quad (15)$$

then the overall cost function can be written as

$$V(\mathbf{x}) = \frac{1}{2} \mathbf{x}' H \mathbf{x}, \quad (16)$$

where

$$H := \begin{bmatrix} V_Q & V_M & & & \\ V_M' & V_Q + V_R & V_M & & \\ & V_M' & \ddots & V_M & \\ & & & V_M' & V_M \\ & & & V_M' & V_R + P \end{bmatrix}$$

and $H \in \mathbb{R}^{(N+1)n \times (N+1)n}$ has half bandwidth $2n$. The matrix H is symmetric if matrices Q , R and P are symmetric. The convexity of (16) is a direct consequence of the convexity of (1).

B. Equality Constraints

Since (11) has been taken into account in the cost function (16), the only remaining equality constraint will be (12), which that can be written as

$$\begin{bmatrix} -(I - BB^+)A & (I - BB^+) \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix} = 0. \quad (17)$$

From now on we will refer to some properties of the linear system (17) as properties of the matrix $\begin{bmatrix} -(I - BB^+)A & (I - BB^+) \end{bmatrix} \in \mathbb{R}^{n \times 2n}$.

It is easy to verify that $\text{rank}(\begin{bmatrix} -(I - BB^+)A & (I - BB^+) \end{bmatrix}) = \text{rank}([I - BB^+])$. It has to be noted that if (10) holds, the square matrix $(I - BB^+) \in \mathbb{R}^{n \times n}$ does not have full rank; this can be verified from the property of the pseudoinverse (6a), which tells us that the null space of

$(I - BB^+)$ is not empty and equal to the range of B . This implies that (17) does not have full rank.

To find the value of $\text{rank}(I - BB^+)$ we can use the *rank-nullity theorem* [6, p. 199], which in our case, if (10) holds, i.e. $\text{rank}(B) = m = \dim(\mathcal{R}(B))$, tells us that $\dim(\mathcal{N}(B^+)) = n - m$. From the property (6b) $\dim(\mathcal{N}(B^+)) = n - m = \text{rank}(I - BB^+)$. To remove the redundancy different methods can be applied, like singular value decomposition, QR factorization, etc. [7]. We rewrite the equality constraint (17) without redundancy as

$$\begin{bmatrix} \hat{\mathfrak{B}} & \mathfrak{B} \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix} = 0,$$

where with $\begin{bmatrix} \hat{\mathfrak{B}} & \mathfrak{B} \end{bmatrix} \in \mathbb{R}^{(n-m) \times 2n}$ we denote the matrix $\begin{bmatrix} -(I - BB^+)A & (I - BB^+) \end{bmatrix}$ without redundancy. In terms of the unknown vector \mathbf{x} , the equality constraints can be written as

$$F \mathbf{x} = f(\hat{\mathbf{x}}), \quad (18)$$

where

$$F := \begin{bmatrix} I & & & \\ \hat{\mathfrak{B}} & \mathfrak{B} & & \\ & & \ddots & \\ & & & \hat{\mathfrak{B}} & \mathfrak{B} \end{bmatrix}, \quad f \hat{\mathbf{x}} := \begin{bmatrix} \hat{\mathbf{x}} \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$F \in \mathbb{R}^{(N+1)n \times N(n-m)+n}$, $f \in \mathbb{R}^{(N+1)n}$ and F has bandwidth $2n$.

C. Inequality Constraints

The inequality constraints are

$$\begin{aligned} u_{\min} &\leq u_i \leq u_{\max} \quad \forall k \in \mathcal{I}, \\ y_{\min} &\leq y_i \leq y_{\max} \quad \forall k \in \mathcal{I} \cup \{N\}, \end{aligned}$$

that, taking advantage of (11), the constraint on u_k can be translated into a constraint between x_k and x_{k+1} :

$$\begin{aligned} u_k &= B^+(x_{k+1} - A x_k) \leq u_{\max} \quad \forall k \in \mathcal{I} \\ -u_k &= -B^+(x_{k+1} - A x_k) \leq -u_{\min} \quad \forall k \in \mathcal{I}. \end{aligned}$$

All the inequality constraints can be written in a single matrix form

$$G \mathbf{x} \leq g, \quad (19)$$

where

$$G := \begin{bmatrix} -B^+A & B^+ & & & \\ B^+A & -B^+ & & & \\ C & 0 & & & \\ -C & 0 & & & \\ & & \ddots & & \\ & -B^+A & B^+ & & \\ & B^+A & -B^+ & & \\ & C & 0 & & \\ & -C & 0 & & \\ & & C & & \\ & & -C & & \end{bmatrix}, \quad g := \begin{bmatrix} u_{\max} \\ -u_{\min} \\ y_{\max} \\ -y_{\min} \\ \vdots \\ u_{\max} \\ -u_{\min} \\ y_{\max} \\ -y_{\min} \\ y_{\max} \\ -y_{\min} \end{bmatrix}$$

with $G \in \mathbb{R}^{2(p+m)N+2p \times (N+1)n}$ having bandwidth $2n$, and $g \in \mathbb{R}^{2(p+m)N+2n}$.

VI. QP SOLUTION VIA AN INTERIOR POINT METHOD

Different methods can be used to compute the solution of a QP problem. In practice, the most used algorithms that offer the best performance are: active set methods and interior point methods. Sometimes interior point methods are preferable because they possess polynomial computational complexity and it is an algorithm where the matrices have a fixed structure. A well stated reference in the computation of MPC with both algorithms is [1]. In this section we will focus on primal-dual interior point methods with an infeasible starting point [8].

It is common to pose the general QP problem in a standard form:

$$\begin{aligned} \min_{\theta} \quad & \frac{1}{2}\theta^T H\theta + \theta^T h \\ \text{subject to} \quad & F\theta = f \\ & G\theta \leq g \end{aligned} \quad (20)$$

To solve problem (20) we can apply the Karush-Kuhn-Tucker necessary conditions that, in the case of a convex QP, give us both necessary and sufficient conditions.

Let us define the Lagrangian

$$\mathcal{L}(\theta, \nu, \lambda) := \frac{1}{2}\theta^T H\theta + h^T \theta + \nu^T (F\theta - f) + \lambda^T (G\theta - g), \quad (21)$$

where ν and λ are the Lagrangian multiplier with opportune dimensions. A primal-dual solution $(\theta^*, \nu^*, \lambda^*)$ of problem (20) satisfies the equations

$$H\theta^* + F^T \nu^* + G^T \lambda^* + h = 0 \quad (22a)$$

$$F\theta^* - f = 0 \quad (22b)$$

$$G\theta^* + s^* - g = 0 \quad (22c)$$

$$\lambda_i^* s_i^* = 0 \quad \forall i \in \mathcal{I} \cup \{N\} \quad (22d)$$

$$(\lambda^*, s^*) \geq 0 \quad (22e)$$

where s^* is a *slack variable* to convert the (22d) into an equality constraint, and θ^* is the solution of the main problem (2). The solution of the optimization problem can be found by finding a root of the set of nonlinear equations

$$F(\theta, \nu, \lambda, s) = \begin{bmatrix} H\theta + F^T \nu + G^T \lambda + h \\ F\theta - f \\ G\theta + s - g \\ \Lambda S e \end{bmatrix} \quad (23)$$

$$(\lambda, s) \geq 0$$

where $\Lambda := \text{diag}(\lambda_1; \lambda_2; \dots)$, $S := \text{diag}(s_1; s_2; \dots)$ and $e := [1 \ 1 \ \dots \ 1]^T$.

The primal-dual interior method is an iterative algorithm that takes advantage of a variant of Newton's method for finding a primal-dual solution $(\theta^*, \nu^*, \lambda^*, s^*)$ of the conditions (22). It is beyond the scope of this paper to give a complete introduction to interior point methods; a good reference that covers the topic is [8]. Here we just introduce basic concepts linked to the Newton method used in the optimization algorithm, emphasizing the computational aspects.

A. Newton Method for IPM

As we said before, the search direction procedure has its origins in Newton's method for the nonlinear equations (23). Newton's method, at each iteration i , forms a linear model of $F(\cdot)$ around the current point and obtains the search direction $(\Delta\theta^i, \Delta\nu^i, \Delta\lambda^i, \Delta s^i)$ by solving, in the case of an infeasible starting point [8], the following system of linear equations:

$$J(\theta^i, \nu^i, \lambda^i, s^i) \begin{bmatrix} \Delta\theta^i \\ \Delta\nu^i \\ \Delta\lambda^i \\ \Delta s^i \end{bmatrix} = - \begin{bmatrix} r_H^i \\ r_F^i \\ r_G^i \\ r_S^i \end{bmatrix},$$

where $J(\cdot)$ is the Jacobian of $F(\cdot)$ and the vector of residuals $\begin{bmatrix} r_H^i & r_F^i & r_G^i & r_S^i \end{bmatrix}'$ is defined as

$$\begin{bmatrix} r_H^i \\ r_F^i \\ r_G^i \\ r_S^i \end{bmatrix} := \begin{bmatrix} H\theta^i + F^T \nu^i + G^T \lambda^i + h \\ F\theta^i - f \\ G\theta^i + s^i - g \\ \Lambda^i S^i e \end{bmatrix}.$$

The next feasible point will be

$$\begin{bmatrix} \theta^{i+1} \\ \nu^{i+1} \\ \lambda^{i+1} \\ s^{i+1} \end{bmatrix} = \begin{bmatrix} \theta^i \\ \nu^i \\ \lambda^i \\ s^i \end{bmatrix} + \alpha \begin{bmatrix} \Delta\theta^i \\ \Delta\nu^i \\ \Delta\lambda^i \\ \Delta s^i \end{bmatrix}, \quad (24)$$

where $\alpha \in \mathbb{R}_{[0 \ 1]} := \{\alpha \in \mathbb{R} \mid 0 \leq \alpha \leq 1\}$ is an opportune step size. If the current point is strictly feasible, the Newton step equations become (for easy readability, from now on, we omit the index i)

$$\begin{bmatrix} H & F^T & G^T & 0 \\ F & 0 & 0 & 0 \\ G & 0 & 0 & I \\ 0 & 0 & S & \Lambda \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\nu \\ \Delta\lambda \\ \Delta s \end{bmatrix} = - \begin{bmatrix} r_H \\ r_F \\ r_G \\ r_S \end{bmatrix}. \quad (25)$$

We now perform block elimination twice to reduce the size of the linear system. The final system to solve is

$$\begin{bmatrix} H + G^T W(\lambda, s) G & F^T \\ F & 0 \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\nu \end{bmatrix} = \begin{bmatrix} r^\theta \\ r^\nu \end{bmatrix}, \quad (26)$$

where $W(\lambda, s) := \text{diag}(\lambda_1/s_1, \lambda_2/s_2, \dots)$ is a diagonal matrix with entries λ/s , and we have defined

$$\begin{bmatrix} r^\theta \\ r^\nu \end{bmatrix} := - \begin{bmatrix} -r_H + G^T W(G\theta - g) \\ r_F \end{bmatrix}. \quad (27)$$

The remaining vector can be found from

$$\begin{aligned} \Delta s &= \Lambda^{-1}(r_S - S\Delta\lambda) \\ \Delta\lambda &= S^{-1}\Lambda(r_G + \Lambda^{-1}r_S + G\Delta x). \end{aligned}$$

For easy readability we define

$$\mathcal{A} := \begin{bmatrix} H + G^T W(\lambda, s) G & F^T \\ F & 0 \end{bmatrix} \quad \mathcal{X} := \begin{bmatrix} \Delta\theta \\ \Delta\nu \end{bmatrix} \quad \mathcal{B} := \begin{bmatrix} r^\theta \\ r^\nu \end{bmatrix} \quad (28)$$

so, at each Newton iteration we have to compute the matrices $\mathcal{A}^i, \mathcal{B}^i$ and find the solution of the linear system:

$$\mathcal{A}^i \mathcal{X}^i = \mathcal{B}^i. \quad (29)$$

This procedure is the most time consuming part of an interior point method.

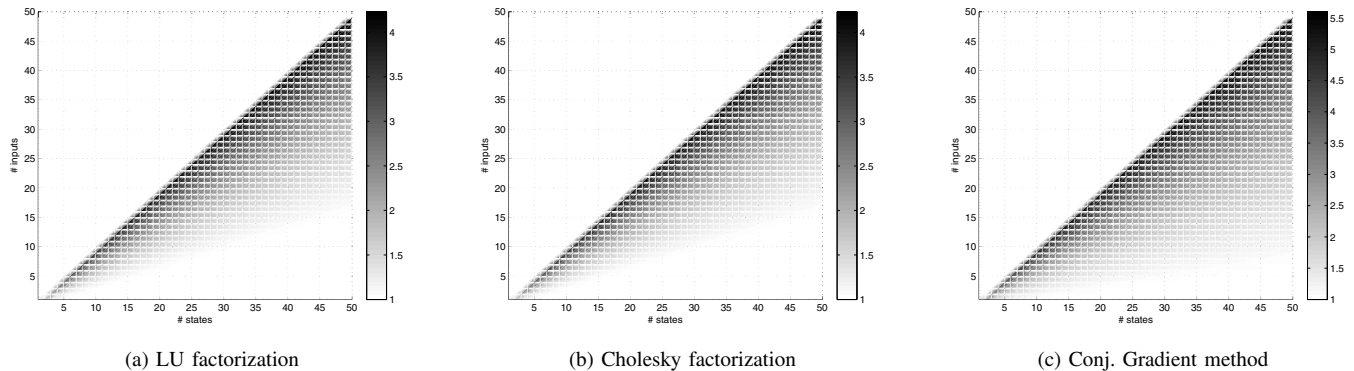


Fig. 1: Speedup between Non-Condensed and Pseudoinverse flops.

the corresponding flops are

$$\begin{aligned}
 \text{LU factorization} & F_{LU} = 4N_l B_l^2 + 6N_l B_l \\
 \text{Cholesky factorization} & F_{Ch} = 4N_l B_l^2 + 4N_l B_l \\
 \text{LDL' factorization} & F_{LDL} = 4N_l B_l^2 + 6N_l B_l \\
 \text{Conj. Gradient method} & F_{CG} = 2B_l N_l^2
 \end{aligned}$$

In Figure 1 are shown the speedups, defined as

$$\text{speedup} := \frac{\text{number flops}_{\text{non-condensed-approach}}}{\text{number flops}_{\text{new-approach}}}, \quad (32)$$

between the non-condensed and pseudoinverse approaches when $N = 30$. In the figures the dark shade represent the values of number of inputs and states in which the new approach performs better then the non-condensed approach. Darker means more speedup. As we can see the non-condensed method is better when the difference between the number of states and number of inputs is very big i.e. $n \gg m$. A suitable application where this approach could perform better can be found in aeronautics, indeed aeroplanes have many control inputs and relatively few states compared to the number of actuators.

VII. CONCLUSION AND FUTURE WORK

In the current paper the authors have introduced a new approach, based on the generalized inverse of the input matrix B . This formulation allows one to find the MPC solution as the solution of a Quadratic Programming problem, where the variable to optimize is only the state trajectory over the prediction horizon. In this paper we have shown that we can save on computational effort when using an interior point method, compared to the existing approaches in the literature. We have demonstrated the improvement in terms of flops with different linear solvers. However, there exist still some cases where $n \gg m$, in which the non-condensed approach has better performance.

We believe that, from the computational point of view, a formulation in only the states as decision variables could have many advantages. However there are many open issues and possible extensions, for example, how one should formulate a soft-constrained problem in this framework.

Warm-starting is another important issue, since the decision variables are linked to each other by the dynamics of the linear system.

From a numerical point of view a better analysis of the conditioning of the problem is needed to assure stability of the optimization algorithm.

ACKNOWLEDGMENTS

Giulio M. Mancuso would like to thank Imperial Control London and all the staff of the Department of Electrical and Electronic Engineering for hosting him during this work. He would also like to thank the grant Erasmus Consortia Placement for the financial support during his staying at Imperial College London.

REFERENCES

- [1] J. M. Maciejowski, *Predictive Control with Constraints*. Prentice Hall, 2000.
- [2] A. Shahzad, "New optimization methods in predictive control," Ph.D. dissertation, Imperial College London, 2010.
- [3] A. G. Wills and W. P. Heath, "Interior-point methods for linear model predictive control," in *Proceedings of the Control 2004 Conference, University of Bath*, D. M. N. Sahinkaya and K. A. Edge, Eds., 2004.
- [4] C. Rao, S. J. Wright, and J. B. Rawlings, "Application of interior-point methods to model predictive control," *Journal of Optimization Theory and Applications*, 1998.
- [5] D. S. Bernstein, *Matrix Mathematics: Theory, Facts, and Formulas with Application to Linear Systems Theory*. Princeton University Press, 2005.
- [6] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, ser. SIAM, SIAM, Ed. SIAM, 2000.
- [7] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2000.
- [8] S. J. Wright, *Primal-Dual Interior-Point Methods*. SIAM, 1987.
- [9] —, "Interior point methods for optimal control of discrete-time systems," *Journal of Optimization Theory and Applications*, 1993.
- [10] —, "Solution of discrete-time optimal control problems on parallel computers," *Parallel Computing*, 1990.
- [11] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.