Solving the infinite-horizon constrained LQR problem using splitting techniques

Giorgos Stathopoulos, Milan Korda and Colin N. Jones

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.

Constrained LQR, Alternating minimization, Operator splitting

I. Introduction

An important extension of the famous result of [14] 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 [19].

There have been few results addressing directly the infinite-horizon constrained LQR problem. The most well-known effort is the work of [20], where they extend the work of [22]. 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.*, [9, 6, 11]). From this family of algorithms, we use the Alternating Minimization Algorithm (AMA) [23] 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

G. Stathopoulos, M. Korda and C.N. Jones are with the Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. E-mail: {georgios.stathopoulos,milan.korda,colin.jones}@epfl.ch .

influencing the cost function of the other) until convergence to the solution to the original problem. This is in contrast to the approach of [20], 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 [17] for the finite-time LQR problem.

Convergence of the scheme, and, consequently, recovery of a stabilizing controller is guaranteed under relatively 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 II we introduce the problem and formulate it by means of the operator splitting framework. In Section III we explain in detail the algorithmic scheme for the solution. Section IV 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 V the main theoretical results are stated. In Section VI 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 VII. Finally, Appendices A, B and C provide the proofs for the results presented in Section V.

II. Problem statement and an operator splitting approach

A. Formulation of the problem

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

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\sum_{i=0}^{\infty} x_i^\top Q x_i + u_i^\top R u_i \\ \text{subject to} & x_{i+1} = Ax_i + Bu_i, \quad i \in \mathbb{N} \\ & x_0 = x_{\text{init}} \\ & Cx_i + Du_i \leq b. \end{array} \tag{1}$$

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

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

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

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

as an element of an l^2-weighted (or l^2_{w}) 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^2_{w} 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

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\sum_{i=0}^{\infty} x_i^\top Q x_i + u_i^\top R u_i \\ \text{subject to} & x_{i+1} = Ax_i + Bu_i, \quad i = 0, \ldots \\ & x_0 = x_{\text{init}} \\ & Cx_i + Du_i - \sigma_i = b, \quad \sigma_i \leq 0. \end{array} \tag{2}$$

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

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

where

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

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

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

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

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

We solve problem (3) by applying the *Alternating Minimization Algorithm (AMA)* [23] 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]

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

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 V, Theorem 1, and proven in Appendix A.

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 [8], popularized by [18] and proven to convergence in a real Hilbert space in [3]. More details on convergence of AMA as used in this paper are in Section V.

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

Written explicitly the iterations of Algorithm 1 become

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

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

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

[1]The permitted range for ρ is given in Theorem 4, Appendix A.

[15]We denote by $(\cdot)^\star$ the *adjoint* of an operator, which has the analogy of a transpose in a finite-dimensional space. More on operator theory is presented in Appendix A.

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):

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} x_{T^k}^\top P_{\mathrm{LQ}} x_{T^k} + \frac{1}{2}\sum_{i=0}^{T^k-1}\big\{x_i^\top Q x_i + u_i^\top R u_i \\ & \qquad\qquad\qquad -\lambda_i^k (C x_i + D u_i)\big\} \\ \text{subject to} & x_{i+1} = A x_i + B u_i, \quad i = 0,\ldots,T^k \\ & x_0 = x_{\mathrm{init}} \ , \end{array} \tag{7}$$

where we minimize over $(x_0,\ldots,x_{T_k})$, $(u_0,\ldots,u_{T_k-1})$, 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 *Karush-Kuhn-Tucker* (KKT) system. The solution involves a single matrix inversion (which can be precomputed off-line for a given T^k; see Section IV 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 $\bar{A} z_i^{k+1} = C x_i^{k+1} + D u_i^{k+1} \leq b$ 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 $\sigma_i^{k+1} = \bar{A} z_i^{k+1} - b$ for all $i \geq T^{k+1}$. As a result, the dual update term $\rho(b - \bar{A} z_i^{k+1} + \sigma_i^{k+1})$ 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

$$T^{k+1} := \min\{T \geq T^k \mid C x_i^{k+1} + D u_i^{k+1} \leq b \;\forall\, i \geq T\}, \tag{8}$$

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 $C x_i^{k+1} + D u_i^{k+1} \leq b$ holds for all $i \in \{T^{k+1},\ldots,T^{\mathcal{S}}\}$. More formally, we have the equality

$$\begin{aligned} T^{k+1} = \min\big\{T \geq T^k \mid \exists\, T^{\mathcal{S}} \text{ s.t. } C x_i^{k+1} + D u_i^{k+1} \leq b \\ \forall\, i \in \{T,\ldots,T^{\mathcal{S}}\} \ \text{ and } \ x_{T^{\mathcal{S}}}^{k+1} \in \mathcal{S}\big\}. \end{aligned} \tag{9}$$

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: $Q \succ 0, R \succ 0$, $\bar{A} = [C\ D]$ full column rank

0a: Determine P_{LQ}, K_{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 $\sigma_i^{k+1} = \left(Cx_i^{k+1} + Du_i^{k+1} - b - \lambda_i^k/\rho\right)_-$ $i = 0, \ldots, T^{k+1}$

4: Set $\lambda_i^{k+1} = \lambda_i^k + \rho(b - Cx_i^{k+1} - Du_i^{k+1} + \sigma_i^{k+1})$ $i = 0, \ldots, T^{k+1}$

until a termination condition is satisfied[3]

IV. 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

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

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

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

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

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}; $A_{11}\succ 0$ 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 [7, 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 $\begin{bmatrix} 0 & \dots & 0 & -A & -B \end{bmatrix}$ matrices, and the columns with the corresponding $(0,0,\dots,I)$ 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.

V. 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*. For the sake of clarity we defer proofs relying on this theory to Appendix A, where we introduce the necessary background; in Appendices B and C we provide 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 [3] and [10]. The course notes by [5] provide a more readable but brief introduction to the subject. Finally, in [11], the connection between various operator splitting methods is analyzed in a clear and comprehensible manner.

A. Convergence of Algorithm 2

Several results exist for convergence of operator splitting methods in infinite-dimensional Hilbert spaces. In [1] 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 was recently proven in [21]. The authors of [15], based on this latest result and using the duality link between the alternating split Bregman and the Douglas-Rachford method, prove weak convergence of the alternating split Bregman method.

It is well-known that AMA can be cast as the forward-backward splitting algorithm (FBS) (see, *e.g.*, [23], [12]). The result is stated in Proposition 1 of Appendix A and the conversion is performed in Appendix B. Making use of the convergence properties of FBS in real Hilbert spaces (Theorem 4, Appendix A, 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.,

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

The proof is provided in Appendix A.

B. 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.

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 $x^\infty_{\tau^\infty} \in \mathbf{int}\,\mathcal{S}$ whereas $x^{k_j}_{\tau^\infty} \notin \mathbf{int}\,\mathcal{S}$ 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 $\|y - x^\infty_{\tau^\infty}\|_2 < \epsilon$. Therefore $\|x^{k_j}_{\tau^\infty} - x^\infty_{\tau^\infty}\|_2 \geq \epsilon$ for all $j \geq j^\star$, and consequently

$$\begin{aligned}\|\boldsymbol{z}^{k_j} - \boldsymbol{z}^\infty\|_{\mathcal{H}_{\boldsymbol{z}}} &= \sqrt{\sum_{i=0}^{\infty} w^{-i}(\|x_i^{k_j} - x_i^\infty\|_2^2 + \|u_i^{k_j} - u_i^\infty\|_2^2)} \\ &\geq \sqrt{w^{-\tau^\infty}\|x^{k_j}_{\tau^\infty} - x^\infty_{\tau^\infty}\|_2^2} \geq w^{-\tau^\infty/2}\epsilon > 0\end{aligned}$$

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

VI. 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 [12]. 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: $\boldsymbol{z}^{k+1} = \underset{\boldsymbol{z}\in\mathcal{H}_{\boldsymbol{z}}}{\operatorname{argmin}} \left\{ h(\boldsymbol{z}) - \left\langle \mathcal{A}^\star \hat{\boldsymbol{\lambda}}^k, \boldsymbol{z} \right\rangle \right\}$
3: $\boldsymbol{\sigma}^{k+1} = \underset{\boldsymbol{\sigma}\in\mathcal{H}_{\boldsymbol{\sigma}}}{\operatorname{argmin}} \left\{ g(\boldsymbol{\sigma}) + \left\langle \hat{\boldsymbol{\lambda}}^k, \boldsymbol{\sigma} \right\rangle + \frac{\rho}{2} \|\mathcal{A}\boldsymbol{z}^{k+1} - \boldsymbol{b} - \boldsymbol{\sigma}\|^2_{\mathcal{H}_{\boldsymbol{\sigma}}} \right\}$
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 VII, 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.

VII. 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 β as is computed in Proposition 1 in Appendix A. The termination criterion is simply set as $\|\boldsymbol{x}^k - \boldsymbol{x}^{k-1}\| \leq 10^{-4}$.

A. 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

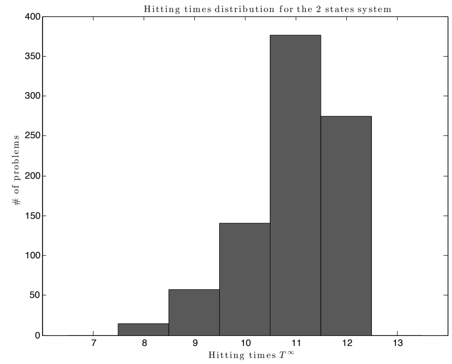

Fig. 1. Histogram of $T^\infty = \max_k\{T^k\}$ for 862 initial conditions of the 2 state system sampled from a normal distribution centered around $(1, -2)$ with standard deviation 0.5.

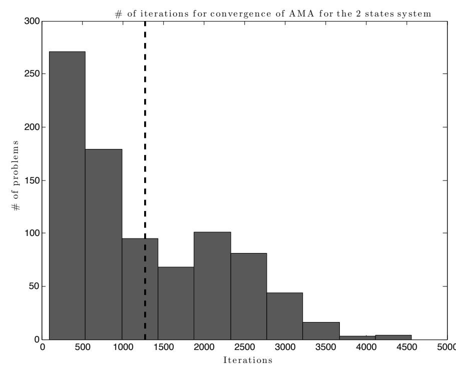

Fig. 2. Number of iterations needed for convergence for 862 instances of the 2 state system using AMA. The mean value of iterations is depicted with the black dashed line.

$T^\infty = \max_k\{T^k\}$ 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 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. An interesting observation is the existence of few problems for which FAMA is actually slower than AMA. The reason for this is that the accelerated sequence might become too aggressive, resulting to oscillatory behavior around the optimum.

B. *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

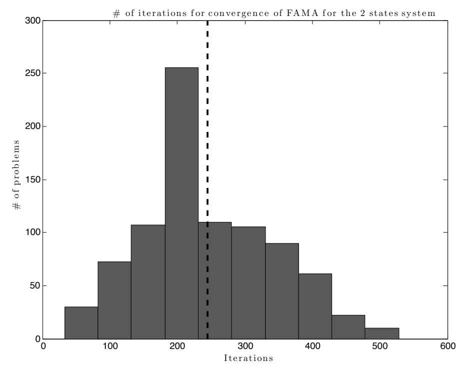

Fig. 3. Number of iterations needed for convergence for 862 instances of the 2 state system using FAMA. The mean value of iterations is depicted with the black dashed line.

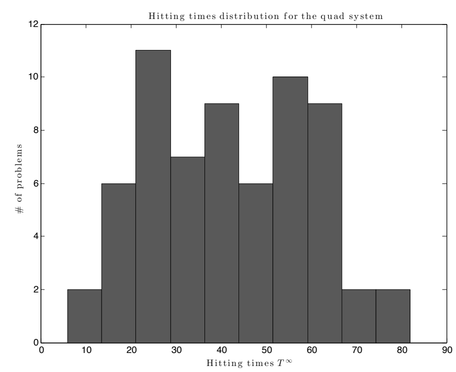

Fig. 4. Histogram of $T^\infty = \max_k\{T^k\}$ for 64 initial conditions sampled uniformly around the origin with standard deviation 0.5.

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 standard deviation 0.5, which would roughly correspond to deviations of $\approx 30^\circ$ in terms of angles and angular velocities. A histogram of $T^\infty = \max_k\{T^k\}$ is 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 [16]; 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.

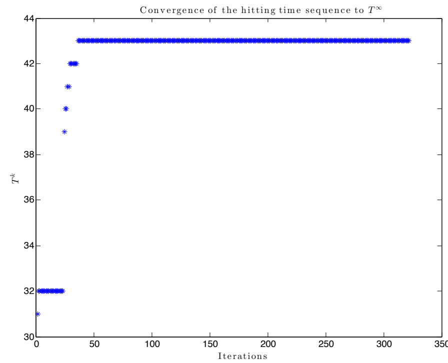

Fig. 5. Evolution of the sequence T^k defined in (8).

APPENDIX A

In this section we introduce some notation and definitions from monotone operator theory. We further present the forward-backward algorithm and its connection with AMA. The subsequent results hold for any general real Hilbert space, including the special case of l^2_{w} we consider. We write variables in normal font, and we use the bold font to describe the infinite-dimensional variables we are manipulating in our problem description.

Monotone Operators

An *operator* $A\colon \mathcal{H}_1 \to 2^{\mathcal{H}_2}$ is a point-to-set map, *i.e.*, A maps every point $x \in \mathcal{H}_1$ to a set $A(x) \subseteq \mathcal{H}_2$. The operator is characterized by its *graph*, $\mathbf{gra}\, A = \{(x,u) \in \mathcal{H}_1 \times \mathcal{H}_2 \mid u \in A(x)\}$. The *inverse* A^{-1} of A is defined through its graph as $\mathbf{gra}\, A^{-1} = \{(u,x) \in \mathcal{H}_2 \times \mathcal{H}_1 \mid (x,u) \in \mathbf{gra}\, A\}$. The set of zeros of A is defined as $\mathbf{zer}\, A = \{x \in \mathcal{H}_1 \mid 0 \in A(x)\}$. Composition, scalar multiplication and addition of operators are well-defined operations (see, *e.g.*, [5]).

Definition 1: [3, Definition 20.1] Let $A\colon \mathcal{H} \to 2^{\mathcal{H}}$. Then A is monotone if

$$\langle x-y, u-v\rangle \geq 0 \quad (\forall (x,u) \in \mathbf{gra}\, A)(\forall (y,v) \in \mathbf{gra}\, A) \ .$$

Definition 2: [3, Definition 20.20] Let $A\colon \mathcal{H} \to 2^{\mathcal{H}}$ be monotone. Then A is maximally monotone if there exists no monotone operator $B\colon \mathcal{H} \to 2^{\mathcal{H}}$ such that $\mathbf{gra}\, B$ properly contains $\mathbf{gra}\, A$, *i.e.*, for every $(x,u) \in \mathcal{H} \times \mathcal{H}$,

$$(x,u) \in \mathbf{gra}\, A \Leftrightarrow (\forall (y,v) \in \mathbf{gra}\, A) \quad \langle x-y, u-v\rangle \geq 0 \ .$$

The best-known example of a maximally monotone operator is the subgradient mapping ∂f of a closed proper convex function $f\colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$.

Definition 3: A linear operator (mapping) $T\colon \mathcal{H}_1 \to \mathcal{H}_2$ between two Hilbert spaces is said to be bounded if the operator norm $\|T\|$ of T, defined as

$$\|T\| := \sup_{\|x\|_{\mathcal{H}_1}=1} \|Tx\|_{\mathcal{H}_2},$$

satisfies $\|T\| < \infty$. Then, $\forall x \in \mathcal{H}_1$ we have $\|Tx\| \leq \|T\|\|x\|$. The set of bounded operators between two Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$ is denoted as $\mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$.

Definition 4: Let $\mathcal{H}_1, \mathcal{H}_2$ be real Hilbert spaces and $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$. The adjoint of T is the unique operator $T^\star \in \mathcal{B}(\mathcal{H}_2, \mathcal{H}_1)$ that satisfies

$$\langle Tx, y\rangle = \langle y, T^\star x\rangle \quad (\forall x \in \mathcal{H}_1)(\forall y \in \mathcal{H}_2) \ .$$

Definition 5: [3, Definition 23.1] Let $A\colon \mathcal{H} \to 2^{\mathcal{H}}$. The resolvent of A is

$$J_A = (I + A)^{-1} \ ,$$

and I stands for the identity operator defined by

$$\mathbf{gra}\, I = \{(x, x) \in \mathcal{H} \times \mathcal{H} \mid x \in A(x)\} \, .$$

Definition 6: We define the range of an bounded operator $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ as $\mathbf{ran}(T) = T(\mathcal{H}_2)$. The kernel of the operator is defined as $\mathbf{ker}(T) = \{x \in \mathcal{H}_1 \mid Tx = 0\}$.
We say that $\mathbf{ran}(T)$ is closed if and only if $\mathbf{ran}(T^\star)$ is closed, which is equivalent to the existence of an $\alpha > 0$ such that $(\forall x \in (\mathbf{ker}(T)^\perp)$, $Tx \geq \alpha\|x\|$ [3, Fact 2.19].

Definition 7: [3, Example 28.14] Let the operator $L \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ be such that $\mathbf{ran}(L)$ is closed, let $y \in \mathbf{ran}(L)$, let $\mathcal{C} = \{x \in \mathcal{H}_1 \mid Lx = y\}$, and $x \in \mathcal{H}_1$. Under the assumption that $LL^\star$ is invertible, we define the projection onto the subspace $\mathcal{C}$ as

$$P_{\mathcal{C}}(x) = x - L^\star(LL^\star)^{-1}(Lx - y) \ .$$

Subsequently, we introduce the notions of weak and strong convergence.

Definition 8: Let $\mathcal{H}$ be a Hilbert space. We say that $\left(x^k\right)_{k\in\mathbb{N}}$ covnerges weakly to x if $\forall y \in \mathcal{H}$ $\left\langle y, x^k\right\rangle \xrightarrow{k\to\infty} \langle y, x\rangle$.

Definition 9: Let $(x^k)_{k\in\mathbb{N}}$ be a sequence in $\mathcal{H}$. Then $\left(x^k\right)_{k\in\mathbb{N}}$ converges strongly to x if $\|x^k - x\| \xrightarrow{k\to\infty} 0$.
The Baillon-Haddad Theorem (see [2]) shows the relationship between the Lipschitz continuity and the cocoerciveness of the gradient of a convex differentiable function and is important for enabling the computation of the permitted stepsize interval for AMA.

Theorem 3 (Baillon-Haddad): Let $f\colon \mathcal{H} \to \mathbb{R}$ be a convex differentiable function. The following are equivalent:

(i) ∇f is Lipschitz continuous with constant β.

(ii) ∇f if $1/\beta$-cocoercive, *i.e.*, $\forall x, y \in \mathcal{H}$,

$$\langle \nabla f(y) - \nabla f(x), y - x\rangle \geq 1/\beta\|\nabla f(y) - \nabla f(x)\|^2.$$

Finally, we are going to need the expression for the dual formulation of (3). If (3) is written in the compact form

$$\min_{\boldsymbol{z}\in\mathcal{H}_{\boldsymbol{z}}} \{h(\boldsymbol{z}) + g(\mathcal{A}\boldsymbol{z} - b)\} \ , \tag{P}$$

then the corresponding dual problem is[4]

$$\min_{\boldsymbol{\lambda}\in\mathcal{H}_{\boldsymbol{\sigma}}} \{h^\star(\mathcal{A}^\star\boldsymbol{\lambda}) + g^\star(-\boldsymbol{\lambda}) - \langle\boldsymbol{\lambda}, b\rangle\} \ , \tag{D}$$

where $h^\star(\cdot)$ and $g^\star(\cdot)$ denote the convex conjugates of $h(\cdot)$ and $g(\cdot)$.

[4]Derivation of the dual is presented in Appendix B.

FBS and AMA

We are now ready to state the result on convergence of the forward-backward (FBS) algorithm [3, Therorem 25.8].

Theorem 4: Let $\Psi\colon \mathcal{H} \to 2^{\mathcal{H}}$ be maximally monotone, let $\beta \in \mathbb{R}_{++}$, let $\Phi\colon \mathcal{H} \to \mathcal{H}$ be $\beta - cocoercive$, let $\rho \in (0, 2\beta)$ and set $\delta = \min\{1, \beta/\rho\} + 1/2$. Furthermore, let $\left(\mu^k\right)_{k\in\mathbb{N}}$ be a sequence in $[0, \delta]$ such that $\sum_{k\in\mathbb{N}} \mu^k(\delta - \mu^k) = +\infty$, and let $\lambda^0 \in \mathcal{H}$. Suppose that $\mathbf{zer}(\Psi + \Phi) \neq \emptyset$ and set, for $k = 0, 1, \ldots$

$$y^k = \lambda^k - \rho\Phi(\lambda^k)$$
$$\lambda^{k+1} = \lambda^k + \mu^k(J_{\rho\Psi}(y^k) - \lambda^k)$$

Then the following hold:

(i) $\left(\lambda^k\right)_{k\in\mathbb{N}}$ converges weakly to a point in $\mathbf{zer}(\Psi + \Phi)$.

(ii) Suppose that $\inf_{k\in\mathbb{N}} \mu^k > 0$ and let $\lambda^\infty \in \mathbf{zer}(\Psi+\Phi)$. Then $\left(\Phi(\lambda^k)\right)_{k\in\mathbb{N}}$ converges strongly to $\Phi(\lambda^\infty)$.

We will also need the following proposition on the relation between AMA and FBS.

Proposition 1: AMA, as given by Algorithm 1, is a special case of FBS when considering the following:

(i) $\Psi(\boldsymbol{\lambda}) = -\partial g^\star(-\boldsymbol{\lambda}) - b, \quad \Phi(\boldsymbol{\lambda}) = \mathcal{A}\nabla h^\star(\mathcal{A}^\star\boldsymbol{\lambda})$.

(ii) $\beta = \frac{1}{\tau\|\bar{\mathcal{A}}\|^2}$, where τ is the Lipschitz constant of $\nabla h^\star$ and $\|\bar{A}\|^2 = \sigma_{\max}(\bar{A})$.

(iii) $\mu^k = 1,\ \forall k \in \mathbb{N}$.

The proof is provided in Appendix B.

Based on Theorem 4 and Proposition 1 we get the following instrumental Lemma:

Lemma 1: The sequence $\mathcal{A}\boldsymbol{z}^k$ converges strongly to $\mathcal{A}\boldsymbol{z}^\infty$.

Proof 1: The result is proven in Appendix B.

Lemma 1 allows as to prove Theorem 1, one of our main results.

Proof of Theorem 1

Denoting $\boldsymbol{e}^k := \mathcal{A}(\boldsymbol{z}^k - \boldsymbol{z}^\infty) \in \mathcal{H}_{\boldsymbol{\sigma}}$, we have:

$$\begin{aligned} \|\boldsymbol{e}^k\|_{\mathcal{H}_{\boldsymbol{\sigma}}} &= \sqrt{\sum_{i=0}^{\infty} w^{-i}\|\bar{A}(z_i^k - z_i^\infty)\|_2^2} \\ &\geq \sqrt{\sum_{i=0}^{\infty} w^{-i}\sigma_{\min}(\bar{A})^2\|z_i^k - z_i^\infty\|_2^2} \\ &= \sigma_{\min}(\bar{A})\|\boldsymbol{z}^k - \boldsymbol{z}^\infty\|_{\mathcal{H}_{\boldsymbol{z}}} \ , \end{aligned}$$

where $\sigma_{\min}(\bar{A}) > 0$ is the smallest singular value of $\bar{A}$; the inequality holds by the assumption that $\bar{A}$ is of full column rank. Thus, $\|\boldsymbol{z}^k - \boldsymbol{z}^\infty\|_{\mathcal{H}_{\boldsymbol{z}}} \leq \sigma_{\min}(\bar{A})^{-1}\|\boldsymbol{e}^k\|_{\mathcal{H}_{\boldsymbol{\sigma}}} \xrightarrow{k\to\infty} 0$ since $\boldsymbol{e}^k$ converges strongly to zero by Lemma 1. Therefore $\boldsymbol{z}_k \xrightarrow{k\to\infty} \boldsymbol{z}^\infty$ strongly as desired.

Appendix B

The goal of this Appendix is to derive (D), prove Proposition 1, and Lemma 1.

Derivation of (D)

Starting from

$$\min_{\boldsymbol{z}} \left\{ h(\boldsymbol{z}) + g(\mathcal{A}\boldsymbol{z} - \boldsymbol{b}) \right\},$$

we can express the Lagrange dual problem by using the slack variable $\boldsymbol{\sigma} = \mathcal{A}\boldsymbol{z} - \boldsymbol{b}$ and the dual variable $\boldsymbol{\lambda}$ as

$$\begin{aligned}
&\max_{\boldsymbol{\lambda}} \left\{ \min_{\boldsymbol{z},\boldsymbol{\sigma}} \left\{ h(\boldsymbol{z}) + g(\boldsymbol{\sigma}) + \langle \boldsymbol{b} - \mathcal{A}\boldsymbol{z} + \boldsymbol{\sigma}, \boldsymbol{\lambda} \rangle \right\} \right\} \Leftrightarrow \\
&\max_{\boldsymbol{\lambda}} \Big\{ -\max_{\boldsymbol{z}} \left\{ \langle \mathcal{A}^{\star}\boldsymbol{\lambda}, \boldsymbol{z} \rangle - h(\boldsymbol{z}) \right\} - \max_{\boldsymbol{\sigma}} \left\{ \langle -\boldsymbol{\lambda}, \boldsymbol{\sigma} \rangle - g(\boldsymbol{\sigma}) \right\} + \langle \boldsymbol{\lambda}, \boldsymbol{b} \rangle \Big\} \ .
\end{aligned}$$

Using *Legendre-Fenchel duality* (see, *e.g.*, Chapter 7, [4]), we can rewrite as

$$\begin{aligned}
&\max_{\boldsymbol{\lambda}} \left\{ -h^{\star}(\mathcal{A}^{\star}\boldsymbol{\lambda}) - g^{\star}(-\boldsymbol{\lambda}) + \langle \boldsymbol{\lambda}, \boldsymbol{b} \rangle \right\} \Leftrightarrow \\
&\min_{\boldsymbol{\lambda}} \left\{ h^{\star}(\mathcal{A}^{\star}\boldsymbol{\lambda}) + g^{\star}(-\boldsymbol{\lambda}) - \langle \boldsymbol{\lambda}, \boldsymbol{b} \rangle \right\} \quad ,
\end{aligned} \tag{12}$$

which is (D).

Proof of Proposition 1

For the sake of simplicity in the proof and without loss of generality, we assume that the matrices Q, R are identities. This is a valid assumption since we can perform a change of basis for optimization problem (1), by first diagonalizing Q and R, and then scale them such that they become identities. Note that this is possible under Assumption 1, *i.e.*, positive definiteness of the matrices. This change of coordinates will result in new variables x_i, u_i, as well as new matrices A, B, C, D (and consequently operators). The resulting operator $\boldsymbol{Q}^{\infty}$ becomes the identity operator, *i.e.*, $\boldsymbol{Q}^{\infty} = \mathbf{I}$. The operator $\mathcal{A}$ essentially has the same block diagonal structure as before. The procedure can be carried out offline.

Looking at (12), one can define

$$\Psi(\boldsymbol{\lambda}) = -\partial g^{\star}(-\boldsymbol{\lambda}) - b \tag{13}$$

$$\Phi(\boldsymbol{\lambda}) = \mathcal{A}\nabla h^{\star}(\mathcal{A}^{\star}\boldsymbol{\lambda}) \ . \tag{14}$$

(Note that it is not immediately obvious that $h^{\star}$ is differentiable, since the function is a restriction of a quadratic form in a subspace, hence not necessarily smooth. We will prove in Appendix C that smoothness indeed holds.) The two operators are maximally monotone (see Definition 2), since $\Psi(\boldsymbol{\lambda})$ is the subdifferential of a lower semicontinuous convex function, and $\Phi(\boldsymbol{\lambda})$ can be written as the affine composition of $h^{\star}$ with $\mathcal{A}^{\star}\boldsymbol{\lambda}$, with $h^{\star}$ being again convex and lower semicontinuous. Hence, (12) can be cast as the problem of finding the zeros of the maximally monotone operator $\Phi(\boldsymbol{\lambda}) + \Psi(\boldsymbol{\lambda})$, *i.e.*,

$$\text{find } \boldsymbol{\lambda} \in \mathcal{H} \text{ such that } 0 \in \Phi(\boldsymbol{\lambda}) + \Psi(\boldsymbol{\lambda}) \ .$$

Let us now take the optimality conditions for the first two steps of AMA (Algorithm 1).

The $\boldsymbol{z}$ minimization involves the computation of the convex conjugate of $h(\boldsymbol{z}) = (f + \delta_{\mathcal{D}})(\boldsymbol{z})$, in other words the conjugate of a quadratic function restricted to the subspace $\mathcal{D} := \{\boldsymbol{z} \mid$

$\boldsymbol{D}\boldsymbol{z} = \boldsymbol{d}\}$, where $\boldsymbol{D}$ is the operator representing the dynamics equation. A way to construct the operator $\boldsymbol{D}$ is to extend A_{21} from Section IV to infinity and, for the sequence $\boldsymbol{d}$, to extend h_2 with infinitely many zeros.

In order to restrict the function to the subspace, we introduce the *orthogonal projection operator*

$$P_{\mathcal{D}}(\boldsymbol{y}) = \Pi_{\mathcal{N}}\boldsymbol{y} + \pi_{\mathcal{D}^\star}, \quad \boldsymbol{y} \in \mathcal{H}_{\boldsymbol{z}} \tag{15}$$

where we have introduced the following:

- $\boldsymbol{D}^\dagger$ is the generalized inverse of the operator $\boldsymbol{D}$ defined as $\boldsymbol{D}^\dagger := \boldsymbol{D}^\star(\boldsymbol{D}\boldsymbol{D}^\star)^{-1}$, where $\boldsymbol{D}^\star$ is the adjoint operator of $\boldsymbol{D}$, as stated in Definition 4.
- $\pi_{\mathcal{D}^\star} := \boldsymbol{D}^\dagger\boldsymbol{d}$.
- $\Pi_{\mathcal{D}^\star} := \boldsymbol{D}^\dagger\boldsymbol{D}$ is the projection operator onto the domain of $\boldsymbol{D}$, $\mathbf{ran}(\boldsymbol{D}^\star)$.
- $\Pi_{\mathcal{N}} := I - \Pi_{\mathcal{D}^\star}$ is the projection operator onto the subspace $\mathcal{N} = \mathbf{ker}(\boldsymbol{D})$, such that $\boldsymbol{D}\Pi_{\mathcal{N}} = 0$ holds.

Note that the projection operator $P_{\mathcal{D}}(\cdot)$ is well-defined according to Definition 7, since $\boldsymbol{D}$ is bounded by $1+\sigma_{\max}([-A \ -B])$ (using the triangle inequality) and is also of closed range since it is a surjective operator (due to the identity blocks).

Using the above notation, the convex conjugate of $h(\boldsymbol{z})$ is defined as

$$h^\star(\boldsymbol{p}) = \frac{1}{2}\langle \boldsymbol{p}, \Pi_{\mathcal{N}}\boldsymbol{p}\rangle + \langle \boldsymbol{p}, \pi_{\mathcal{D}^\star}\rangle - \frac{1}{2}\langle \pi_{\mathcal{D}^\star}, \pi_{\mathcal{D}^\star}\rangle \quad . \tag{16}$$

The formula is derived in Appendix C.

Writing the optimality condition for Step 1 of Algorithm 1, we have that:

$$\begin{aligned}
&\nabla h(\boldsymbol{z}^{k+1}) - \mathcal{A}^\star\boldsymbol{\lambda}^k = 0\\
&\mathcal{A}^\star\boldsymbol{\lambda}^k = \nabla h(\boldsymbol{z}^{k+1})\\
&\boldsymbol{z}^{k+1} = \nabla h^\star(\mathcal{A}^\star\boldsymbol{\lambda}^k)\\
&\mathcal{A}\boldsymbol{z}^{k+1} = \mathcal{A}\nabla h^\star(\mathcal{A}^\star\boldsymbol{\lambda}^k)\\
&\mathcal{A}\boldsymbol{z}^{k+1} = \Phi(\boldsymbol{\lambda}^k)
\end{aligned} \tag{17}$$

From the $\boldsymbol{\sigma}$ minimization we have:

$$\begin{aligned}
&\partial g(\boldsymbol{\sigma}^{k+1}) + \boldsymbol{\lambda}^k - \rho(-\boldsymbol{\sigma}^{k+1} + \mathcal{A}\boldsymbol{z}^{k+1} - b) \ni 0\\
&\partial g(\boldsymbol{\sigma}^{k+1}) \ni (-\boldsymbol{\lambda}^k + \rho(\mathcal{A}\boldsymbol{z}^{k+1} - \boldsymbol{\sigma}^{k+1} - b)
\end{aligned}$$

From the $\boldsymbol{\lambda}^k$ update of the algorithm we have:

$$\begin{aligned}
\partial g(\boldsymbol{\sigma}^{k+1}) &\ni -\boldsymbol{\lambda}^{k+1}\\
\boldsymbol{\sigma}^{k+1} &\in \partial g^\star(-\boldsymbol{\lambda}^{k+1})\\
-\boldsymbol{\sigma}^{k+1} - b &\in -\partial g^\star(-\boldsymbol{\lambda}^{k+1}) - b\\
-\boldsymbol{\sigma}^{k+1} - b &\in \Psi(\boldsymbol{\lambda}^{k+1})
\end{aligned} \tag{18}$$

Finally, using again the $\boldsymbol{\lambda}^k$ update we have:

$$\begin{aligned}
&0 \in \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k - \rho(b - \mathcal{A}\boldsymbol{z}^{k+1} + \boldsymbol{\sigma}^{k+1})\\
&0 \in \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k + \rho\Phi(\boldsymbol{\lambda}^k) + \rho\Psi(\boldsymbol{\lambda}^{k+1})
\end{aligned}$$

Hence, we get that

$$\boldsymbol{\lambda}^{k+1} + \rho\Psi(\boldsymbol{\lambda}^{k+1}) = \boldsymbol{\lambda}^{k} - \rho\Phi(\boldsymbol{\lambda}^{k})$$
$$(I + \rho\Psi)(\boldsymbol{\lambda}^{k+1}) = (I - \rho\Phi)(\boldsymbol{\lambda}^{k})$$

From the above, we recover the FBS iteration

$$\boldsymbol{\lambda}^{k+1} = (I + \rho\Psi)^{-1}(I - \rho\Phi)(\boldsymbol{\lambda}^{k}) \ , \tag{19}$$

or, using the resolvent definition (Definition 5), (19) can be written as

$$\boldsymbol{\lambda}^{k+1} = J_{\rho\Psi}(\boldsymbol{\lambda}^{k} - \rho\Phi(\boldsymbol{\lambda}^{k})) \ ,$$

which is the scheme from Theorem 4 with $\mu^k = 1$.
This proves points (i) and (iii) of the proposition.

It is now left to compute the β parameter as it appears in point (ii).
Note that from Theorem 3 and (13), computing the Lipschitz constant of $\Phi(\boldsymbol{\lambda})$ is sufficient in order to recover the β-cocoercivity parameter.
It is known that if $\nabla h^\star$ is τ-Lipschitz, then $\mathcal{A}\nabla h^\star(\mathcal{A}^\star\boldsymbol{\lambda})$ is $\tau\|\mathcal{A}^\star\|^2$. Following the same steps as in Theorem 1, we can prove that $\mathcal{A}^\star$ is a bounded operator, *i.e.*, $\|\mathcal{A}^\star\| \leq \sigma_{\max}(\bar{\mathcal{A}})$. On the other hand, we have from (16) and the fact that the orthogonal projection operator $\Pi_{\mathcal{N}}$ is bounded by 1 that

$$\langle \nabla h^\star(\boldsymbol{p}_1) - \nabla h^\star(\boldsymbol{p}_2), \boldsymbol{p}_1 - \boldsymbol{p}_2 \rangle \leq \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|_{\mathcal{H}_{\boldsymbol{z}}} \ .$$

Consequently, $\Phi(\boldsymbol{\lambda}) = \mathcal{A}\nabla f^\star(\mathcal{A}^\star\boldsymbol{\lambda})$ is β-cocoercive with $\beta = 1/\|\bar{\mathcal{A}}\|_2^2$.

Proof of Lemma 1

The proof follows directly from point (ii) of Theorem 4 along with the inclusion (17).

APPENDIX C

The goal of this Appendix is to derive the conjugate of $h(\boldsymbol{z})$, as given in (16).

From [13, Proposition 1.3.2], we have that the Legendre-Fenchel conjugate of a convex function f restricted to a subspace $\mathcal{H}_D \subset \mathcal{H}_z$ is given by

$$\begin{aligned}(f + \delta_{\mathcal{H}_D})^\star(p) &= (f \circ P_{\mathcal{H}_D})^\star \circ P_{\mathcal{H}_D}(y) \\ &= \sup\{\langle p, z\rangle - f(z) \ \mid \ z \in \mathcal{H}_D\} \\ &= \sup\{\langle P_{\mathcal{H}_D}(p), y\rangle - f(P_{\mathcal{H}_D}(y)) \ \mid \ y \in \mathcal{H}_z\} \ ,\end{aligned}$$

where $P_{\mathcal{H}_D}$ is the operator of orthogonal projection onto $\mathcal{H}_D$.

In our case, we can write that

$$h^\star(\boldsymbol{p}) = \sup\{\langle \boldsymbol{p}, P_{\mathcal{D}}(\boldsymbol{y})\rangle - f(P_{\mathcal{D}}(\boldsymbol{y})) \ \mid \ \boldsymbol{y} \in \mathcal{H}_{\boldsymbol{z}}\} \ .$$

The operator $P_{\mathcal{D}}$ is defined in (15), and thus we have

$$h^\star(\boldsymbol{p}) = \sup_{\boldsymbol{y}} \left\{ \langle \boldsymbol{p}, \Pi_{\mathcal{N}}\boldsymbol{y} + \pi_{\mathcal{D}^\star}\rangle - \frac{1}{2}\langle \Pi_{\mathcal{N}}\boldsymbol{y} + \pi_{\mathcal{D}^\star}, \Pi_{\mathcal{N}}\boldsymbol{y} + \pi_{\mathcal{D}^\star}\rangle \right\} \ .$$

Since the orthogonal projection operators are self-adjoint, *i.e.*, $\Pi_{\mathcal{N}}^{\star} = \Pi_{\mathcal{N}}$ and $\Pi_{\mathcal{N}}\Pi_{\mathcal{N}} = \Pi_{\mathcal{N}}$, we can write

$$h^{\star}(\boldsymbol{p}) = \sup_{\boldsymbol{y}} \Big\{ \langle \Pi_{\mathcal{N}}\boldsymbol{p}, y\rangle + \langle \boldsymbol{p}, \pi_{\mathcal{D}^{\star}}\rangle - \frac{1}{2}\langle \boldsymbol{y}, \Pi_{\mathcal{N}}\boldsymbol{y}\rangle - \langle \boldsymbol{y}, \Pi_{\mathcal{N}}\pi_{\mathcal{D}^{\star}}\rangle - \frac{1}{2}\langle \pi_{\mathcal{D}^{\star}}, \pi_{\mathcal{D}^{\star}}\rangle \Big\} \ . \tag{20}$$

The term $\Pi_{\mathcal{N}}\pi_{\mathcal{D}^{\star}}$ equals zero. Taking the gradient of the above expression, we have that

$$\Pi_{\mathcal{N}}\boldsymbol{y} = \Pi_{\mathcal{N}}\boldsymbol{p} \ .$$

A solution of the above equation is $\boldsymbol{y} = \boldsymbol{p}$. Then $h^{\star}(\boldsymbol{p})$ can be uniquely defined by substituting $\boldsymbol{y} = \boldsymbol{p}$ in (20), since it is the optimal value of a concave function. Finally, the conjugate of $h(\boldsymbol{z})$ is given by

$$h^{\star}(\boldsymbol{p}) = \frac{1}{2}\langle \boldsymbol{p}, \Pi_{\mathcal{N}}\boldsymbol{p}\rangle + \langle \boldsymbol{p}, \pi_{\mathcal{D}^{\star}}\rangle - \frac{1}{2}\langle \pi_{\mathcal{D}^{\star}}, \pi_{\mathcal{D}^{\star}}\rangle \ .$$

VIII. 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 (e.g., ADMM), and extensions to broader problem classes (e.g., tracking, soft constraints).

Acknowledgments

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013)/ ERC Grant Agreement n. 307608.

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

References

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

[2] J.-B. Baillon and G. Haddad. Quelques propriétés des opérateurs angle-bornés et n-cycliquement monotones. *Israel J. Math.*, 1977.

[3] H.H. Bauschke and P.L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, 2011.

[4] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, 2003.

[5] S. Boyd and N. Parikh. Monotone operators. EE364b Course Notes.

[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.*, 2011.
[7] S.P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
[8] R.J. Bruck. On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in hilbert space. *J. Math. Anal. Appl.*, 1977.
[9] P.L. Combettes and J-C. Pesquet. Proximal Splitting Methods in Signal Processing. Technical Report arXiv:0912.3522, 2009.
[10] J. Eckstein and D.P. Bertsekas. On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program.*, 1992.
[11] J.E. Esser. *Primal Dual Algorithms for Convex Models and Applications to Image Restoration, Registration and Nonlocal Inpainting*. 2010.
[12] T. Goldstein, B. O'Donoghue, and S. Setzer. Fast Alternating Direction Optimization Methods. *arXiv.org*, 2012.
[13] J.B. Hiriart-Urruty and C. Lemarechal. *Convex Analysis and Minimization Algorithms: Part 2: Advanced Theory and Bundle Methods*. Springer, 2010.
[14] R.E. Kalman. Contributions to the theory of optimal control. *Boletin de la Sociedad Matematica Mexicana*, 1960.
[15] A. Moradifam and A. Nachman. Convergence of the alternating split Bregman algorithm in infinite-dimensional Hilbert spaces.
[16] B. O'Donoghue and E.J. Candes. Adaptive Restart for Accelerated Gradient Schemes. *arXiv.org*, 2012.
[17] B. O'Donoghue, G. Stathopoulos, and S. Boyd. A splitting method for optimal control. *IEEE Transactions on Control Systems Technology*, 2012.
[18] G.B. Passty. Ergodic convergence to a zero of the sum of monotone operators in hilbert space. *J. Math. Anal. Appl.*, 1977.
[19] J. A. Primbs. The analysis of optimization based controllers. *Automatica*, 2001.
[20] P.O.M. Scokaert and J. B. Rawlings. Constrained Linear Quadratic Regulation. *IEEE Transactions on Automatic Control*, 1998.
[21] B.F. Svaiter. On weak convergence of the douglas-rachford method. *SIAM J. Control and Optimization*, 2011.
[22] M. Sznaier and M.J. Damborg. Suboptimal control of linear systems with state and control inequality constraints. In *IEEE Conference on Decision and Control*, 1987.
[23] P. Tseng. Applications of splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM J. Control Optim.*, 1991.