Fast Alternating Minimization Algorithm for Model Predictive Control

Ye Pu * **Melanie N. Zeilinger** ** **Colin N. Jones** *

* *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland (e-mail: {y.pu,colin.jones}@epfl.ch)*
** *Department of Electrical Engineering and Computer Sciences, UC Berkeley, CA 94702, USA (e-mail: mzeilinger@berkeley.edu)*

Abstract: In this work, we apply the fast alternating minimization algorithm (FAMA) to model predictive control (MPC) problems with polytopic and second-order cone constraints. We present a splitting strategy, which speeds up FAMA by reducing each iteration to simple operations. We show that FAMA provides not only good performance for solving MPC problems when compared to other alternating direction methods, but also superior theoretical properties. Specifically, we derive complexity bounds on the number of iterations for both dual and primal variables, which are of particular relevance in the context of real-time MPC to bound the required online computation time. For MPC problems with polyhedral and ellipsoidal constraints, an off-line pre-conditioning method is presented to further improve the convergence speed of FAMA by decreasing the complexity upper-bounds and enlarging the step-size of the algorithm. Finally, we demonstrate the performance of FAMA compared to other alternating direction methods using a quadroter example.

1. INTRODUCTION

The strength of Model Predictive Control (MPC) is that it allows constraints on the states and control inputs to be integrated into the controller design. However, the cost is that at each sampling time an optimization problem needs to be solved, which has traditionally restricted MPC to applications with slow dynamics and long sampling times. This limitation has given rise to an increasing interest in the development of new methods to either improve the on-line computation, driven by the increase in computational power of hardware and newly developed optimization techniques, or to approximate the optimum with a sub-optimal but stabilizing solution.

One technique to reduce the on-line computation is multi-parametric programming, which pre-computes the solution for every state off-line, see Alessio and Bemporad [2009] for more details and references. Zeilinger et al. [2011] presents a method combining explicit MPC with on-line computation. However, all explicit and approximate explicit methods are limited to small-scale problems.

For medium and larger scale MPC problems, on-line computation methods are used. Various approaches have been proposed to improve the on-line computation time. The authors in Richter et al. [2012] employ the fast gradient method introduced in Nesterov [1983] to solve MPC problems with box constraints on inputs. Efficient implementations of interior-point methods have been studied in Wang and Boyd [2010] and Domahidi et al. [2012]. Accelerated gradient methods with dual decomposition are investigated in Kögel and Findeisen [2011] and in Giselsson et al. [2013] in the context of distributed MPC. Ferreau et al. [2008] and Bartlett and Biegler [2006] present efficient active set methods for MPC.

This work applies accelerated alternating direction methods for solving MPC problems. Alternating direction methods offer a powerful tool for general mathematical programming and optimization and have attracted a lot of attention in recent years, see e.g. Goldstein et al. [2012], Boyd et al. [2011] and Combettes and Pesquet [2011]. An important advantage of alternating direction methods is that they split a complex minimization problem into simple sub-problems and solve them in an alternating manner. For instance, for a problem with multiple objectives, instead of computing the descent direction of the sum of several objectives, alternating direction methods take a combination of the descent directions of each objective. This can save a significant amount of time, in particular when the objectives have different properties, for instance a quadratic function and an l_1-norm.

A variety of different alternating direction methods exist, using different assumptions on the problem set-up and having different properties, see e.g. Goldstein et al. [2012] and Combettes and Pesquet [2011] for an overview. The alternating direction method of multipliers (ADMM) has received most attention recently and was demonstrated to perform well in practice, e.g. in O'Donoghue et al. [2013], where it was shown to solve optimal control problems both rapidly and robustly. However, the convergence rate of ADMM is only $O(\frac{1}{k})$, and until now no theoretical complexity bound on the number of iterations has been shown. The complexity bound is important in the context of real-time MPC in order to ensure that the optimization problem can be solved in the available amount of time. An

⋆ The research of M.N. Zeilinger has received funding from the EU FP7 under grant agreement no. PIOF-GA-2011-301436-"COGENT".

accelerated variant of ADMM, namely the fast alternating direction method of multipliers (FADMM) in Goldstein et al. [2012], is based on the same assumptions and has the same convergence rate as ADMM but provides better performance than ADMM.

In this paper, we propose the use of the fast (accelerated) alternating minimization algorithm (FAMA) in Tseng [1991] and Goldstein et al. [2012], for the solution of MPC problems offering superior theoretical properties while providing similar or even better performance. Compared to ADMM, FAMA requires stronger assumptions on the objectives of the optimization problems, which can, however, be satisfied by standard MPC problem formulations with polytopic and second-order cone constraints. In return, FAMA offers a better convergence rate of $O(\frac{1}{k^2})$ and provides complexity bounds on the number of iterations. The main contributions of this work are:

- Second-order cone constraints: In this work, we consider both polytopic and second-order cone constraints, which allows for solving, for example, MPC problems with ellipsoidal constraints.
- Splitting strategy: We present a formulation of MPC problems that satisfies the assumptions of FAMA and propose a splitting strategy to reduces each iteration of FAMA to simple operations.
- Convergence rate: We show that FAMA offers a better convergence rate of $O(\frac{1}{k^2})$ compared with ADMM and FADMM.
- Complexity bound: Complexity upper-bounds on the number of iterations to achieve a certain solution accuracy for both primal and dual variables are derived.
- Preconditioning: For MPC problems with polytopic and ellipsoidal constraints, we propose an off-line pre-conditioning method to further improve the convergence speed of FAMA. The method enlarges the step-size of the algorithm by scaling the polytopic constraints and reshaping the ellipsoidal constraints.

All properties above are demonstrated for the simulation example of a quadroter.

2. FAST ALTERNATING MINIMIZATION ALGORITHM (FAMA) FOR MPC

2.1 Notation

Let f be a strongly convex function. σ_f denotes the convexity modulus of f, i.e. $\langle p-q, x-y\rangle \geq \sigma_f \|x-y\|^2$, where $p \in \partial f(x)$, $q \in \partial f(y)$ and $\partial(\cdot)$ denotes the set of sub-gradients of the function at a given point. The operators max, $\leq$ and $\geq$ are defined to work on vectors as well as scalars. For vectors, the operators are defined to be element-wise. Let C be a matrix. $\rho(C)$ denotes the largest eigenvalue of C^TC. For a positive definite matrix H, $\lambda_{min}(H)$ denotes the smallest eigenvalue of H. Let $\mathbb{C}$ be a convex set. The indicator function $I_{\mathbb{C}}(\sigma)$ on $\mathbb{C}$ is defined to be zero if $\sigma \in \mathbb{C}$ and infinity otherwise. Let $\mathbb{C}$ be a convex cone. The set $\mathbb{C}' = \{w \mid v^Tw \leq 0,\ \forall v \in \mathbb{C}\}$ denotes the polar cone of $\mathbb{C}$. The set $\mathbb{C}^\star = \{w \mid v^Tw \geq 0,\ \forall v \in \mathbb{C}\}$ denotes the dual cone of $\mathbb{C}$. A cone $\mathbb{C}$ is called self-dual, if $\mathbb{C} = \mathbb{C}^\star$.

2.2 Fast Alternating Minimization Algorithm (FAMA)

In Tseng [1991], an alternating direction method called alternating minimization algorithm (AMA) is proposed. In this section, we apply the accelerated variant of AMA, named fast alternating minimization algorithm in Goldstein et al. [2012], to MPC problems. Before going into the details of the algorithm, the difference between FAMA and ADMM as well as its accelerated variant FADMM (see Algorithm 8 in Goldstein et al. [2012]) are highlighted. ADMM and FADMM require the objectives to be convex functions for convergence, and with this assumption both algorithms offer the same convergence rate $O(\frac{1}{k})$. In addition, the convergence of FADMM can only be guaranteed by integrating the restarting rules proposed in Goldstein et al. [2012]. FADMM can have a faster convergence rate, if both objectives are strongly convex. FAMA requires the stronger assumption that one objective needs to be strongly convex. In return, it achieves a faster convergence rate $O(\frac{1}{k^2})$. The second difference is that FAMA provides a complexity upper-bound on the number of iterations for a given accuracy, which is useful for real-time MPC problems, while for ADMM and FADMM such a bound is not available. The third difference is that ADMM and FADMM don't require any condition on the step-size. Theoretically, any positive step-size guarantees convergence, however, not any positive step-size results in good convergence speed. The question of how to best tune the step-size still remains largely unclear and is usually done by trial and error. FAMA, in contrast, requires the step-size to be smaller than the reciprocal of the Lipschitz constant of the gradient of the dual objectives. This condition simplifies the selection of the step-size. The complexity bound together with the condition on the step-size allow for pre-conditioning to speed up the algorithm, which will be discussed in Section 4. The differences between ADMM, FADMM and FAMA are summarized in Table 1.

In the following, we show how FAMA can be applied to MPC to exploit the properties discussed above. We consider an MPC problem for a system with linear and time-invariant dynamics, state- and input-constraints in the form of polytopic and/or second-order cone constraints and quadratic stage and terminal costs. By eliminating all state variables and moving the inequality constraints to the cost in the form of indicator functions, MPC problems of this class can be reformulated in the following form with one strongly and one weakly convex objective, suitable for the application of FAMA.

Problem 2.1.

$$\min_{\mathbf{u}} \quad \underbrace{\mathbf{u}^T H \mathbf{u} + h^T \mathbf{u}}_{f(\mathbf{u})} + \underbrace{\sum_{i=1}^{M} I_{\mathbb{C}_i}(\sigma_i)}_{g(\sigma)}$$
$$s.t. \quad C_i \mathbf{u} - d_i = \sigma_i, \quad i = 1, \cdots, M.$$

where $\mathbf{u} = [u_0^T, u_1^T, \cdots, u_{N-1}^T]^T \in \mathbb{R}^{N\cdot m}$ denotes the sequence of inputs over the control horizon N and $\sigma = [\sigma_1^T, \cdots, \sigma_M^T]^T \in \mathbb{R}^{N_\sigma}$ are auxiliary variables. $\mathbb{C}_i$ denote the constraints on the states and inputs. In this paper, $\mathbb{C}_i$ are given either by the non-negative orthand, i.e., $\mathbb{C}_i := \{v \mid v \geq 0\}$, or simplified second-order cone constraints,

Methods	Convergence rate	Assumptions	Complexity bound	Condition on step-size
ADMM	$O(\frac{1}{k})$	both objectives convex	no	no
FADMM	$O(\frac{1}{k})/O(\frac{1}{k^2})$	both objectives convex/strongly convex	no	no/yes
FAMA	$O(\frac{1}{k^2})$	one objective strongly convex	yes	yes

Table 1. Summary of assumptions and properties of ADMM, FADMM and FAMA.

i.e., $\mathbb{C}_i := \{[v_1, v_2] | \ |v_1|_2 \geq v_2\}$. Note that these cover all polytopic and second-order cone constraints on $\mathbf{u}$ by involving the affine coupling $C_i\mathbf{u} - d_i = \sigma_i$. Both the non-negative orthant and the second-order cone are self-dual cones. This fact will be used in the proof of Theorem 3.6.

Remark 2.2. The matrix H is independent of the initial state $\bar{x}$ and the matrix h is a linear function of $\bar{x}$.

Assumption 2.3. The matrix H is positive definite, i.e., $H \succ 0$.

Remark 2.4. Assumption 2.3 holds, if a strictly convex cost on the inputs is chosen and the dynamical system in the MPC problem is controllable.

Remark 2.5. If Assumption 2.3 holds, the first objective function f is strongly convex and the convexity modulus σ_f is given by the minimum eigenvalue of the matrix H, i.e., $\sigma_f = \lambda_{min}(H)$.

Remark 2.6. The second objective $g(\sigma)$ is an indicator function of a convex cone, which is convex. If Assumption 2.3 holds, which means that the first objective function $f(\mathbf{u})$ is strongly convex, then all assumptions required by FAMA (see Tseng [1991] and Goldstein et al. [2012]) are satisfied.

Algorithm 1 Fast alternating minimization algorithm (FAMA)

Require: Initialize $\alpha^0 = 1$, $\alpha^1 = (1+\sqrt{5})/2$, $\lambda^0 = \lambda^{start} \in \mathbb{R}^{N_\sigma}$, and $\tau < \sigma_f/\rho(C) = \lambda_{min}(H)/\rho(C)$.

loop

1: for $k = 1, 2, \cdots$ do

2: $\mathbf{u}^k = argmin \quad \mathbf{u}^T H\mathbf{u} + h^T\mathbf{u} - \sum_{i=1}^{M} {\lambda_i^{k-1}}^T C_i\mathbf{u}$

3: $\hat{\mathbf{u}}^k = \mathbf{u}^k(\alpha^{k-1} + \alpha^k - 1)/\alpha^k - \mathbf{u}^{k-1}(\alpha^{k-1} - 1)/\alpha^k$

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

loop

5: for $i = 1, \cdots, M$ do

6: $\sigma_i^k = \mathbf{Pr}_{\mathbb{C}_i}(C_i\hat{\mathbf{u}}^k - \frac{1}{\tau}\hat{\lambda}_i^k - d_i)$

7: $\lambda_i^k = \hat{\lambda}_i^k + \tau(d_i - C_i\hat{\mathbf{u}}^k + \sigma_i^k)$

8: $\hat{\lambda}_i^{k+1} = \lambda_i^k + (\alpha^k - 1)(\lambda_i^k - \lambda_i^{k-1})/\alpha^{k+1}$

9: end for

end loop

10: end for

end loop

We apply FAMA to MPC Problem 2.1 resulting in Algorithm 1, where the matrix C is $[C_1^T, \cdots, C_M^T]^T$. The advantage of the splitting strategy in Problem 2.1 is that the two objectives $f(\mathbf{u})$ and $g(\sigma)$ are very easy to minimize separately. The solution to the unconstrained minimization problem in Step 2 can be obtained analytically, i.e. $\mathbf{u}^k = \frac{1}{2}H^{-1}(\sum_{i=1}^{M} C_i^T\lambda_i^{k-1} - h)$, where the inverse H^{-1} can be computed off-line. Step 6 involves basic projections onto the non-negative orthant and simplified second-order cone. We denote the projection operation as $\mathbf{Pr}_{\mathbb{C}}(\cdot)$. For the non-negative orthant, the projection is:

$$\mathbf{Pr}_{\mathbb{C}}(v) = \max\{0, v\}. \tag{1}$$

For the second-order cone, the projection is:

$$\mathbf{Pr}_{\mathbb{C}}([v_1, v_2]) = \begin{cases} [v_1, v_2] & \text{if } |v_1|_2 \leq v_2 \\ \frac{v_2+|v_1|_2}{2|v_1|_2}[v_1, |v_1|_2] & \text{if } |v_1|_2 > v_2,\ v_1 \neq 0 \\ [0, 0] & \text{if } |v_1|_2 \leq -v_2. \end{cases} \tag{2}$$

The projections in (1) and (2) are computationally cheap and either reduce to simply a clipping or a scaling operation.

Remark 2.7. Step 2 and 3 in Algorithm 1 are equivalent to $\mathbf{u}^k = argmin \quad \mathbf{u}^T H\mathbf{u} + h^T\mathbf{u} - \sum_{i=1}^{M} {\hat{\lambda}_i^{k-1}}^T C_i\mathbf{u}$. By splitting this step into two steps, $\mathbf{u}^k$ is expressed as a function of λ^k, i.e. $\mathbf{u}^k = \frac{1}{2}H^{-1}(C^T\lambda^k - h)$. This allows us to derive the complexity bound on $|\mathbf{u}^k - \mathbf{u}^\star|$, which will be presented in Section 3.

3. COMPEXITY BOUNDS OF FAMA FOR MPC

In this section, we will derive the complexity upper-bounds on the number of iterations to achieve a certain solution accuracy for both the primal and dual sequences $\{\mathbf{u}^k\}$ and $\{\lambda^k\}$ generated by Algorithm 1. The complexity upper-bounds are important for real-time MPC by providing a certificate that a solution of pre-specified sub-optimality can be obtained within the available computation time.

3.1 Notation

Let $f : \Theta \to \Omega$ be a function. The conjugate function of f is defined as $f^\star(y) = \sup_{x\in\Theta}(y^T x - f(x))$. It holds that $p \in \partial f(q) \Leftrightarrow q \in \partial f^\star(p)$. If f is a strongly convex function with the convexity modulus σ_f, then the gradient of the conjugate function of f has a Lipschitz constant $L(\nabla f^\star) = \sigma_f^{-1}$. We refer to Bertsekas et al. [2003] and Boyd and Vandenberghe [2004] for details on the definitions and properties above.

3.2 Fast iterative shrinkage-thresholding algorithm

As shown in Goldstein et al. [2012], FAMA corresponds to applying the fast iterative shrinkage-thresholding algorithm (FISTA) in Beck and Teboulle [2009] to the dual problem. We will therefore first introduce the convergence results for FISTA in the following, which will allow us to derive an upper bound on the number of iterations for FAMA. FISTA solves problems of the following form:

Problem 3.1.

$$\min \quad F(z) + G(z), \quad z \in \mathbb{R}^n. \tag{3}$$

Note that this is a special case of the general problem formulation addressed by alternating direction methods in Goldstein et al. [2012], imposing the additional coupling $z_1 - z_2 = 0$ on the more general objective min $F(z_1) + G(z_2)$.

Assumption 3.2.
- F is a continuous convex function with Lipschitz continuous gradient $L(\nabla F)$:
$$|\nabla F(z_1) - \nabla F(z_2)| \leq L(\nabla F)|z_1 - z_2| \quad \forall z_1, z_2 \in \mathbb{R}^n.$$
- G is a convex function.

The conceptual idea of the fast iterative shrinkage algorithm is to build a linearization and regularization of the differentiable function part in the objective at each iteration and to apply Nesterov's acceleration step to achieve $O(\frac{1}{k^2})$ convergence rate. Lemma 3.3 states the complexity upper-bound on the number of iterations for FISTA.

Lemma 3.3. (Beck and Teboulle [2009], Theorem 4.4). Let $\{z^k\}$ be generated by applying FISTA to Problem 3.1. If Assumption 3.2 holds, then for any $k \geq 1$ we have:
$$J(z^k) - J(z^\star) \leq \frac{2L(\nabla F)|z^{start} - z^\star|^2}{(k+1)^2} \tag{4}$$
where $J(z) = F(z) + G(z)$, and z^{start} and $z^\star$ denote the starting point and the optimizer of Problem 3.1, respectively.

FISTA cannot directly be applied to Problem 2.1, since the matrix C is not equal to an identity matrix. However, as will be shown in Theorem 3.6, the dual of Problem 2.1 satisfies the assumptions of FISTA and applying FAMA to Problem 2.1 is equivalent to applying FISTA to the dual. Hence, the complexity bound for FISTA can be extended to FAMA. Before we present the main results in Theorem 3.6, we state two lemmas, which will be used in the proof of Theorem 3.6.

Lemma 3.4. Let $\mathbb{C}$ be a convex cone. The conjugate function of the indicator function of the set $\mathbb{S} := \{v | -v \in \mathbb{C}\}$ is equal to the indicator function of the dual cone of $\mathbb{C}$, i.e., $I^\star_{\mathbb{S}}(v) = I_{\mathbb{C}^\star}(v)$.

Proof. By the definition of a convex cone, the set $\mathbb{S}$ is still a convex cone. Example 7.3.5 in Bertsekas et al. [2003] shows $I^\star_{\mathbb{S}}(v) = I_{\mathbb{S}'}(v)$, where $\mathbb{S}'$ denotes the polar cone of $\mathbb{S}$. By the definitions of the polar cone and the dual cone, we know $\mathbb{S}' = \{w \mid w'v \leq 0,\ -v \in \mathbb{C}\} = \{w \mid w'v \geq 0,\ v \in \mathbb{C}\} = \mathbb{C}^\star$, and the result $I^\star_{\mathbb{S}}(v) = I_{\mathbb{C}^\star}(v)$ follows.

Lemma 3.5. Let $\mathbb{C}$ be the non-negative orthant $\mathbb{C} := \{v \mid v \geq 0\}$ or a second order cone $\mathbb{C} := \{[v_1, v_2] | \ |v_1| \leq v_2\}$. For any $v \in \mathbb{R}^{N_{\mathbb{C}}}$, the point $z = \mathbf{Pr}_{\mathbb{C}}(v) - v$ satisfies $z \in \mathbb{C}$.

Proof. For the non-negative orthant, it is easy to show that $z = \mathbf{Pr}_{\mathbb{C}}(v) - v = \max\{0, v\} - v \geq 0$. For a second-order cone, we denote $z = [z_1, z_2]$ and show that $|z_1| \leq z_2$ holds for the three cases in equation (2). For the first case, it can easily be verified that $|z_1| \leq z_2$. For the second case, we have:
$$|z_1|_2 = \left| \frac{v_2 + |v_1|_2}{2|v_1|_2} v_1 - v_1 \right|_2 = \left| \frac{v_2 - |v_1|_2}{2} \right|,$$
$$z_2 = \frac{v_2 + |v_1|_2}{2|v_1|_2}|v_1|_2 - v_2 = \frac{|v_1|_2 - v_2}{2}.$$
Since in this case it holds that $|v_1|_2 > v_2$, we get $|z_1|_2 \leq z_2$. For the third case, we have $|z_1|_2 = |v_1|_2$ and $z_2 = -v_2$. Since $|v_1|_2 \leq -v_2$, we prove $|z_1|_2 \leq z_2$.

Theorem 3.6. Consider Problem 2.1. Let $\{\mathbf{u}^k\}$ and $\{\lambda^k\}$ be generated by Algorithm 1, where $\lambda^k = [\lambda_1^T, \cdots, \lambda_M^T]^T$ and λ_i are the Lagrange multipliers associated with the constraint $C_i\mathbf{u} - d_i = \sigma_i$ at iteration k. If Assumption 2.3 is satisfied, then for any $k \geq 1$
$$D(\lambda^\star) - D(\lambda^k) \leq \frac{2\rho(C)|\lambda^{start} - \lambda^\star|^2}{\lambda_{min}(H)(k+1)^2}. \tag{5}$$
If $\lambda_i^{start} \in \mathbb{C}_i$ for all $i = 1, \cdots, M$, then $\lambda_i^k \in \mathbb{C}_i$ for all $k \geq 1$ and $i = 1, \cdots, M$ and
$$|\mathbf{u}^{k+1} - \mathbf{u}^\star|^2 \leq \frac{4\rho(C)|\lambda^{start} - \lambda^\star|^2}{\lambda^2_{min}(H)(k+1)^2}, \tag{6}$$
where λ^{start} and $\lambda^\star$ denote the starting point and the optimizer, respectively.

Proof. Theorem 2 in Goldstein et al. [2012] shows that applying FAMA to Problem 2.1 is equivalent to applying FISTA to the dual problem of Problem 2.1. The dual function of Problem 2.1 is:
$$\begin{aligned} D(\lambda) &= \underbrace{-f^\star(C^T\lambda)}_{-F(\lambda)} + \underbrace{d^T\lambda - g^\star(-\lambda)}_{-G(\lambda)} \\ &= \underbrace{-\frac{1}{4}\lambda^T C H^{-1} C^T \lambda + \frac{1}{2} h^T H^{-1} C^T \lambda - \frac{1}{4} h^T H^{-1} h}_{-F(\lambda)} \\ &\quad + \underbrace{d^T\lambda - \sum_{i=1}^{M} I^\star_{-\lambda_i \in \mathbb{C}_i}}_{-G(\lambda)}. \end{aligned}$$

$F(\lambda)$ and $G(\lambda)$ are both convex, since the conjugate functions and linear functions as well as their weighted sum are always convex (conjugate function is the point-wise supremum of a set of affine functions). By Assumption 3.2, $f(x)$ is strongly convex with $\sigma_f = \lambda_{min}(H)$. By the property of the conjugate function, a Lipschitz constant of $\nabla f^\star$ is given by:
$$L(\nabla f^\star) = \sigma_f^{-1} = \frac{1}{\lambda_{min}(H)}.$$
Then, we get a Lipschitz constant of ∇F:
$$L(\nabla F(\lambda)) = \sigma_f^{-1} \cdot \rho(C) = \frac{\rho(C)}{\lambda_{min}(H)}.$$
By Lemma 3.3, it follows that the sequence $\{\lambda_k\}$ generated by Algorithm 1 satisfies the complexity bound (5).

The second step is to prove that if $\lambda_i^{start} \in \mathbb{C}_i$ for all $i = 1, \cdots, M$, then $\lambda_i^k \in \mathbb{C}_i$ for all $k \in \mathbb{N}$ and $i = 1, \cdots, M$. From Step 4 and 5 in Algorithm 1, we know that $\lambda_i^k = \hat{\lambda}_i^k + \tau(d_i - C_i\mathbf{u}^k + \sigma_i^k) = \tau(\mathbf{Pr}_{\mathbb{C}_i}(C_i\mathbf{u}^k - \frac{1}{\tau}\hat{\lambda}_i^k - d_i) - (C_i\mathbf{u}^k - \frac{1}{\tau}\hat{\lambda}_i^k - d_i))$. By the fact that $\tau > 0$ and Lemma 3.5, we can conclude $\lambda_i^k \in \mathbb{C}_i$ for all $k = 1, \cdots$ and $i = 1, \cdots, M$.

The last step is to prove inequality (6). From Step 1 in Algorithm 1 we have:
$$\mathbf{u}^{k+1} = \frac{1}{2} H^{-1}(C^T\lambda^k - h)$$
which implies:

$$
\begin{aligned}
&|\mathbf{u}_{k+1} - \mathbf{u}^\star|^2 \\
&= |\frac{1}{2}H^{-1}C^T(\lambda^k - \lambda^\star)|^2 \\
&\le \frac{1}{2\lambda_{min}(H)}(\lambda^k - \lambda^\star)^T CH^{-1}C^T(\lambda^k - \lambda^\star) \\
&= \frac{2}{\lambda_{min}(H)}[\frac{1}{4}\lambda^{k^T}CH^{-1}C^T\lambda^k - (\frac{1}{2}h^TH^{-1}C^T + d^T)\lambda_k \\
&\quad - \frac{1}{4}\lambda^{\star^T}CH^{-1}C^T\lambda^\star + (\frac{1}{2}h^TH^{-1}C^T + d^T)\lambda^\star \\
&\quad + \frac{1}{2}((\lambda^\star)^T CH^{-1}C^T\lambda^\star - \lambda^{k^T}CH^{-1}C^T\lambda^\star) \\
&\quad + (\frac{1}{2}h^TH^{-1}C^T + d^T)(\lambda^k - \lambda^\star)].
\end{aligned}
$$

Since we know that all $\mathbb{C}_i$ are self-dual cones, i.e. $\mathbb{C}_i^\star = \mathbb{C}_i$, Lemma 3.4 implies that the dual function can be simplified as:

$$
\begin{aligned}
D(\lambda) = &-\frac{1}{4}\lambda^T CH^{-1}C^T\lambda + (\frac{1}{2}h^TH^{-1}C^T + d^T)\lambda \\
&- \frac{1}{4}h^TH^{-1}h - \sum_{i=1}^{M} I_{\lambda_i \in \mathbb{C}_i},
\end{aligned}
$$

and the gradient of the dual function for $\lambda_i \in \mathbb{C}_i$ is :

$$
\nabla D(\lambda) = -\frac{1}{2}C(H^{-1})^TC^T\lambda + (\frac{1}{2}C(H^{-1})^Th + d). \tag{7}
$$

Since we have shown that $\lambda_i^k \in \mathbb{C}_i$ for all $k \ge 0$ and $i = 1, \cdots, M$, we obtain:

$$
\begin{aligned}
|\mathbf{u}_{k+1} - \mathbf{u}^\star|^2 \le \frac{2}{\lambda_{min}(H)}&(D(\lambda^\star) - D(\lambda^k) \\
&- \nabla D^T(\lambda^\star)(\lambda^k - \lambda^\star)).
\end{aligned}
$$

By optimality, we conclude:

$$
\begin{aligned}
&\frac{2}{\lambda_{min}(H)}(D(\lambda^\star) - D(\lambda_k) - \nabla D(\lambda^\star)(\lambda_k - \lambda^\star)) \\
&\le \frac{2}{\lambda_{min}(H)}(D(\lambda^\star) - D(\lambda_k)) \\
&\le \frac{4\rho(C)|\lambda^{start} - \lambda^\star|^2}{\lambda_{min}^2(H)(k+1)^2}.
\end{aligned}
$$

Remark 3.7. Theorem 3.6 can be directly extended to the MPC problem with positive semi-definite cone constraints, since a positive semi-definite cone is also a self-dual cone.

Remark 3.8. The proof of inequality (6) is an extension of the proof in Giselsson et al. [2013] for the case of polytopic constraints in the context of distributed MPC problems.

4. PRE-CONDITIONING

Pre-conditioning has been observed to offer significant compuational speedups in gradient based methods in Richter et al. [2012] and Giselsson [2013]. In this section, we present a pre-conditioning technique to improve the performance of FAMA, when applied to MPC problems with polytopic and ellipsoidal (a special case of second-order cone) constraints. The idea of the method is to enlarge the step-size and decrease the complexity bound of the algorithm by scaling the polytopic constraints and reshaping the ellipsoidal constraints. It is important to point out that for the case with only polytopic constraints the proposed pre-conditioning does not change the original constraints and thereby the optimal solution, whereas for the case with both polytopic and ellipsoidal constraints, the solution is modified since the shape of the ellipsoidal constraints is redesigned. Feasibility of the solution is maintained, by choosing the new ellipsoidal constraints as inner-approximations of the original ellipsoids. If the ellipsoidal constraint represents a terminal state constraint, the proposed pre-conditioning method maintains the stability properties. Recall the condition on the step-size $\tau < \lambda_{min}(H)/\rho(C)$ and the complexity bound in Theorem 3.6. The value $\rho(C)$ affects them in the way that the smaller $\rho(C)$ is, the larger the step-size can be and the smaller the complexity bound will be. The constraints are therefore pre-conditioned to reduce the value of $\rho(C)$ and thereby enlarge the step-size and decrease the positive constant in the complexity bound in Theorem 3.6. The pre-conditioning is hence enabled by the existence of upper-bounds on the step-size and on the number of iterations. Since ADMM and FADMM do not provide such bounds, it is unclear how to provide a similar pre-conditioning method for them. The derived theoretical properties of FAMA therefore offer significant benefits. They not only provide a bound on the required on-line computation time, but also allow for tuning the algorithm and improve its performance for the particular problem at hand.

Consider the discrete-time linear time-invariant system $x_{t+1} = Ax_t + Bu_t$, where x_t and u_t denote the state and input at time t. Let $\mathbb{X}$ and $\mathbb{U}$ be the state and input constraints and K be a linear control law, such that $A + BK$ is stable.

Definition 4.1. (Positive invariant (PI) set): A set $\mathbb{S} \subseteq \mathbb{R}^n$ is a positively invariant set of system $x_{t+1} = Ax_t + BKx_t$, if $Ax_t + BKx_t \in \mathbb{S}$ and $Kx_t \in \mathbb{U}$ for all $x_t \in \mathbb{S}$.

In order to simplify the notation, we assume that Problem 2.1 only has two constraints, a polytopic constraint $C_1\mathbf{u} - d_1 \ge 0$ and an ellipsoidal constraint $|C_2\mathbf{u} - d_2| \le 1$. We consider the more difficult set-up, where the ellipsoidal constraint originates from a state constraint $|Ex_t - e| \le 1$, with $E \succ 0$. Since x_t can be represented by a linear combination of the control sequence $\mathbf{u}$ and the initial state $\bar{x}$, i.e., $x_t = M_1\mathbf{u} + M_2\bar{x}$, it follows that $C_2 = EM_1$ and $d_2 = e - EM_2\bar{x}$. Note that ellipsoidal constraints on the input would directly be of the right form.

We introduce two positive-definite matrices P_1 and P_2, P_1 to scale the polytopic constraints $P_1C_i\mathbf{u} - P_1d_1 \ge 0$ and $P_2 \in \mathbb{R}^{n_{C_2} \times n_{C_2}}$ to reshape the ellipsoidal constraint $|P_2C_2\mathbf{u} - P_2d_2| \le 1$, where n_{C_1} and n_{C_2} denote the number of rows of the matrices C_1 and C_2, respectively. P_1 is set to be a diagonal matrix. The goal is to minimize $\rho(C)$, where $C = [C_1^T, C_2^T]^T$, which can be achieved by minimizing the condition number of C by the following minimization problem (see Chapter 3.1 in Boyd et al. [1994]). Let $W_1 = P_1^TP_1$ and $W_2 = P_2^TP_2$.

Problem 4.2.

$$
\begin{aligned}
\min_{\alpha, W_1, W_2} \quad & \alpha \\
s.t. \quad & \mu I \preceq \begin{bmatrix} C_1^T & C_2^T \end{bmatrix} \begin{bmatrix} W_1 & 0 \\ 0 & W_2 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \preceq \alpha I, \\
& W_1 \succ 0, \quad W_2 \succ 0,
\end{aligned}
$$

where μ is equal to the minimum eigenvalue of C^TC. By setting W_2 to be a positive definite diagonal matrix, the pre-conditioning does not modify the polytopic constraint. However, it changes the ellipsoidal constraint on x_t to be $|P_2Ex_t - P_2e| \le 1$. In order to guarantee that the solution given by the pre-conditioned problem is still a sub-optimal and feasible solution to Problem 2.1, the pre-conditioned ellipsoid is scaled to provide an inner approximation of the original ellipsoid. Problem 4.3 computes the maximum such inner approximation. The first LMI constraint in Problem 4.3 guarantees that the new scaled ellipsoidal constraint $|\frac{1}{\beta}P_2Ex_t - \frac{1}{\beta}P_2e| \le 1$ is contained in the original constraint $|Ex_t - e| \le 1$, (see Chapter 8.4.2 in Boyd and Vandenberghe [2004]). Note that after solving Problem 4.2, the matrix P_2 in Problem 4.3 is not an optimization variable but a constant.

Problem 4.3.

$$\min_{\beta,\omega} \quad \beta$$
$$s.t. \quad \begin{bmatrix} -\omega+1 & 0 & 0 \\ 0 & \omega I & \beta(P_2E)^{-1} \\ 0 & \beta(P_2E)^{-1} & (E^TE)^{-1} \end{bmatrix} \succeq 0,$$
$$\omega \ge 0.$$

Since $x_t = M_1\mathbf{u} + M_2\bar{x}$, the new constraint on $\mathbf{u}$ is set to be $|\frac{1}{\beta}P_2C_2\mathbf{u} - \frac{1}{\beta}P_2d_2| \le 1$.

In the case that the ellipsoidal constraint has to be an invariant set, which is generally the case for quadratic terminal sets in MPC, an additional invariance condition has to be imposed on W_2 in Problem 4.2. We exemplify this procedure for an ellipsoidal constraint originating from an invariant ellipsoidal state constraint of the form $\mathbb{X}_f = \{x_N | x_N^T P x_N < 1\}$, where $P \succ 0$. Invariance of the pre-conditioned ellipsoid can be ensured by enforcing the following constraint $(A+BK)^T {P^{\frac{1}{2}}}^T W_2 P^{\frac{1}{2}}(A+BK) - {P^{\frac{1}{2}}}^T W_2 P^{\frac{1}{2}} \preceq 0$ in Problem 4.2, which can be written as an LMI by using Schur complements. Note that in the case of a terminal set, if the state and input constraints $\mathbb{X}$ and $\mathbb{U}$ are polytopic sets, then the inner approximation in Problem 4.3 can be relaxed by only requiring the scaled new ellipsoid to be contained in $\mathbb{X} \cap K\mathbb{U}$. To summarize, all properties of the original MPC controller, such as invariance and stability of the closed-loop system, are maintained under the proposed pre-conditioning.

Remark 4.4. Since the matrices C_1, E and M_1 are independent of the initial state $\bar{x}$, the computation of the pre-conditioning matrices P_1 and P_2 and the parameter β can be performed off-line.

Remark 4.5. The pre-conditioning method introduced in this section can be easily extended to the case with more than two constraints and with ellipsoidal input constraints.

5. NUMERICAL EXAMPLE

This section illustrates the theoretical findings of the paper and demonstrates the performance of FAMA for solving MPC problems. We consider a quadroter model, see Mellinger and Kumar [2011], which is driven by four independently controlled rotors. In this experiment, we use a cascaded control structure and design an MPC controller to control the derivative of the height of the quadroter, the roll, pitch and yaw angles and the derivative of these angles, i.e $x = [\dot{z}, \alpha, \beta, \gamma, \dot{\alpha}, \dot{\beta}, \dot{\gamma}]^T$. The resulting linearized and discretized dynamics are defined by $x_{t+1} = Ax_t + Bu_t$. The state of the system is subject to constraints on the maximum angle, maximum angle velocity as well as maximum velocity in the z direction - these constraints are mainly chosen to ensure validity of the linearized model and have been specified as: $|\dot{z}| \le 1m/s$, $|\alpha| \le 10°$, $|\beta| \le 10°$, $|\dot{\alpha}| \le 15°$, $|\dot{\beta}| \le 15°$ and $|\dot{\gamma}| \le 60°$. The input constraint is $0 \le u \le 1$. The horizon of the MPC controller is set to $N = 25$. The terminal state x_N is subject to a positive invariant ellipsoidal terminal constraint.

In the simulation shown in Fig. 1, we collect 1000 randomly sampled initial states in the set $\{\bar{x} | [-0.5m/s, -5°, -5°, -60°, -5°/s, -5°/s, -30°/s]^T \le \bar{x} \le [0.5m/s, 5°, 5°, 60°, 5°/s, 5°/s, 30°/s]^T\}$ and compare the proposed FAMA algorithm (red line), as well as the FAMA algorithm with pre-conditioning presented in Section 4 (black line), against ADMM (green line) and FADMM (blue line) for solving the MPC problem with these 1000 initial states. The step-size for FAMA is set to $0.99 * \lambda_{min}(H)/\rho(C)$, while for ADMM and FADMM it is set to the best value obtained by manual tuning. Performance is measured by the percentage of samples, for which $|\mathbf{u}^k - \mathbf{u}^\star|/|\mathbf{u}^\star| < \delta$ after k iterations. In Fig. 1a, δ is set to 10^{-4} and in Fig. 1b to 10^{-6}. In both cases, FAMA with preconditioning shows the best performance, fastest convergence speed and good accuracy after few iterations. FAMA without preconditioning converges more slowly but still faster than ADMM and FADMM. Fig. 1b shows that the solution accuracy given by ADMM and FADMM is inferior to FAMA. They achieve a solution accuracy of $\delta = 10^{-6}$ in 1000 iterations for only 10% of the samples.

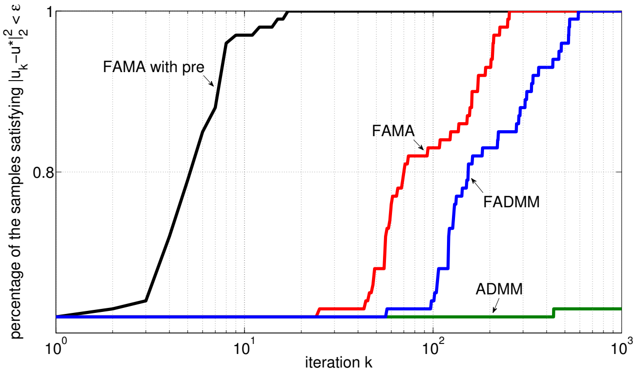
Fig. 2 illustrates the sequences $|\mathbf{u}^{k+1} - \mathbf{u}^\star|^2$ (solid lines) generated by Algorithm 1 with (black lines) and without (red lines) the pre-conditioning method and the corresponding complexity upper-bounds in (6) (dotted lines) for the initial state $\bar{x} = [-0.5, 10°, -10°, 60°, 10°/s, 10°/s, 30°/s]^T$. The complexity bounds are approximated by setting $\lambda^{start} = 0$ and assessing $|\lambda^*|$ by sampling. It can be clearly seen that the pre-conditioning method improves the convergence speed of the algorithm and reduces the complexity upper-bound. However, as the number of iteration k increases, the complexity upper-bound of Algorithm 1 with pre-conditioning appears to be less tight.

6. CONCLUSION

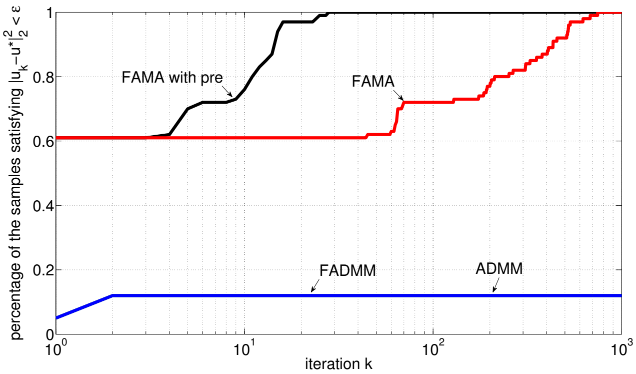
In this paper, the alternating direction method FAMA was proposed for solving MPC problems with polytopic and second-order cone constraints. An efficient splitting strategy simplifying of the computation at each iteration was presented. Upper-bounds on the number of iterations for a certain accuracy for both primal and dual variables were derived, which have enabled the proposition of a preconditioning method to improve the convergence speed.

REFERENCES

A. Alessio and A. Bemporad. A survey on explicit model predictive control. In L. Magni, D. M. Raimondo, and F. Allgower, editors, *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 345–369. Springer Berlin Heidelberg, 2009.



(a) $\delta = 10^{-4}$



(b) $\delta = 10^{-6}$

Figure 1. Performance of ADMM, FADMM, FAMA and FAMA with pre-conditioning for the quadroter example

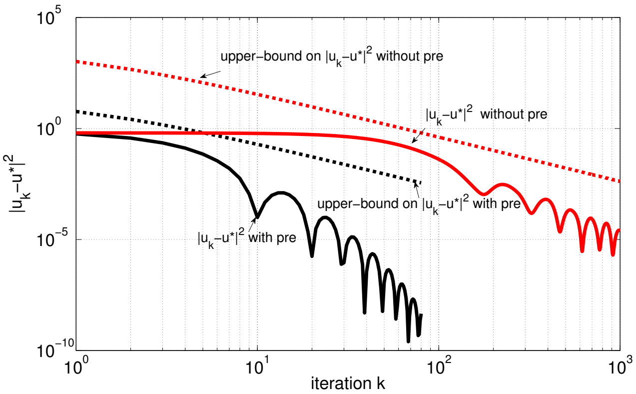


Figure 2. Illustration of $|\mathbf{u}^{k+1} - \mathbf{u}^\star|^2$ and the complexity upper-bounds in Theorem (3.6) for $x_0 = [-0.5, 10^\circ, -10^\circ, 60^\circ, 10^\circ/s, 10^\circ/s, 30^\circ/s]$.

R. A. Bartlett and L. T. Biegler. QPSchur: a dual, active-set, schur-complement method for large-scale and structured convex quadratic programming. *Optimization and Engineering*, 7:5–32, 2006.

A. Beck and M. Teboulle. A fast iterative shrinkage thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, pages 183–202, 2009.

D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar. *Convex analysis and optimization*. Athena Scientific Belmont, 2003.

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

S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, 1994.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3:1–122, 2011.

P. L. Combettes and J-C. Pesquet. Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, Springer Optimization and Its Applications, pages 185–212. Springer New York, 2011.

A. Domahidi, A. U. Zgraggen, M. N. Zeilinger, M. Morari, and C. N. Jones. Efficient interior point methods for multistage problems arising in receding horizon control. In *Proc. 51st IEEE Conf. on Decision Control*, pages 668–674, 2012.

H. I. Ferreau, H. G. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit mpc. *International Journal of Robust and Nonlinear Control*, 18(8):816–830, 2008.

P. Giselsson. Optimal preconditioning and iteration complexity bounds for gradient-based optimization in model predictive control. In *Proc. Amer. Control Conf.*, pages 358–364, 2013.

P. Giselsson, M. D. Doan, T. Keviczky, B. D. Schutter, and A. Rantzer. Accelerated gradient methods and dual decomposition in distributed model predictive control. *Automatica*, 49:829–833, 2013.

T. Goldstein, B. O'Donoghue, and S. Setzer. Fast alternating direction optimization methods. *UCLA Computational and Applied Mathematics Report*, pages 12–35, 2012.

M. Kögel and R. Findeisen. Fast predictive control of linear systems combining Nesterov's gradient method and the method of multipliers. In *Proc. 50th IEEE Conf. on Decision Control*, pages 501–506, 2011.

D. Mellinger and V. Kumar. Minimum snap trajectory generation and control for quadrotors. In *IEEE International Conference on Robotics and Automation*, pages 2520–2525, 2011.

Y. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27:372–376, 1983.

B. O'Donoghue, G. Stathopoulos, and S. Boyd. A splitting method for optimal control. *IEEE Trans. on Control Systems Technology*, 21:2432–2442, 2013.

R. Richter, C. N. Jones, and M. Morari. Computational complexity certification for real-time MPC with input constraints based on the fast gradient method. *IEEE Trans. Autom. Control*, 57(6):1391–1403, 2012.

P. Tseng. Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM Journal on Control and Optimization*, 29:119–138, 1991.

Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Trans. on Control Systems Technology*, 18:267–278, 2010.

M.N. Zeilinger, C.N. Jones, and M. Morari. Real-time suboptimal model predictive control using a combination of explicit MPC and online optimization. *IEEE Trans. on Autom. Control*, 56(7):1524–1534, 2011.