Solving the infinite-horizon constrained LQR problem using splitting techniques

Giorgos Stathopoulos* Milan Korda* Colin N. Jones*

* *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, Station 9, CH-1015, Lausanne, Switzerland; email:* {*georgios.stathopoulos,milan.korda,colin.jones*}*@epfl.ch*

Abstract: This paper presents a method to solve the constrained *infinite-time* linear quadratic regulator (LQR) problem. We use an operator splitting technique, namely the alternating minimization algorithm (AMA), to split the problem into an *unconstrained* LQR problem and a projection step, which are solved repeatedly, with the solution of one influencing the other. The first step amounts to the solution of a system of linear equations (with the possibility to pre-factor) and the second step is a simple clipping. Therefore, each step can be carried out efficiently. The scheme is proven to converge to the solution to the infinite-time constrained LQR problem and is illustrated by numerical examples.

Keywords: Constrained LQR, Alternating minimization, Operator splitting

1. INTRODUCTION

An important extension of the famous result of Kalman (1960) on the closed form solution of the infinite-horizon linear quadratic regulator (LQR) problem is the case where the input and state variables are constrained. This problem is computationally significantly more difficult and has been by and large addressed only approximately. A prime example of an approximation scheme is model predictive control (MPC) which approximates the infinite-time constrained problem by a finite-time one. Stability of such MPC controllers is then typically enforced by adding a suitable terminal constraint and a terminal penalty. The inclusion of a terminal constraint limits the feasible region of the MPC, and, consequently, the region of attraction of the closed-loop system. In practical applications, this problem is typically overcome by simply choosing a "sufficiently" long horizon based on the process insight (e.g., dominant time constant). Closed-loop behavior is then analyzed a posteriori, for instance by exhaustive simulation or by investigating the set of optimality conditions of the underlying optimization problem Primbs (2001).

There have been few results addressing directly the infinite-horizon constrained LQR problem. The most well-known effort is the work of Scokaert and Rawlings (1998), where they extend the work of Sznaier and Damborg (1987). The idea is to solve a sequence of quadratic programs (QPs) of finite horizon length, which is monotonically non-decreasing. After each QP has been solved, a membership condition for the terminal state is checked. If the condition is not satisfied, the horizon was insufficient and hence has to be increased.

Our approach is inspired from the framework of *operator splitting methods*, a class of algorithms that has recently gained considerable attention in, e.g., the compressed sensing, machine learning and image processing communities (see, *e.g.*, Combettes and Pesquet (2011); Boyd et al. (2011); Esser (2010)). From this family of algorithms, we use the Alternating Minimization Algorithm (AMA) Tseng (1991) to split the infinite-horizon constrained LQR problem into two parts, an *unconstrained* LQR problem and a proximal minimization problem. These two problems

are solved repeatedly (with the solution of one influencing the cost function of the other) until convergence to the solution to the original problem. This is in contrast to the approach of Scokaert and Rawlings (1998), which requires the solution of a sequence of *constrained* QPs. We show that both sub-problems of the proposed algorithm can be solved tractably (which is not a priori obvious since we are working with infinite sequences), the first one by solving a single finite-dimensional system of linear equations and the second one by simple clipping of finitely many real numbers on the non-positive real line. The proposed method is inspired by the splitting scheme used in O'Donoghue et al. (2012) for the finite-time LQR problem.

Convergence of the scheme, and, consequently, recovery of a stabilizing controller is guaranteed under mild assumptions. Therefore the proposed algorithmic scheme provides a means to compute the solution of the infinite-horizon constrained LQR problem with guaranteed convergence. The algorithm can address large-scale problems and, we believe, is potentially competitive for real-time control.

The paper is organized as follows: In Section 2 we introduce the problem and formulate it by means of the operator splitting framework. In Section 3 we explain in detail the algorithmic scheme for the solution. Section 4 discusses the computational aspects; we propose a method to efficiently solve the linear system that appears in each iteration of the algorithm, which is the most computationally demanding step. In Section 5 the main theoretical results are stated. In Section 6 we briefly introduce the idea of accelerating the algorithm using Nesterov's relaxation scheme. Both the basic and the accelerated version of the algorithm are illustrated with two examples in Section 7. The theoretical justification of most of the results presented here derives from operator splitting theory in Hilbert spaces. All proofs that are not presented due to space limitations are included in the extended version of the paper Stathopoulos et al. (2014).

2. PROBLEM STATEMENT AND AN OPERATOR SPLITTING APPROACH

2.1 Formulation of the problem

The goal of the paper is to solve the infinite-time constrained LQR problem

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\sum_{i=0}^{\infty} x_i^\top Q x_i + u_i^\top R u_i \\ \text{subject to} & x_{i+1} = A x_i + B u_i, \quad i \in \mathbb{N} \\ & x_0 = x_{\text{init}} \\ & C x_i + D u_i \le b. \end{array} \quad (1)$$

where $x_i \in \mathbb{R}^n$ and $u_i \in \mathbb{R}^m$ and $b \in \mathbb{R}^p$. We make the following standing assumption:

Assumption 1. The pair (A, B) is stabilizable, the optimal value of problem (1) is finite, the set

$$\mathcal{X} := \{ x \in \mathbb{R}^n \mid C x \le b \}$$

contains the origin in the interior, the matrix $[C \ D]$ has full column rank and the matrices Q and R are positive definite.

Remark 1. Assumption 1 is standard except for the requirement that Q be positive definite; this requirement facilitates the use of the Alternating minimization algorithm (AMA) to solve problem (1) and can be dropped by considering the dense form of (1); this is the subject of future work.
The full column rank assumption on the matrix $[C \ D]$ can be trivially satisfied by adding redundant constraints on states and inputs, *e.g.*, box constraints with sufficiently large diameter so that they are never activated. Note that the condition is a technicality in order for the convergence proof to hold true, and does not appear in the algorithmic implementation.

Remark 2. (Stability). Clearly, under Assumption 1, the optimal control sequence for problem (1) is stabilizing. Therefore, there is no need to enforce stability ad hoc as is commonly done when the infinite-time problem (1) is approximated by a finite-time one solved in a receding horizon fashion.

We view any infinite sequence

$$\boldsymbol{z} := (z_0, z_1, \dots) := \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{u} \end{pmatrix} := \begin{pmatrix} x_0, x_1, \dots \\ u_0, u_1, \dots \end{pmatrix}$$

as an element of an l^2-weighted (or l_w^2) real Hilbert space $\mathcal{H}_{\boldsymbol{z}}$ induced by the inner product

$$\langle \boldsymbol{z}, \boldsymbol{y} \rangle = \sum_{i=0}^{\infty} w^{-i} z_i^\top y_i, \quad \forall \, \boldsymbol{y} \in \mathcal{H}_{\boldsymbol{z}}, \, \boldsymbol{z} \in \mathcal{H}_{\boldsymbol{z}},$$

where $w > 1$. The norm of any $\boldsymbol{z} \in \mathcal{H}_{\boldsymbol{z}}$ is thus given by

$$\|\boldsymbol{z}\|_{\mathcal{H}_{\boldsymbol{z}}} := \sqrt{\langle \boldsymbol{z}, \boldsymbol{z} \rangle} = \sqrt{\sum_{i=0}^{\infty} w^{-i} \|z_i\|_2^2}.$$

Unless stated otherwise, for the rest of the paper by a Hilbert space we mean the l_w^2 real Hilbert space as just introduced.

In order to solve the problem (1) by making use of operator splitting techniques, we can rewrite (1) using the slack variables $\sigma_i \in \mathbb{R}^p$, $i \in \mathbb{N}$, as

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\sum_{i=0}^{\infty} x_i^\top Q x_i + u_i^\top R u_i \\ \text{subject to} & x_{i+1} = A x_i + B u_i, \quad i = 0, \dots \\ & x_0 = x_{\text{init}} \\ & C x_i + D u_i - \sigma_i = b, \quad \sigma_i \le 0. \end{array} \quad (2)$$

Viewing the sequence $\boldsymbol{\sigma} := (\sigma_i)_{i \in \mathbb{N}}$ as an element of $\mathcal{H}_{\boldsymbol{\sigma}}$, an l_w^2 Hilbert space defined analogously to $\mathcal{H}_{\boldsymbol{z}}$, we can further rewrite problem (2) as

$$\begin{array}{ll} \text{minimize} & h(\boldsymbol{z}) + g(\boldsymbol{\sigma}) \\ \text{subject to} & \mathcal{A}\boldsymbol{z} - \boldsymbol{\sigma} = \boldsymbol{b}, \end{array} \quad (3)$$

where

- $h(\boldsymbol{z}) = f(\boldsymbol{z}) + \delta_{\mathcal{D}}(\boldsymbol{z}) = \frac{1}{2}\langle \boldsymbol{z}, Q^\infty \boldsymbol{z} \rangle + \delta_{\mathcal{D}}(\boldsymbol{z})$, with

$$\delta_{\mathcal{D}}(\boldsymbol{z}) = \begin{cases} 0 & x_{i+1} - A x_i - B u_i = 0, \ i \in \mathbb{N} \\ & x_0 = x_{\text{init}} \\ \infty & \text{otherwise} , \end{cases}$$

  and $Q^\infty = \mathbf{diag}(\mathcal{Q}, \mathcal{Q}, \dots)$, where $\mathcal{Q} = \mathbf{diag}(Q, R)$.

-

$$g(\boldsymbol{\sigma}) = \begin{cases} 0 & \sigma_i \le 0 \ \forall i \in \mathbb{N} \\ \infty & \text{otherwise} , \end{cases}$$

- The operator $\mathcal{A} : \mathcal{H}_{\boldsymbol{z}} \to \mathcal{H}_{\boldsymbol{\sigma}}$ is defined by $(\mathcal{A}\boldsymbol{z})_i = \bar{\mathcal{A}} z_i$, where $\bar{\mathcal{A}} := [C \ D]$
- $\boldsymbol{b} = (b, b, b, \dots) \in \mathcal{H}_{\boldsymbol{\sigma}}$.

We solve problem (3) by applying the *Alternating Minimization Algorithm (AMA)* Tseng (1991) in an infinite-dimensional Hilbert space framework. AMA belongs to the family of operator splitting methods, thus allowing for decomposition of a complex optimization problem into a sequence of simpler ones. The method is presented below.

Algorithm 1 AMA for Problem (3).

0: Initialize $\boldsymbol{\lambda}^0 \in \mathcal{H}_{\boldsymbol{\sigma}}, \rho \in (0, 2\beta)$ [1]
repeat
  1: $\boldsymbol{z}^{k+1} = \underset{\boldsymbol{z} \in \mathcal{H}_{\boldsymbol{z}}}{\operatorname{argmin}} \left\{ h(\boldsymbol{z}) - \left\langle \mathcal{A}^\star \boldsymbol{\lambda}^k, \boldsymbol{z} \right\rangle \right\}$
  2: $\boldsymbol{\sigma}^{k+1} = \underset{\boldsymbol{\sigma} \in \mathcal{H}_{\boldsymbol{\sigma}}}{\operatorname{argmin}} \left\{ g(\boldsymbol{\sigma}) + \left\langle \boldsymbol{\lambda}^k, \boldsymbol{\sigma} \right\rangle + \frac{\rho}{2}\|\mathcal{A}\boldsymbol{z}^{k+1} - \boldsymbol{b} - \boldsymbol{\sigma}\|_{\mathcal{H}_{\boldsymbol{\sigma}}}^2 \right\}$
  3: $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \rho(\boldsymbol{b} - \mathcal{A}\boldsymbol{z}^{k+1} + \boldsymbol{\sigma}^{k+1})$
until termination condition is satisfied

The algorithm produces a sequence (of sequences) $\boldsymbol{z}^k$ converging to $\boldsymbol{z}^\infty$, the sequence optimal in (1). This result is stated rigorously in Section 5, Theorem 1.

Contrary to the most popular operator splitting method, the *Alternating Direction Method of Multipliers (ADMM)*, AMA considers the minimization of the standard Lagrangian at Step 1 of Algorithm 1 and the *augmented* Lagrangian at Step 2, while ADMM considers minimizing the augmented Lagrangian in both steps. It will become apparent later that this attribute is crucial in the case of the problem we are trying to solve, but comes with the extra restriction that the function $h(\boldsymbol{z})$ has to be strongly convex in order to guarantee convergence. Furthermore, it introduces restrictions to the range of feasible stepsizes ρ for AMA to provably converge.

In order to prove convergence of the method in a real Hilbert space, we view the AMA as a special case of the *forward-backward splitting* algorithm, first introduced by Bruck (1977), popularized by Passty (1977) and proven to convergence in a real Hilbert space in Bauschke and Combettes (2011). More details on convergence of AMA as used in this paper are in Section 5.

3. A FINITE DIMENSIONAL REPRESENTATION

The goal of this section is to show that each step of Algorithm 1 can be carried out in a computationally

---

[1] The permitted range for ρ depends on the spectral radius of $\bar{\mathcal{A}}$, *i.e.*, $\beta = 1/\|\bar{\mathcal{A}}\|_2^2$. The derivation of this result is presented in (Stathopoulos et al., 2014, Appendix A, Theorem 4).

tractable way (which is not a priori obvious since it involves infinite sequences of real numbers).

Written explicitly the iterations of Algorithm 1 become

$$\boldsymbol{z}^{k+1} = \underset{\boldsymbol{z}}{\arg\min} \left\{ \delta_{\mathcal{D}}(\boldsymbol{z}) + \frac{1}{2} \langle \boldsymbol{z}, Q^{\infty}\boldsymbol{z} \rangle - \langle \mathcal{A}^{\star}\boldsymbol{\lambda}^k, \boldsymbol{z} \rangle \right\} \quad (4)$$

$$\sigma_i^{k+1} = \left(\bar{\mathcal{A}}z_i^{k+1} - b - \lambda_i^k/\rho \right)_-, \ i \in \mathbb{N} \quad (5)$$

$$\lambda_i^{k+1} = \lambda_i^k + \rho(b - \bar{\mathcal{A}}z_i^{k+1} + \sigma_i^{k+1}), \ i \in \mathbb{N} \ , \quad (6)$$

where $(\cdot)_- = \min\{\cdot, 0\}$. The first step of the algorithm (Eq. (4)) is an unconstrained LQ problem with a biasing term $\langle \mathcal{A}^{\star}\boldsymbol{\lambda}^k, \boldsymbol{z} \rangle$. Therefore, if for each iterate $k \in \mathbb{N}$ the sequence $\boldsymbol{\lambda}^k = (\lambda_i^k)_{i \in \mathbb{N}}$ is zero from some time point T^k on, the first step is equivalent to the finite-dimensional equality-constrained quadratic program (QP):

$$\begin{aligned} \text{minimize} \quad & \tfrac{1}{2}x_{T^k}^{\top}P_{\mathrm{LQ}}x_{T^k} + \tfrac{1}{2}\sum_{i=0}^{T^k-1}\big\{ x_i^{\top}Qx_i + u_i^{\top}Ru_i \\ & \qquad\qquad\qquad\qquad -\lambda_i^k(Cx_i + Du_i)\big\} \\ \text{subject to} \quad & x_{i+1} = Ax_i + Bu_i, \quad i = 0,\dots,T^k \\ & x_0 = x_{\mathrm{init}} \ , \end{aligned}$$
$$(7)$$

where we minimize over $(x_0, \dots, x_{T_k})$, $(u_0, \dots, u_{T_k-1})$, and P_{LQ} is the solution to the Riccati equation corresponding to the standard linear quadratic regulator problem associated with the matrices (A, B, Q, R). Problem (7) can be efficiently solved by formulating the corresponding KKT system. The solution involves a single matrix inversion (which can be precomputed off-line for a given T^k; see Section 4 for details on how to efficiently carry out this step). For $i \geq T^k$, the control law is $u_i = K_{\mathrm{LQ}}x_i$, where the LQ gain K_{LQ} is given by $K_{\mathrm{LQ}} = (R + B^{\top}P_{\mathrm{LQ}}B)^{-1}B^{\top}P_{\mathrm{LQ}}A$. In conclusion, the first step (Eq. (4)) can be carried out efficiently as long as we can guarantee that for each k a finite time T^k exists such that $\lambda_i^k = 0$ for $i \geq T^k$.

To see that this is indeed true we need to analyze the second and third steps (Eq. (5), (6)). First, notice that when initialized with $\lambda_i^0 = 0$ for all $i \in \mathbb{N}$, the statement trivially holds for $k = 0$. Assume now $k \in \mathbb{N}$ and $\lambda_i^k = 0$ for all $i \geq T^k$. Then according to the previous discussion, for times $i \geq T^k$, the sequence x_i^{k+1} is generated by the LQ controller $u_i^{k+1} = K_{\mathrm{LQ}}x_i^{k+1}$ and therefore x_i^{k+1} converges to the origin. Consequently, by Assumption 1, there exists a time $T^{k+1} \geq T^k$ such that $\bar{\mathcal{A}}z_i^{k+1} = Cx_i^{k+1} + Du_i^{k+1} \leq b$ for all $i \geq T^{k+1}$. Looking at (5) and noticing that $\lambda_i^k = 0$ for $i \geq T^k$, it follows that $\sigma_i^{k+1} = \bar{\mathcal{A}}z_i^{k+1} - b$ for all $i \geq T^{k+1}$. As a result, the dual update term $\rho(b - \bar{\mathcal{A}}z_i^{k+1} + \sigma_i^{k+1})$ in (6) is equal to zero for all $i \geq T^{k+1}$ and therefore also $\lambda_i^{k+1} = 0$ for all $i \geq T^{k+1}$. Therefore, there indeed exists a sequence $(T^k)_{k \in \mathbb{N}}$ defined by the recursion

$$T^{k+1} := \min\{T \geq T^k \mid Cx_i^{k+1} + Du_i^{k+1} \leq b \ \forall i \geq T\}, \quad (8)$$

with $T^0 = 0$, such that $\lambda_i^k = 0$ for all $i \geq T^k$. To determine T^{k+1} computationally (given T^k and $\boldsymbol{x}^{k+1}$ and $\boldsymbol{u}^{k+1}$) we simply find the first time $T^{\mathcal{S}}$ that x_i^{k+1} enters a given subset $\mathcal{S}$, with $0 \in \mathbf{int}\,\mathcal{S}$, of the maximum positively invariant set of the system $x^+ = (A + BK_{\mathrm{LQ}})x$ subject to the constraint $(C + DK_{\mathrm{LQ}})x \leq b$. The time T^{k+1} is then equal to the first time greater than T^k such that $Cx_i^{k+1} + Du_i^{k+1} \leq b$ holds for all $i \in \{T^{k+1}, \dots, T^{\mathcal{S}}\}$. More formally, we have the equality

$$T^{k+1} = \min\big\{T \geq T^k \mid \exists T^{\mathcal{S}} \text{ s.t. } Cx_i^{k+1} + Du_i^{k+1} \leq b \ (9)$$
$$\forall i \in \{T, \dots, T^{\mathcal{S}}\} \text{ and } x_{T^{\mathcal{S}}}^{k+1} \in \mathcal{S}\big\}.$$

Remark 3. In practice, to determine T^{k+1} after solving (7), we iterate forward the system dynamics $x^+ = (A + BK_{\mathrm{LQ}})x$ starting from the initial condition $x_{T^k}^{k+1}$ until $x_i^{k+1} \in \mathcal{S}$.

Remark 4. The set $\mathcal{S}$ is determined offline and is *not* required to be invariant. A good candidate is the set $\{x \mid x^{\top}P_{\mathrm{LQ}}x \leq 1\}$ scaled such that it is included in $\{x \mid (C + DK_{\mathrm{LQ}})x \leq b\}$, or any subset of this set containing the origin in the interior.

The preceding discussion is summarized in the following algorithm:

Algorithm 2 AMA for the constrained LQR

Require: $Q \succ 0, R \succ 0, \bar{\mathcal{A}} = [C\ D]$ full column rank
  0a: Determine $P_{\mathrm{LQ}}, K_{\mathrm{LQ}}$ solving the unconstrained LQR
       problem associated with the matrices (A, B, Q, R).
  0b: Determine a set $\mathcal{S}$, with $0 \in \mathbf{int}\,\mathcal{S}$, included in
       any positively invariant set for the system
       $x^+ = (A + BK_{\mathrm{LQ}})x$ subject to the constraint
       $(C + DK_{\mathrm{LQ}})x \leq b$. See Remark 4.
  0c: Initialize $\lambda_i^0 = 0$, $T^0 = 0$.
  repeat
    1: Solve problem (7) to get $\boldsymbol{x}^{k+1}$, $\boldsymbol{u}^{k+1}$
    2: Determine T^{k+1} using (9) (see Remark 3)
    3: Set $\sigma_i^{k+1} = \left(Cx_i^{k+1} + Du_i^{k+1} - b - \lambda_i^k/\rho\right)_-$
       $i = 0, \dots, T^{k+1}$
    4: Set $\lambda_i^{k+1} = \lambda_i^k + \rho(b - Cx_i^{k+1} - Du_i^{k+1} + \sigma_i^{k+1})$
       $i = 0, \dots, T^{k+1}$
  until a termination condition is satisfied [2]
---

4. COMPUTATIONAL ASPECTS

The most expensive step of Algorithm 2 is step 1, which requires the solution of the equality-constrained QP (7). Necessary and sufficient optimality conditions for this problem are given by the KKT system

$$\begin{bmatrix} A_{11} & A_{21}^{\top} \\ A_{21} & 0 \end{bmatrix} \begin{bmatrix} \tilde{z} \\ \nu \end{bmatrix} = \begin{bmatrix} -h_1 \\ h_2 \end{bmatrix} \ . \quad (10)$$

The involved matrices and vectors are defined as follows:

$$\tilde{z} = \begin{bmatrix} x_0 \\ u_0 \\ \vdots \\ x_T \end{bmatrix}, \qquad h_1 = \begin{bmatrix} \rho C^{\top}\lambda_0^k \\ \rho D^{\top}\lambda_0^k \\ \vdots \\ \rho C^{\top}\lambda_T^k \end{bmatrix}, \qquad h_2 = \begin{bmatrix} x_{\mathrm{init}} \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$
$$A_{11} = \mathbf{diag}(I_{T^k} \otimes \mathcal{Q}, P_{\mathrm{LQ}}),$$

and

$$A_{21} = \begin{bmatrix} I & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -A & -B & I & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & -A & -B & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & I & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & -A & -B & I \end{bmatrix},$$

where A_{11} is block diagonal with T^k blocks of size $(n + m) \times (n + m)$ and the last block $n \times n$ for P_{LQ}; $A_{11} \succ 0$

---
[2] Several termination criteria exist. We simply measure the progress of the error in the states for two subsequent iterates, as described in Section 7.

since $\mathcal{Q} \succ 0$ by assumption. The matrix A_{21} is of full row rank $(T^k + 1)n$.

We use *block elimination* to solve equation (10) (see (Boyd and Vandenberghe, 2004, Appendix C)). The procedure involves inverting $S = -A_{12}^\top A_{11}^{-1} A_{12}$, which can be done by using *Cholesky factorization* on $-S$ and forward-backward substitution.

Note that the size of the QP (7) can only grow in the subsequent iterations of AMA since, by definition (8), the sequence T^k is nondecreasing. We thus look for an efficient way to solve the upcoming QPs without seriously increasing the computational load. This can be done by observing that:

(1) Regarding matrix A_{11}, increase of T^k by $\Delta T^k = T^k - T^{k-1}$ translates to inserting ΔT^k blocks $\mathcal{Q}^{-1}$ to A_{11}^{-1} such that P_{LQ}^{-1} remains the last bock. Thus A_{11} does not have to be re-inverted.

(2) Regarding matrix A_{21}, the rows are expanded by ΔT^k additional $[\, 0 \, \ldots \, 0 \, -A \, -B \,]$ matrices, and the columns with the corresponding $(0, 0, \ldots, I)$ matrices of suitable dimension.

Hence, the matrices do not need to be reformulated. A Cholesky factorization can be performed every time the matrices augment, *i.e.*, at every iterate that $\Delta T^k > 0$. Empirically, we observe that T^k changes just a few times during the first iterates and converges to a stationary value, typically long before the algorithm itself has terminated.

Remark 5. The method for solving (7) presented here is just one among many and not necessarily the most efficient one. For instance, Riccati recursion (with the bulk of it carried out offline for a sufficiently large estimate of T^k) could be significantly more efficient. This is subject to further investigation.

5. CONVERGENCE RESULTS

In this section we analyze convergence of Algorithm 2. In particular we show that (i) the state-input sequence $\boldsymbol{z}^k$ converges to the optimal state-input sequence, and (ii) that the sequence T^k defined in (8) is bounded. In order to do so, we use *monotone operator theory*. We defer proofs relying on this theory to (Stathopoulos et al., 2014, Appendix A), where we introduce the necessary background; Appendix B of the same document provides some supplementary proofs to further clarify the results from Appendix A.

For an introduction to monotone operator theory and the corresponding algorithms, the interested reader is referred to Bauschke and Combettes (2011) and Eckstein and Bertsekas (1992). The course notes Boyd and Parikh (2011) provide a more readable but brief introduction to the subject. Finally, in Esser (2010), the connection between various operator splitting methods is analyzed in a clear and comprehensible manner.

5.1 Convergence of Algorithm 2

Several results exist for convergence of operator splitting methods in infinite-dimensional Hilbert spaces. In Attouch and Soueycatt (2008) the authors prove convergence of

a variant of ADMM, namely the *Proximal Alternating Direction Method of Multipliers* (PADMM) in the weak sense. Weak convergence of the Douglas-Rachford method and of ADMM were recently proven in Svaiter (2011) and Moradifam and Nachman (2011), respectively.

It is well-known that AMA can be cast as the forward-backward splitting algorithm (FBS) (see, *e.g.*, Tseng (1991), Goldstein et al. (2012)). The result is stated in Proposition 1, Appendix A and the conversion is performed in Appendix B of Stathopoulos et al. (2014). Making use of the convergence properties of FBS in real Hilbert spaces, we can establish the following crucial result:

Theorem 1. The state-input sequence $\left(\boldsymbol{z}^k\right)_{k \in \mathbb{N}}$ generated from Algorithm 2 converges strongly to the optimal state-input sequence $\boldsymbol{z}^\infty$, i.e.,

$$\|\boldsymbol{z}^k - \boldsymbol{z}^\infty\|_{\mathcal{H}_{\boldsymbol{z}}} \xrightarrow{k \to \infty} 0.$$

Proof. The proof is provided in (Stathopoulos et al., 2014, Appendix A).

5.2 Boundedness of the sequence T^k

In this section we prove that sequence defined in (8), which guarantees that the size of the equality-constrained QP (7) solved in each iteration of Algorithm 2 is bounded. We establish this by proving that the sequence of the first hitting times of the interior of the set $\mathcal{S}$ is bounded.

Theorem 2. The sequence T^k generated by the Algorithm 2 is bounded.

Proof. First note that for the statement to hold it is sufficient to show that

$$\limsup_{k \to \infty} T^k < \infty. \tag{11}$$

To prove (11), define the sequence of the first hitting times of the interior of $\mathcal{S}$ as

$$\tau^k := \inf\{i \in \mathbb{N} \mid x_i^k \in \mathbf{int}\, \mathcal{S}\}, \quad k \in \mathbb{N} \cup \{+\infty\},$$

where $\tau^\infty < \infty$ is the hitting time of the optimal state sequence $\boldsymbol{x}^\infty$. Clearly, $\tau^k \geq T^k$ and $\tau^k < \infty$ since the origin is in the interior of $\mathcal{S}$ and for each $k \in \mathbb{N}$ the sequence $(x_i^k)_{i \in \mathbb{N}}$ generated by the Algorithm 2 converges to the origin as $i \to \infty$. We shall prove that $\limsup_{k \to \infty} \tau^k \leq \tau^\infty < \infty$, which implies (11).

For the purpose of contradiction assume that there exists a subsequence τ^{k_j}, $j \in \mathbb{N}$, with $\lim_{j \to \infty} \tau^{k_j} \geq \tau^\infty + 1$. Since the sequence of hitting times τ^k is integer valued, this implies that there exists a $j^\star \in \mathbb{N}$ such that $\tau^{k_j} \geq \tau^\infty + 1$ for all $j \geq j^\star$. We now use this to contradict the strong convergence of $\boldsymbol{x}^k$ to $\boldsymbol{x}^\infty$ from Theorem 1. To this end, observe that $x_{\tau^\infty}^\infty \in \mathbf{int}\, \mathcal{S}$ whereas $x_{\tau^\infty}^{k_j} \notin \mathbf{int}\, \mathcal{S}$ for all $j \geq j^\star$. By the definition of the interior there exists an $\epsilon > 0$ such that $y \in \mathbf{int}\, \mathcal{S}$ for all y with $\|y - x_{\tau^\infty}^\infty\|_2 < \epsilon$. Therefore $\|x_{\tau^\infty}^{k_j} - x_{\tau^\infty}^\infty\|_2 \geq \epsilon$ for all $j \geq j^\star$, and consequently

$$\|\boldsymbol{z}^{k_j} - \boldsymbol{z}^\infty\|_{\mathcal{H}_{\boldsymbol{z}}} = \sqrt{\sum_{i=0}^\infty w^{-i}(\|x_i^{k_j} - x_i^\infty\|_2^2 + \|u_i^{k_j} - u_i^\infty\|_2^2)}$$

$$\geq \sqrt{w^{-\tau^\infty}\|x_{\tau^\infty}^{k_j} - x_{\tau^\infty}^\infty\|_2^2} \geq w^{-\tau^\infty/2}\epsilon > 0$$

for all $j \geq j^\star$, contradicting the strong convergence of $\boldsymbol{z}^k$ to $\boldsymbol{z}^\infty$ asserted by Theorem 1.

6. ACCELERATION

In this section we discuss how we can accelerate the convergence of Algorithm 2 by using the fast version of AMA, called FAMA, accelerated through Nesterov's optimal over-relaxation sequence. For the particular case of AMA, the acceleration first appeared in Goldstein et al. (2012). The scheme is very simple:

Algorithm 3 FAMA for Problem (3).

0: Initialize $\boldsymbol{\lambda}^0 = \hat{\boldsymbol{\lambda}}^0 \in \mathcal{H}_{\boldsymbol{\sigma}}$ and $\alpha^0 = 1$.
repeat

2: $\boldsymbol{z}^{k+1} = \underset{\boldsymbol{z} \in \mathcal{H}_{\boldsymbol{z}}}{\operatorname{argmin}} \left\{ h(\boldsymbol{z}) - \left\langle \mathcal{A}^\top \hat{\boldsymbol{\lambda}}^k, \boldsymbol{z} \right\rangle \right\}$

3: $\boldsymbol{\sigma}^{k+1} = \underset{\boldsymbol{\sigma} \in \mathcal{H}_{\boldsymbol{\sigma}}}{\operatorname{argmin}} \left\{ g(\boldsymbol{\sigma}) + \left\langle \hat{\boldsymbol{\lambda}}^k, \boldsymbol{\sigma} \right\rangle + \frac{\rho}{2} \| \mathcal{A} \boldsymbol{z}^{k+1} - \boldsymbol{b} - \boldsymbol{\sigma} \|_{\mathcal{H}_{\boldsymbol{\sigma}}}^2 \right\}$

4: $\boldsymbol{\lambda}^{k+1} = \hat{\boldsymbol{\lambda}}^k + \rho(\boldsymbol{b} - \mathcal{A} \boldsymbol{z}^{k+1} + \boldsymbol{\sigma}^{k+1})$

5: $\alpha^{k+1} = (1 + \sqrt{1 + 4(\alpha^k)^2})/2$

6: $\hat{\boldsymbol{\lambda}}^{k+1} = \boldsymbol{\lambda}^k + \frac{\alpha^k - 1}{\alpha^{k+1}} (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1})$

until termination condition is satisfied

As demonstrated in the numerical examples in Section 7, the scheme can, depending on the particular problem instance, lead to a significant performance improvement (i.e., reduce the number of iteration needed for the algorithm to converge). On the other hand, currently there is no proof of convergence of $\boldsymbol{z}^k$ to $\boldsymbol{z}^\infty$ in general Hilbert spaces, although the authors expect that such a result should hold and are currently investigating it.

7. EXAMPLES

For illustrative purposes, we run the algorithm on two systems, a small system with two states and one input and a linearized model of a quadrocopter with 12 states and 4 inputs. We are interesting in the generated times T^k as k tends to infinity (denoted as T^∞), as well as the number of iterations that the algorithm needs for convergence. In order to do this, we sample a set of feasible initial conditions and solve the corresponding problems. The stepsize is set to the median of the allowed interval, i.e., at $\beta = 1/\|\bar{\mathcal{A}}\|_2^2$. The termination criterion is simply set as $\|\boldsymbol{x}^k - \boldsymbol{x}^{k-1}\| \leq 10^{-4}$.

7.1 Two states, one input system

Consider the following system defined as
$$A = \begin{bmatrix} 1.988 & -0.998 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1.125 \\ 0 \end{bmatrix},$$
$$x_{i+1} = Ax_i + Bu_i,$$
with constraints
$$\|x\|_\infty \leq 3, \quad \|u\|_\infty \leq 8$$
and $Q = I$, $R = 10I$.
The system is simulated for 862 different initial conditions x_0. In Figure 1 the distribution of $T^\infty = \max_k\{T^k\}$ is depicted. We see that T^∞ never exceeds 12. In Figure 2 the distribution of the iterations needed from AMA to reach the specified accuracy is presented. Many problems converge within less than 500 iterations, while a few need around 4000. The mean was computed to be 1280 iterations. Although the iterations are cheap to compute, we can state that AMA does not perform that well in terms of the number of iterations. The distribution of the iterations needed in case we use FAMA is illustrated in
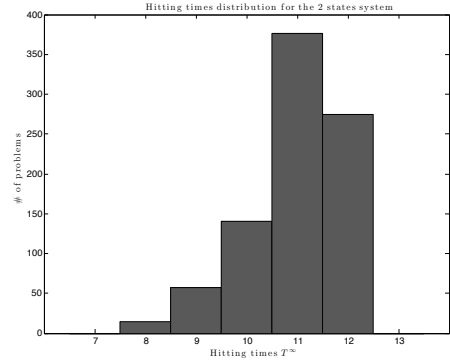

Fig. 1. Histogram of $T^\infty = \max_k\{T^k\}$ for 862 initial conditions of the 2 state system sampled from a normal distribution centered around $(1, -2)$ with standard deviation 0.5.
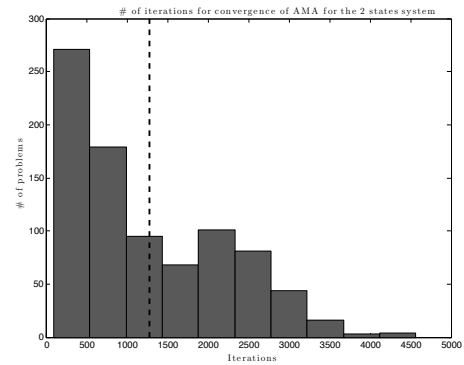

Fig. 2. Number of iterations needed for convergence for 862 instances of the 2 state system using AMA. The mean value of iterations is depicted with the black dashed line.
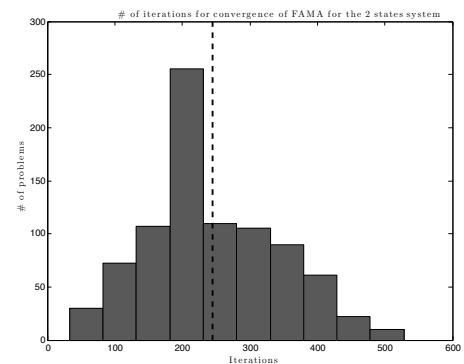

Fig. 3. Number of iterations needed for convergence for 862 instances of the 2 state system using FAMA. The mean value of iterations is depicted with the black dashed line.

Figure 3. The acceleration is significant, with problems solved up to 17 times faster than when using AMA. The average speedup is 5.2 times.

7.2 Quadcopter system

The next system we consider is a quadcopter linearized in a hovering equilibrium. The system has 12 states which correspond to position, angle and the corresponding velocities. There are 4 inputs corresponding to the 4 propellers. There are box constraints in all states and inputs, mainly ensuring the validity of the linearized model.

We simulate 64 different initial conditions sampled from a normal distribution centered around the origin with
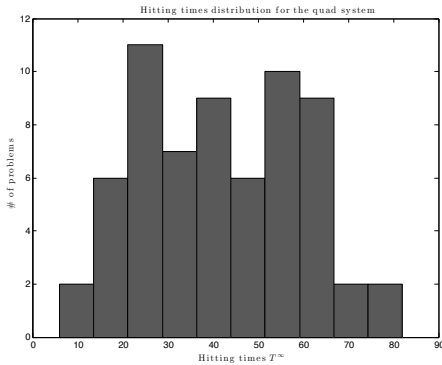
Fig. 4. Histogram of $T^\infty = \max_k\{T^k\}$ for 64 initial conditions sampled uniformly around the origin with standard deviation 0.5.
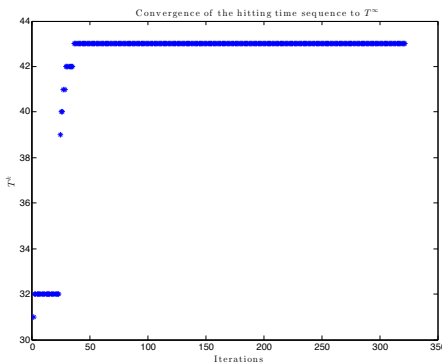

Fig. 5. Evolution of the sequence T^k defined in (8).

standard deviation 0.5, which would roughly correspond to deviations of $\approx 30°$ in terms of angles and angular velocities. A histogram of $T^\infty = \max_k\{T^k\}$ is presented in Figure 4. We can observe that the values are significantly larger than those of the previous system. Accelerating by means of FAMA was not particularly useful in this case due to the oscillatory behavior of the method near the termination threshold. There are ways to remedy this behavior, *e.g.*, the use of an adaptive restarting scheme as suggested in O'Donoghue and Candes (2013); this is a topic of further investigation. Lastly, we would like to illustrate the time evolution of the sequence T^k for a specific instance of the problem, in Figure 5. It is worth mentioning that T^k was updated in total 7 times in 322 iterations, which means that the KKT matrix was factorized only 7 times. For the rest of the iterations, we only needed to perform a forward-backward substitution.

8. CONCLUSION

We have presented a method to solve the infinite-time constrained LQR problem using the alternating minimization method (AMA) and its accelerated version. Future work will investigate another acceleration techniques (e.g., adaptive restarts, preconditioning), more efficient numerical implementation, the use of other splitting techniques, and extensions to broader problem classes (e.g., tracking).

In addition, the authors would like to thank Jean-Hubert Hours and Ye Pu for fruitful discussions.

REFERENCES

Attouch, H. and Soueycatt, M. (2008). Augmented Lagrangian and Proximal Alternating Direction Methods of Multipliers in Hilbert spaces. Applications to Games, PDE's and Control. *Pacific Journal of Optimization 5*.

Bauschke, H. and Combettes, P. (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer.

Boyd, S. and Parikh, N. (2011). Monotone operators. EE364b Course Notes.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.*

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Bruck, R. (1977). On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in Hilbert space. *J. Math. Anal. Appl.*

Combettes, P. and Pesquet, J.C. (2011). *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, chapter Proximal Splitting Methods in Signal Processing. Springer.

Eckstein, J. and Bertsekas, D. (1992). On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program.*

Esser, J. (2010). *Primal Dual Algorithms for Convex Models and Applications to Image Restoration, Registration and Nonlocal Inpainting*. URL http://books.google.ch/books?id=EVkHcgAACAAJ.

Goldstein, T., O'Donoghue, B., and Setzer, S. (2012). Fast Alternating Direction Optimization Methods. URL ftp://ftp.math.ucla.edu/pub/camreport/cam12-35.pdf.

Kalman, R. (1960). Contributions to the theory of optimal control. *Boletin de la Sociedad Matematica Mexicana.*

Moradifam, A. and Nachman, A. (2011). Convergence of the alternating split Bregman algorithm in infinite-dimensional Hilbert spaces. ArXiv:1112.1960.

O'Donoghue, B. and Candes, E. (2013). Adaptive Restart for Accelerated Gradient Schemes. *Foundations of computational mathematics.*

O'Donoghue, B., Stathopoulos, G., and Boyd, S. (2012). A splitting method for optimal control. *IEEE Transactions on Control Systems Technology.*

Passty, G. (1977). Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. *J. Math. Anal. Appl.*

Primbs, J.A. (2001). The analysis of optimization based controllers. *Automatica*, 37, 933–938.

Scokaert, P. and Rawlings, J.B. (1998). Constrained Linear Quadratic Regulation. *IEEE Transactions on Automatic Control.*

Stathopoulos, G., Korda, M., and Jones, C.N. (2014). Solving the infinite-horizon constrained LQR problem using splitting techniques. Technical report, École Polytechnique Fédérale de Lausanne. URL http://infoscience.epfl.ch/record/197368/files/CLQR_long.pdf.

Svaiter, B. (2011). On weak convergence of the Douglas-Rachford method. *SIAM J. Control and Optimization.*

Sznaier, M. and Damborg, M. (1987). Suboptimal control of linear systems with state and control inequality constraints. In *IEEE Conference on Decision and Control.*

Tseng, P. (1991). Applications of splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM J. Control Optim.*