



**HAL**  
open science

# Suboptimality Based Early Stop in Dual Decomposition for Model Predictive Control

Xiang Dai, Romain Bourdais, Hervé Guéguen

► **To cite this version:**

Xiang Dai, Romain Bourdais, Hervé Guéguen. Suboptimality Based Early Stop in Dual Decomposition for Model Predictive Control. 2020. hal-03249141

**HAL Id: hal-03249141**

**<https://hal.science/hal-03249141>**

Preprint submitted on 3 Jun 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0  
International License

# $\epsilon$ Suboptimality Based Early Stop in Dual Decomposition for Model Predictive Control

Xiang Dai, Romain Bourdais, Herv Guguen

---

## Abstract

Dual decomposition is an efficient tool in dealing with Model Predictive Control (MPC) problems, particularly for distributed MPC. In this paper, we propose to limit the iteration number required in solving that problem, by stopping the iterative process once the solution is close enough to the optimal one. To do so, we introduce the concepts of primal and dual suboptimality, and derive respectively the projection and stopping condition for them. By exploiting the particular structure of the MPC problem where only the first step inputs are applied to the system, we devise an early  $\epsilon$  suboptimality stopping condition, focusing on components only at the first step of the prediction horizon, thus to further reduce the iteration number needed. Beyond the theoretical proofs that are developed, we illustrate the method's efficiency, both in computation time and iteration number, by various simulations.

*Key words:* Model predictive control; Suboptimality; Dual decomposition

---

## 1 Introduction

The linear time invariant system considered in this paper is composed of  $m$  subsystems and characterized by coupling dynamics and global linear equality constraints, which is governed by a MPC criterion. Specifically, the coupling dynamics are formulated as:

$$x_l(j) = \sum_{i=1}^m (A_{li}x_i(j-1) + B_{li}u_i(j-1)), \quad (1)$$

where for  $\forall l = 1, \dots, m$ ,  $j = 1, \dots, N$  ( $N \in \mathbb{N}_+$  is the prediction horizon),  $x_l(j) \in \mathbb{R}^{n_{x_l}}$  and  $u_l(j) \in \mathbb{R}^{n_{u_l}}$  are the states and inputs of  $l$ -th subsystem at step  $j$ ,  $A_{li} \in \mathbb{R}^{n_{x_l} \times n_{x_i}}$ ,  $B_{li} \in \mathbb{R}^{n_{x_l} \times n_{u_i}}$  are system matrix, and  $x_l(0) = \bar{x}_l$ ,  $\bar{x}_l \in \mathbb{R}^{n_{x_l}}$  is the initial state of  $l$ -th subsystem.

The global equality constraints of input and state vari-

ables are imposed to each subsystem as:

$$\sum_{l=1}^m A_j^l x_l(j) + \sum_{l=1}^m B_j^l u_l(j) = a_j, \quad (2)$$

where  $A_j^l \in \mathbb{R}^{n_{a_j} \times n_{x_l}}$ ,  $B_j^l \in \mathbb{R}^{n_{a_j} \times n_{u_l}}$  and  $a_j \in \mathbb{R}^{n_{a_j}}$ . Note that local equality constraints are indeed incorporated in (2).

It is well known that an iterative process in practice [3] [1] [10] typically solves the corresponding optimization problem. A major motivation of this paper is to reduce the iteration needed while achieving a solution close to the optimum.

Due to the attractive accessibility and scalability, the updated mainstream techniques in solving the above optimization problem are based on the first-order gradient or subgradient method [21], where the step size is a decisive parameter to determine. And vast research has been implemented in this field, such as: for fixed step size, e.g., exact first-order algorithm (EXTRA) [18], and distributed inexact gradient method and the gradient tracking technique (DIGing) [11]; for diminishing step size, e.g., distributed (sub)gradient descent (DGD) algorithm [12]. It is worth mentioning that a method originally proposed by Nesterov [14] has been proved by Theorem 2.2.2 [13] to be the optimal first-order gradient method for strongly convex optimization (with one time continuously differentiable objective function

---

\* The first author was supported by the program of China Scholarships Council (No.201701810044). The material in this paper was not presented at any conference. Corresponding author X. Dai.

The authors are with the Department of Automatic, CentraleSupélec, CNRS, IETR [(Institut d'Électronique et des Technologies du numéRique)] - UMR 6164, F-35000, Rennes, France. (email: {xiang.dai, romain.bourdais, herve.guguen}@centralesupelec.fr)

whose first derivative is Lipschitz continuous). The Nesterov method has been developed in [2] [14], and has been fitted to MPC in [7] named accelerated gradient method, which is adopted for iterative process in this paper.

It has been reviewed in [21] that the absolute optimal solution is strictly hard to obtain by iteration in complex constrained distributed optimization. Instead, a numerical tolerance is usually applied, enabling the iterative process to be terminated within finite iterations [19]. As can be expected, the equality constraints are also required to satisfy this numerical tolerance. Here, we introduce  $\epsilon$  ( $\epsilon > 0$ ), not as a numerical tolerance, but as a specified suboptimality tolerance, which permits the strict feasibility study of the optimization problem.

To generate  $\epsilon$  suboptimal solution, the updated dual objective value is used to approximate primal optimum generate in [8] [17], but the lasting approximation gap are presumed to cause over conservative solutions. Since the dual problem is exactly unconstrained optimization, we can apply the gradient-based suboptimality condition in [4], whose implementation is spared of approximation gap and requires no knowledge of the optimal solution.

Another key problem that arises after the iterative process termination is how to generate a feasible solution with suboptimality guarantee. One typical way is to employ an adaptable constraint tightening technique during the iteration [8] [22], yet heuristic adaption may cause extra iterations. Alternatively, an  $\epsilon$  accuracy feasibility, the largest violation of feasible constraints, is introduced in [17] as a stopping condition to obtain a "good enough" solution, which is still not strictly feasible. In this paper, we propose a projection mechanism to obtain a primal feasible solution with a suboptimality guarantee based on the gradient norm of dual function.

This paper is an extension of our previous work [6], where a heuristic for reducing the local problem size has been elaborated to diminish the complexity in distributed MPC generated optimization. Our main contribution in this paper is that, by formalizing this heuristic to MPC setting with coupling dynamics and general equality linear constraints, we provide the  $\epsilon$  suboptimality condition and related mathematical proof. Most significantly, we develop the first step focused  $\epsilon$  suboptimality condition and feasibility-ensured projection mechanism.

This paper is organized as follows. Section 2 sets up the optimization problem and fundamentals. Section 3 proposes gradient-based stopping condition and projection ensured feasibility and  $\epsilon$ -suboptimality. Section 4 demonstrates the first step focused stopping condition and the first step projection with proof of feasibility and  $\epsilon$ -suboptimality. Numerical experiments and results discussions are presented in Section 5. And conclusions are given in Section 6.

*Notation:* The norm  $\|\cdot\|$  denotes the Euclidean norm, for  $x \in \mathbb{R}^n$ ,  $n \in \mathbb{N}_+$  and  $R \in \mathbb{S}_+^n$ ,  $\|x\|_R = \sqrt{x^T R x}$ ,  $\min \text{eig}(R)$  denotes the minimal eigenvalue of  $R$ . For matrix  $A \in \mathbb{R}^{m_1 \times n}$  and  $B \in \mathbb{R}^{m_2 \times n}$  ( $m_1, m_2 \in \mathbb{N}_+$ ),  $A \oplus B = (A^T, B^T)^T$ , and  $\text{rank}(A)$  denotes the rank of  $A$ . By default, all the matrix, vector, scalar, projection operator, set and variable refer to that of full length prediction horizon  $N$  if no subscript appears. In contrast, for  $\forall a, b \in \mathbb{N}$  and  $1 \leq a \leq b \leq N$ , the subscript  $(\cdot)_{(a:b)}$  refers to its affiliated matrix, vector, scalar, projection operator, set and variable to the steps interval from step  $a$  to  $b$  in the prediction horizon  $N$ . The same notation essence also suits in the context where we specify "from step  $a$  to  $b$ ".

## 2 Problem statement and fundamentals

### 2.1 Problem statement

Let  $U_l = \bigoplus_{j=1}^N u_l(j-1)$  and  $X_l = \bigoplus_{j=1}^N x_l(j)$ , each subsystem is governed by a MPC criterion, resulting the local objective function as:

$$J_l(U_l, X_l) = \frac{1}{2} \sum_{j=1}^N (\|u_l(j-1)\|_{R_{ul}}^2 + \|x_l(j)\|_{R_{xl}}^2), \quad (3)$$

where  $R_{ul} \in \mathbb{S}_{++}^{nu_l}$  and  $R_{xl} \in \mathbb{S}_{++}^{nx_l}$ .

Accordingly, the local optimization problem for  $l$ -th subsystem is formulated as:

$$J_l^* = \min_{U_l, X_l} J_l(U_l, X_l), \quad (4)$$

*s.t.* (1), (2).

The global MPC criterion is defined as the sum of  $J_l(U_l, X_l)$  of all  $m$  subsystems. Hereby, we formulate the global optimization problem in a compact form as:

$$\mathcal{J}^* = \min_{\mathbf{y}} \mathcal{J}(\mathbf{y}), \quad (5a)$$

$$\textit{s.t. } \mathbf{A}\mathbf{y} = \mathbf{b}, \quad (5b)$$

where  $\mathcal{J}(\mathbf{y}) = \frac{1}{2} \|\mathbf{y}\|_{\mathbf{R}}^2 = \sum_{l=1}^m J_l(U_l, X_l)$ ,  $\mathbf{y} = \bigoplus_{j=1}^N \mathbf{y}(j:j)$ ,  $\mathbf{y}(j:j) = \mathbf{x}(j:j) \oplus \mathbf{u}(j-1:j-1)$ ,  $\mathbf{x}(j:j) = \bigoplus_{l=1}^m x_l(j)$ ,  $\mathbf{u}(j-1:j-1) = \bigoplus_{l=1}^m u_l(j-1)$ ,  $\mathbf{y} \in \mathbb{R}^{ny}$ , let  $n_e = N \sum_{l=1}^m nx_l + \sum_{j=1}^N na_j$ ,  $\mathbf{A} \in \mathbb{R}^{n_e \times ny}$ ,  $\mathbf{b} \in \mathbb{R}^{n_e}$  and  $\mathbf{R} \in \mathbb{S}_{++}^{ny}$  possesses the block diagonal structure.

**Assumption 1** We assume that the feasible set of problem (5) is not empty, and all constraints of (5b) are linear independent.

Correspondingly, the dual problem of problem (5) is defined as:

$$g^* = \max_{\boldsymbol{\theta}} g(\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \min_{\mathbf{y}} \mathcal{L}(\mathbf{y}, \boldsymbol{\theta}), \quad (6)$$

$$\mathcal{L}(\mathbf{y}, \boldsymbol{\theta}) = \frac{1}{2} \|\mathbf{y}\|_{\mathbf{R}}^2 + \boldsymbol{\theta}^T (\mathbf{A}\mathbf{y} - \mathbf{b}), \quad (7)$$

where  $\boldsymbol{\theta} \in \mathbb{R}^{n_e}$  is the dual variables associated with constraint (5b). Likewise the relation between problem (4) and (5),  $\mathcal{L}(\mathbf{y}, \boldsymbol{\theta})$  could be converted into the sum of local Lagrangian, which enables problem (6) to be solved distributively.

**Remark 1** *Since problem (6) is summation of its local counterpart, distributed or centralized manner would deliver the same solution at each iteration. We hence apply a centralized formulation hereafter to keep a concise notation, and to highlight the step-partitioned based methodology straightforwardly.*

**Remark 2** *Since problem (5) by nature is strongly convex, the Slater's condition is satisfied [4]. Then the strong duality holds by Slater's theorem, namely  $\mathcal{J}^* = g^*$ .*

## 2.2 $\epsilon$ -suboptimality definition

**Definition 1**  $\mathbf{y}$  is said to be an  $\epsilon$  primal solution of problem (5) if and only if:  $\mathbf{A}\mathbf{y} = \mathbf{b}$ , and  $\|\mathcal{J}(\mathbf{y}) - \mathcal{J}^*\| \leq \epsilon$ .

As we intend to solve the dual problem (6) prior to get a primal solution, correspondingly the  $\epsilon$  suboptimal solution from the dual point of view is defined as follows.

**Definition 2**  $(\mathbf{y}_{\boldsymbol{\theta}}, \boldsymbol{\theta})$  is said to be an  $\epsilon$  dual solution of problem (6) if and only if:  $\mathbf{y}_{\boldsymbol{\theta}} = \arg \min_{\mathbf{y}} \mathcal{L}(\mathbf{y}, \boldsymbol{\theta})$  and  $\|\mathcal{L}(\mathbf{y}_{\boldsymbol{\theta}}, \boldsymbol{\theta}) - \mathcal{J}^*\| \leq \epsilon$ .

Note that, an  $\epsilon$  suboptimal solution  $(\mathbf{y}_{\boldsymbol{\theta}}, \boldsymbol{\theta})$  of problem (6) cannot guarantee that  $\mathbf{y}_{\boldsymbol{\theta}}$  is primal feasible, which consequently demands a further feasibility verification with respect to problem (5).

## 2.3 Accelerated gradient method

To solve dual decomposition based convex optimization problem (6) iteratively, we recall the accelerated gradient method in [7] as:

$$\boldsymbol{\theta}^{k+1} = \tilde{\boldsymbol{\theta}}^k + \frac{1}{L} (\mathbf{A}\tilde{\mathbf{y}}^k - \mathbf{b}), \quad (8)$$

$$\mathbf{y}^{k+1} = -\mathbf{R}^{-1} \mathbf{A}^T \boldsymbol{\theta}^{k+1}, \quad (9)$$

where  $\tilde{\boldsymbol{\theta}}^k = \boldsymbol{\theta}^k + \frac{k-1}{k+2} (\boldsymbol{\theta}^k - \boldsymbol{\theta}^{k-1})$ ,  $\tilde{\mathbf{y}}^k = \mathbf{y}^k + \frac{k-1}{k+2} (\mathbf{y}^k - \mathbf{y}^{k-1})$ , and  $L = \|\mathbf{A}\mathbf{R}^{-1} \mathbf{A}^T\|$ .

## 3 The gradient based stopping condition and projection

In this section, we first study the stopping condition to guarantee a predefined suboptimality  $\epsilon$  in solving problem (6). Then we propose a linear projection that produces a primal  $\epsilon$ -suboptimal solution.

### 3.1 Stopping condition of $\epsilon$ suboptimal solution

As formulated in (7), the Lagrangian is a continuous quadratic function with positive definite hessian, thus differentiable. Given  $\boldsymbol{\theta} \in \mathbb{R}^{n_e}$ , using the first order necessary optimality condition  $\nabla_{\mathbf{y}} \mathcal{L}(\mathbf{y}, \boldsymbol{\theta}) = \mathbf{0}$ , we have:

$$\mathbf{y}_{\boldsymbol{\theta}} = -\mathbf{R}^{-1} \mathbf{A}^T \boldsymbol{\theta}, \quad (10)$$

Substituting (10) into (7) yields:

$$g(\boldsymbol{\theta}) = -\frac{1}{2} \boldsymbol{\theta}^T \mathbf{A}\mathbf{R}^{-1} \mathbf{A}^T \boldsymbol{\theta} - \boldsymbol{\theta}^T \mathbf{b}. \quad (11)$$

Based on the explicit expression of  $g(\boldsymbol{\theta})$  in (11), we have its first and second order gradient respectively as:

$$\nabla g(\boldsymbol{\theta}) = \mathbf{A}\mathbf{y}_{\boldsymbol{\theta}} - \mathbf{b}, \quad (12)$$

$$\nabla^2 g(\boldsymbol{\theta}) = -\mathbf{A}\mathbf{R}^{-1} \mathbf{A}^T. \quad (13)$$

Let us define  $\beta = \min \text{eig}(\mathbf{A}\mathbf{R}^{-1} \mathbf{A}^T)$ , and we have  $\beta > 0$  since  $\mathbf{R} \in \mathbb{S}_{++}^{n_y}$ .

**Lemma 1** *In implementing iterative process (8) (9), if*

$$\|\mathbf{A}\mathbf{y}^k - \mathbf{b}\|^2 \leq 2\beta\epsilon \quad (14)$$

*is satisfied, then  $(\mathbf{y}^k, \boldsymbol{\theta}^k)$  is an  $\epsilon$  dual solution of problem (6).*

**PROOF.** First, we introduce a convex optimization problem as:

$$F^* = \min_{\boldsymbol{\theta}} F(\boldsymbol{\theta}), \quad (15)$$

where  $F(\boldsymbol{\theta}) = -g(\boldsymbol{\theta})$  and  $F^* = -g^*$ .

Subsequently, we have :

$$\|\nabla g(\boldsymbol{\theta})\|^2 = \|\nabla F(\boldsymbol{\theta})\|^2 = \|\mathbf{A}\mathbf{y}_{\boldsymbol{\theta}} - \mathbf{b}\|^2, \quad (16)$$

$$\|F(\boldsymbol{\theta}) - F^*\| = \|g^* - g(\boldsymbol{\theta})\|. \quad (17)$$

Viewing (9) and (10), combined with (12), we actually have at each iteration:

$$\nabla g(\boldsymbol{\theta}^k) = \mathbf{A}\mathbf{y}^k - \mathbf{b}, \quad (18)$$

Recall 9.1.2 of [4], for problem (15) to attain  $\epsilon$ -suboptimality such that  $F(\boldsymbol{\theta}) - F^* \leq \epsilon$ , the sufficient condition is:

$$\|\nabla F(\boldsymbol{\theta})\| \leq (2\beta\epsilon)^{1/2}. \quad (19)$$

Using (16) (17) and (19), we can conclude the proof. ■

### 3.2 From dual $\epsilon$ suboptimal solution to primal $\epsilon$ suboptimal solution

If  $(\mathbf{y}_\theta, \boldsymbol{\theta})$  is an  $\epsilon$  suboptimal solution of problem (6), we do not necessarily have  $\mathbf{y} \in Y = \{\mathbf{y} \mid \mathbf{A}\mathbf{y} = \mathbf{b}\}$ , then an specific projection from  $\mathbf{y}$  onto feasible set of problem (5) is needed. We first propose a matrix  $\mathbf{F} \in \mathbb{R}^{ny \times (ny - n_e)}$  satisfying that  $\mathbf{A}\mathbf{F} = \mathbf{0}$ , and

$$\min \text{eig}(\mathbf{F}^T \mathbf{R} \mathbf{F}) = \beta. \quad (20)$$

Particularly, (20) could be fulfilled by the following treatment: given any  $\mathbf{F}' \in \mathbb{R}^{ny \times (ny - n_e)}$  with  $\mathbf{A}\mathbf{F}' = \mathbf{0}$  and  $\min \text{eig}(\mathbf{F}'^T \mathbf{R} \mathbf{F}') \neq \beta$ , let  $b = \frac{\beta}{\min \text{eig}(\mathbf{F}'^T \mathbf{R} \mathbf{F}')}$ , then  $\mathbf{F} = \sqrt{b}\mathbf{F}'$ . Because we have:

$$\begin{aligned} \min \text{eig}(\mathbf{F}^T \mathbf{R} \mathbf{F}) &= \min \text{eig}(b\mathbf{F}'^T \mathbf{R} \mathbf{F}') \\ &= b(\min \text{eig}(\mathbf{F}'^T \mathbf{R} \mathbf{F}')) = \beta. \end{aligned}$$

Then, we can formulate the feasible set  $Y$  as:

$$Y = \{\mathbf{y} \mid \mathbf{A}\mathbf{y} = \mathbf{b}\} = \{\hat{\mathbf{y}} + \mathbf{F}\mathbf{t} \mid \mathbf{t} \in \mathbb{R}^{ny - n_e}\}, \quad (21)$$

this characterization is based on any  $\hat{\mathbf{y}} \in Y$ .

Next, inspired by linear projection operator  $P_1$  in [5], we propose  $P_e$  from  $\mathbf{y} \in \mathbb{R}^{ny}$  onto  $Y$  as  $P_e(\mathbf{y}): \mathbf{y} \mapsto \mathbf{y}_e$ .

$$\mathbf{y}_e = \mathbf{y} - \mathbf{A}_p^T (\mathbf{A}_p \mathbf{A}_p^T)^{-1} (\mathbf{A}_p \mathbf{y} - \mathbf{b}_p), \quad (22)$$

where  $\mathbf{A}_p = \mathbf{F}^T \mathbf{R} \oplus \mathbf{A}$ ,  $\mathbf{b}_p = \mathbf{h} \oplus \mathbf{b}$ , let  $p = ny - n_e$ ,  $\mathbf{h} \in \mathbb{R}^{ny - n_e}$  is any vector satisfying

$$\|\mathbf{h}\| \leq \|\nabla g(\boldsymbol{\theta})\|. \quad (23)$$

**Lemma 2** *If  $(\mathbf{y}_\theta, \boldsymbol{\theta})$  is an  $\epsilon$  suboptimal dual solution of problem (6), then  $\mathbf{y}_e = P_e(\mathbf{y})$  is an  $\epsilon$  primal solution of problem (5).*

**PROOF.** First, we prove that  $\mathbf{y}_e$  is a feasible solution of problem (5). Left multiplying the right side of (22) by  $\mathbf{A}_p$ , we have  $\mathbf{A}_p \mathbf{y}_e = \mathbf{b}_p$ , which could be partitioned as:

$$\mathbf{A}\mathbf{y}_e = \mathbf{b}, \quad (24)$$

$$\mathbf{F}^T \mathbf{R} \mathbf{y}_e = \mathbf{h}. \quad (25)$$

By (24), we can conclude that  $\mathbf{y}_e \in Y_e$  is a primal feasible solution of problem (5).

Next, we prove the  $\epsilon$  suboptimality of  $\mathbf{y}_e$ . By (21), Problem (5) is equivalent as:

$$\mathbf{J}^* = \min_{\mathbf{t}} \mathbf{J}(\mathbf{t}), \quad (26)$$

where  $\mathbf{J}(\mathbf{t}) = \frac{1}{2} \|\hat{\mathbf{y}} + \mathbf{F}\mathbf{t}\|_{\mathbf{R}}^2$ , and  $\mathbf{J}^* = \tilde{\mathcal{J}}^*$ .

Accordingly, we have

$$\nabla \mathbf{J}(\mathbf{t}) = \mathbf{F}^T \mathbf{R} (\hat{\mathbf{y}} + \mathbf{F}\mathbf{t}),$$

$$\nabla^2 \mathbf{J}(\mathbf{t}) = \mathbf{F}^T \mathbf{R} \mathbf{F}.$$

By making  $\mathbf{y}_e = \hat{\mathbf{y}} + \mathbf{F}\mathbf{t}_e$ , we have  $\nabla \mathbf{J}(\mathbf{t}_e) = \mathbf{F}^T \mathbf{R} \mathbf{y}_e$ .

Next, by (23) we have

$$\|\nabla \mathbf{J}(\mathbf{t}_e)\|^2 = \|\mathbf{h}\|^2 \leq \|\nabla g(\boldsymbol{\theta})\|^2 = \|\mathbf{A}\mathbf{y} - \mathbf{b}\|^2. \quad (27)$$

As  $\mathbf{J}(\mathbf{t}_e) = \mathcal{J}(\mathbf{y}_e)$ , then by (14) and (20), we have  $\|\mathcal{J}(\mathbf{y}_e) - \tilde{\mathcal{J}}^*\| \leq \epsilon$ . And this completes the proof. ■

Integrating  $\epsilon$  suboptimality stopping condition (14) and projection operator  $P_e$ , Algorithm 1 presents the procedures to generate  $\epsilon$  primal solution for problem (5).

---

**Algorithm 1** Full Prediction Horizon Stopping Condition With Projection (FPH-P)

---

- 1: Initialize:  $\boldsymbol{\theta}^0 = \boldsymbol{\theta}^{-1}$ ,  $\epsilon$ ,  $\beta$  and  $k = 0$ .  $\mathbf{y}^{-1}$  and  $\mathbf{y}^0$  are given by (9).
  - 2: **while** (14) is not satisfied **do**
  - 3:     Update primal and dual variables by (9) and (8) respectively
  - 4:      $k \leftarrow k + 1$
  - 5: **end while**
  - 6:  $\mathbf{y}_e^k = P_e(\mathbf{y}^k)$
- 

## 4 The first step focused stopping condition and projection

In the context of MPC, where only inputs of the first step in the current prediction horizon would be applied, two questions in turn arise. Can we transform (14) to stopping condition focused only on the first step? Can we find a way to generate a feasible solution only using the first step elements while guarantee the  $\epsilon$  suboptimality? These two questions are consecutively addressed in this section.

#### 4.1 The first step focused stopping condition

Note that in the MPC context, the step separated structure existing in problem (5) for both objective function and constraints enables a step-based partition, which will serve as the basis of methods studied in this section.

Here, we propose to convert the stopping condition (14) into the first step oriented, which consists of 2 steps. The first step is to partition the prediction horizon into two parts: the first step, and from step 2 to  $N$ .

As a prerequisite, we decompose  $\mathbf{A}$  and  $\mathbf{b}$  into block form as:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{(1:1)} & \mathbf{0} \\ \mathbf{B}_{(2:N)} & \mathbf{A}_{(2:N)} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_{(1:1)} \\ \mathbf{b}_{(2:N)} \end{bmatrix}. \quad (28)$$

Based on (28), the gradient  $\nabla g(\boldsymbol{\theta})$  and iteration of  $\mathbf{y}^k$  are partitioned as:

$$\mathbf{y}_{(1:1)}^k = -\mathbf{R}_{(1:1)}^{-1}(\mathbf{A}_{(1:1)}^T \boldsymbol{\theta}_{(1:1)}^k + \mathbf{B}_{(2:N)}^T \boldsymbol{\theta}_{(2:N)}^k), \quad (29)$$

$$\mathbf{y}_{(2:N)}^k = -\mathbf{R}_{(2:N)}^{-1} \mathbf{A}_{(2:N)}^T \boldsymbol{\theta}_{(2:N)}^k, \quad (30)$$

$$\nabla g_{(1:1)}^k = \mathbf{A}_{(1:1)} \mathbf{y}_{(1:1)}^k - \mathbf{b}_{(1:1)}, \quad (31)$$

$$\nabla g_{(2:N)}^k = \mathbf{B}_{(2:N)} \mathbf{y}_{(1:1)}^k + \mathbf{A}_{(2:N)} \mathbf{y}_{(2:N)}^k - \mathbf{b}_{(2:N)}, \quad (32)$$

where the variables in step partitioned gradient expression are omitted hereafter to lighten the notation.

The second step is to reconstruct (14): setting the required gradient norm from step 2 to  $N$  as 0. By doing so, we create a less demanding stopping condition for the first step with  $\epsilon$  suboptimality guarantee, which enables a possible earlier termination of iterative process (8)-(9). As such, the early stopping condition is proposed as:

$$\|\nabla g_{(1:1)}^k\|^2 \leq 2\beta\epsilon. \quad (33)$$

**Theorem 1** Let  $\bar{\mathbf{y}}_{(2:N)}$  and  $\bar{\boldsymbol{\theta}}$  be solved by the following equations:

$$\mathbf{B}_{(2:N)} \mathbf{y}_{(1:1)}^k + \mathbf{A}_{(2:N)} \bar{\mathbf{y}}_{(2:N)} - \mathbf{b}_{(2:N)} = \mathbf{0}, \quad (34)$$

$$\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)} = -\mathbf{R}^{-1} \mathbf{A}^T \bar{\boldsymbol{\theta}}. \quad (35)$$

Then, if (33) is satisfied in implementing iterative process (8) (9),  $(\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)}, \bar{\boldsymbol{\theta}})$  is an  $\epsilon$  dual solution of problem (6).

**PROOF.** By Assumption 1, all constraints of (5b) are linear independent, thus  $\mathbf{A}$  is full row rank. Next, by partition (28),  $\mathbf{A}_{(2:N)}$  is correspondingly full row rank.

As such, by (34) (35),  $\bar{\mathbf{y}}_{(2:N)}$  and  $\bar{\boldsymbol{\theta}}$  could be solved explicitly using Pseudo-inverse as:

$$\begin{aligned} \bar{\mathbf{y}}_{(2:N)} &= \mathbf{A}_{(2:N)}^T (\mathbf{A}_{(2:N)} \mathbf{A}_{(2:N)}^T)^{-1} (\mathbf{b}_{(2:N)} - \mathbf{B}_{(2:N)} \mathbf{y}_{(1:1)}^k), \\ \bar{\boldsymbol{\theta}} &= -(\mathbf{A} \mathbf{A}^T)^{-1} \mathbf{A} \mathbf{R} (\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)}). \end{aligned}$$

Since  $\bar{\mathbf{y}}_{(2:N)}$  and  $\bar{\boldsymbol{\theta}}$  are solvable, by (10) and (35), we have

$$\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)} = \arg \min_{\mathbf{y}} \mathcal{L}(\mathbf{y}, \bar{\boldsymbol{\theta}}). \quad (36)$$

which means  $(\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)}, \bar{\boldsymbol{\theta}})$  is a dual solution of problem (6).

It remains to prove the following inequality.

$$g^* - \mathcal{L}(\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)}, \bar{\boldsymbol{\theta}}) \leq \epsilon.$$

By Lemma 1, we need to have:

$$\|\nabla g(\bar{\boldsymbol{\theta}})\|^2 \leq 2\beta\epsilon.$$

By (31) (32) and (35), we have

$$\begin{aligned} \|\nabla g(\bar{\boldsymbol{\theta}})\|^2 &= \|\nabla g_{(1:1)}^k\|^2 + \\ &\|\mathbf{B}_{(2:N)} \mathbf{y}_{(1:1)}^k + \mathbf{A}_{(2:N)} \bar{\mathbf{y}}_{(2:N)} - \mathbf{b}_{(2:N)}\|^2. \end{aligned}$$

We can conclude the proof by (34) and (33). ■

**Lemma 3** In implementing iterative process (8)-(9), let  $k_1$  and  $k_2$  be defined as:

$$k_1 = \inf\{k \mid \|\nabla g_{(1:1)}^k\|^2 \leq 2\beta\epsilon\}, \quad (37)$$

$$k_2 = \inf\{k \mid \|\nabla g^k\|^2 \leq 2\beta\epsilon\}, \quad (38)$$

then  $k_1 \leq k_2$ .

**PROOF.** Since for  $\forall k$ ,  $\|\nabla g^k\|^2 = \|\nabla g_{(1:1)}^k\|^2 + \|\nabla g_{(2:N)}^k\|^2$ , and  $\|\nabla g_{(2:N)}^k\|^2 \geq 0$ , we have

$$\|\nabla g_{(1:1)}^{k_2}\|^2 \leq \|\nabla g^{k_2}\|^2 \leq 2\beta\epsilon.$$

As a consequence, by (37) we have  $k_1 \leq k_2$ . And this completes the proof. ■

#### 4.2 The first step focused projection

In this subsection, we specify the projection operator  $P_{e,(1:1)}$  from  $\mathbf{y}_{(1:1)} \in \mathbb{R}^{ny(1:1)}$  onto  $Y_{(1:1)}$  as

$P_{e,(1:1)}(\mathbf{y}_{(1:1)}): \mathbf{y}_{(1:1)} \mapsto \mathbf{y}_{e,(1:1)}$ .

$$\begin{aligned} \mathbf{y}_{e,(1:1)} &= \mathbf{y}_{(1:1)} - \\ \mathbf{A}_{p,(1:1)}^T (\mathbf{A}_{p,(1:1)} \mathbf{A}_{p,(1:1)}^T)^{-1} (\mathbf{A}_{p,(1:1)} \mathbf{y}_{(1:1)} - \mathbf{b}_{p,(1:1)}), \end{aligned} \quad (39)$$

where  $\mathbf{A}_{p,(1:1)} = \mathbf{F}_{(1:1)}^T \mathbf{R}_{(1:1)} \oplus \mathbf{A}_{(1:1)}$ ,  $\mathbf{b}_{p,(1:1)} = \mathbf{h}_{(1:1)} \oplus \mathbf{b}_{(1:1)}$ ,  $\nabla g_{(1:1)}(\boldsymbol{\theta}_{(1:1)}) = \mathbf{A}_{(1:1)} \mathbf{y}_{(1:1)} - \mathbf{b}_{(1:1)}$ , let  $p_{(1:1)} = ny_{(1:1)} - n_{e,(1:1)}$  and  $\mathbf{h}_{(1:1)} \in \mathbb{R}^{p_{(1:1)}}$  is any vector satisfying  $\|\mathbf{h}_{(1:1)}\| \leq \|\nabla g_{(1:1)}(\boldsymbol{\theta}_{(1:1)})\|$ .

**Lemma 4** *If  $\mathbf{y}_{(1:1)}^k$  satisfies (33), then  $\mathbf{y}_{e,(1:1)}^k = P_{e,(1:1)}(\mathbf{y}_{(1:1)}^k)$  is the first step components of an  $\epsilon$  primal solution of problem (5).*

**PROOF.** By Theorem 1,  $(\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)}, \bar{\boldsymbol{\theta}})$  is an  $\epsilon$  dual solution of problem (6). Implementing projection  $P_e$ , we have by Lemma 2 an  $\epsilon$  primal solution of problem (5) as  $\bar{\mathbf{y}}_e = P_e(\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)})$ , which could be partitioned as  $\bar{\mathbf{y}}_e = \bar{\mathbf{y}}_{e,(1:1)} \oplus \bar{\mathbf{y}}_{e,(2:N)}$ .

By (24) (28), for feasibility fulfillment we have:

$$\mathbf{A}_{(1:1)} \bar{\mathbf{y}}_{e,(1:1)} = \mathbf{b}_{(1:1)}, \quad (40)$$

$$\mathbf{A}_{(2:N)} \bar{\mathbf{y}}_{e,(2:N)} + \mathbf{B}_{(2:N)} \bar{\mathbf{y}}_{e,(1:1)} = \mathbf{b}_{(2:N)}. \quad (41)$$

By (27) (28), for  $\epsilon$  suboptimality fulfillment we have:

$$\begin{aligned} \|\mathbf{F}_{(1:1)}^T \mathbf{R}_{(1:1)} \bar{\mathbf{y}}_{e,(1:1)}\|^2 &\leq \|g_{(1:1)}^k\|^2 = \\ \|\mathbf{A}_{(1:1)} \mathbf{y}_{(1:1)}^k - \mathbf{b}_{(1:1)}\|^2, \end{aligned} \quad (42)$$

$$\|\mathbf{F}_{(2:N)}^T \mathbf{R}_{(2:N)} \bar{\mathbf{y}}_{e,(2:N)}\|^2 = 0. \quad (43)$$

Let  $\mathbf{y}_{e,(1:1)}^k = P_{e,(1:1)}(\mathbf{y}_{(1:1)}^k)$ , we have  $\mathbf{y}_{e,(1:1)}^k$  satisfying (40) and (42). Then substituting  $\bar{\mathbf{y}}_{e,(1:1)}$  by  $\mathbf{y}_{e,(1:1)}^k$  in (41), we can form a linear equation group (with  $\mathbf{y}_{e,(2:N)}^k$  being variable to be solved) as:

$$\begin{cases} \mathbf{A}_{(2:N)} \mathbf{y}_{e,(2:N)}^k + \mathbf{B}_{(2:N)} \mathbf{y}_{e,(1:1)}^k = \mathbf{b}_{(2:N)}, \\ \mathbf{F}_{(2:N)}^T \mathbf{R}_{(2:N)} \mathbf{y}_{e,(2:N)}^k = \mathbf{0}. \end{cases} \quad (44)$$

As  $\mathbf{y}_{e,(2:N)}^k \in \mathbb{R}^{ny_{(2:N)}}$ ,  $\mathbf{A}_{(2:N)} \in \mathbb{R}^{n_{e,(2:N)} \times ny_{(2:N)}}$ ,  $\mathbf{F}_{(2:N)} \in \mathbb{R}^{ny_{(2:N)} \times p_{(2:N)}}$  and  $\mathbf{R}_{(2:N)} \in \mathbb{S}_{++}^{ny_{(2:N)}}$ , by Assumption 1 we have  $\text{rank}(\mathbf{A}_{(2:N)}) = n_{e,(2:N)}$ ,  $\text{rank}(\mathbf{F}_{(2:N)}^T \mathbf{R}_{(2:N)}) = p_{(2:N)}$ , thus (44) has the unique solution of  $\mathbf{y}_{e,(2:N)}^k$ .

Consequently,  $\mathbf{y}_{e,(1:1)}^k \oplus \mathbf{y}_{e,(2:N)}^k$  is indeed an  $\epsilon$  primal solution of problem (5) by Definition 1, of which  $\mathbf{y}_{e,(1:1)}^k$  is the first step components. And this completes the proof. ■

Algorithm 2 depicts the mechanism combining the first step focused stopping condition and the first step projection.

---

**Algorithm 2** First Step Stopping Condition With Projection (FS-P)

---

- 1: Initialize:  $\boldsymbol{\theta}^0 = \boldsymbol{\theta}^{-1}$ ,  $\epsilon$ ,  $\beta$  and  $k = 0$ .  $\mathbf{y}^{-1}$  and  $\mathbf{y}^0$  are given by (9).
  - 2: **while** (33) is not satisfied **do**
  - 3:     Update primal and dual variables by (9) and (8) respectively
  - 4:      $k \leftarrow k + 1$
  - 5: **end while**
  - 6:  $\mathbf{y}_{e,(1:1)}^k = P_{e,(1:1)}(\mathbf{y}_{(1:1)}^k)$
- 

Note that in this section, the presence of  $\bar{\mathbf{y}}_{(2:N)}$ ,  $\bar{\boldsymbol{\theta}}$  and  $\mathbf{y}_{e,(2:N)}^k$  are purely to complete the mathematical proof, we only need to obtain  $\mathbf{y}_{(1:1)}^k$  and  $\mathbf{y}_{e,(1:1)}$  in implementing FS-P.

## 5 Numerical experiments

Table 1

Suboptimality performance comparison (with  $\mathcal{J}^*$  as the benchmark) among FS, FS-P, FPH and FPH-P. (The magnitude of  $1 \times 10^{-3}$ ,  $1 \times 10^{-4}$  and  $1 \times 10^{-5}$  are omitted from the results according to the relative suboptimality referred for space-saving. As a result, any result showed with absolute value less than 1 means predefined suboptimality is guaranteed.)

Rel. $\epsilon$	Alg.	Ave. Rel. Error					Max. Rel. Error				
		Prediction Horizon					Prediction Horizon				
		10	20	30	40	50	10	20	30	40	50
$10^{-3}$	FPH	-0.58	-0.48	-0.47	-0.44	-0.46	-0.94	-0.91	-0.92	-0.89	0.95
	FS	-0.48	-0.56	-0.58	-0.67	-0.70	-0.91	-0.95	-0.96	-0.99	-0.98
	FPH-P	0.89	0.87	0.86	0.83	0.85	0.99	0.99	0.99	0.99	0.99
	FS-P	0.83	0.85	0.84	0.86	0.85	0.99	0.99	0.99	0.99	0.99
$10^{-4}$	FPH	-0.54	-0.51	-0.51	-0.50	-0.50	-0.91	-0.93	-0.93	-0.97	-0.92
	FS	-0.55	-0.50	-0.50	-0.50	-0.47	-0.98	-0.97	-0.94	-0.98	-0.99
	FPH-P	0.88	0.87	0.88	0.86	0.85	0.99	0.99	0.99	0.99	0.99
	FS-P	0.85	0.83	0.83	0.86	0.85	0.99	0.99	0.99	0.99	0.99
$10^{-5}$	FPH	-0.57	-0.56	-0.54	-0.51	-0.50	-0.92	-0.94	-0.96	-0.94	-0.92
	FS	-0.53	-0.48	-0.45	-0.46	-0.50	-0.92	-0.96	-0.94	-0.98	-0.91
	FPH-P	0.89	0.89	0.87	0.84	0.85	0.99	0.99	0.99	0.99	0.99
	FS-P	0.81	0.81	0.80	0.81	0.81	0.99	0.99	0.99	0.99	0.99

In this section, FPH-P and FS-P are tested under 5 prediction horizons, from 10 to 50 with the incremental interval of 10. Of each prediction horizon, 100 independent randomly generated numerical experiments are carried out using MATLAB 2018b on a Windows 10 PC with 2.20 GHz Core i7-8750H CPU and 16GB RAM.

The tested system, with  $A_{li}$  and  $B_{li}$  randomly generated by MATLAB command `syss`, consists of 5 subsystems, each of which contains 2 inputs and 2 outputs. More in

details, each element of  $\mathbf{x}_0$  is randomly drawn from uniform distribution  $[-0.5, 0.5]$ . The equality constraints are  $\sum_{i=1}^m u_i(j) = 1$ . The penalty matrix  $\mathbf{R} = \mathbf{I}$ , and the relative suboptimality<sup>1</sup> tested are  $1 \times 10^{-3}$ ,  $1 \times 10^{-4}$  and  $1 \times 10^{-5}$ .

Particularly, for projection treatment, we simply make the first element of  $\mathbf{h}$  equal to  $\|\nabla g(\boldsymbol{\theta}^k)\|$  and rest elements equal 0 in Step 6 of FPH-P to get  $\mathbf{y}_e^k$ ; and make the first element of  $\mathbf{h}_{(1:1)}$  equal to  $\|\nabla g_{(1:1)}^k\|$  and rest elements equal 0 in Step 6 of FS-P to get  $\mathbf{y}_{e,(1:1)}^k$ . Subsequently,  $\mathbf{y}_{e,(2:N)}^k$  is solved by (44) using  $\mathbf{y}_{e,(1:1)}^k$ .

In Table 1, FPH refers to  $g(\boldsymbol{\theta}^k)$  when (14) in FPH-P is satisfied. Specifically, FS refers to the full length prediction objective value  $g(\bar{\boldsymbol{\theta}})$ , and  $\bar{\boldsymbol{\theta}}$  is solved by (34) (35) when (33) is satisfied. The benchmark value  $\mathcal{J}^*$  for relative error comparison is solved by commercial optimization solver MOSEK programmed in platform YALMIP [9].

Table 1 shows that the predefined suboptimality of all  $N$  cases is guaranteed by implementing FS, FPH, and their projections. The discrepancies of both average and maximal relative error between FS and FPH are comparably small, so as that between FS-P and FPH-P, which suggests that using the first step stopping condition does not impact fulfillment of feasibility and  $\epsilon$  suboptimality.

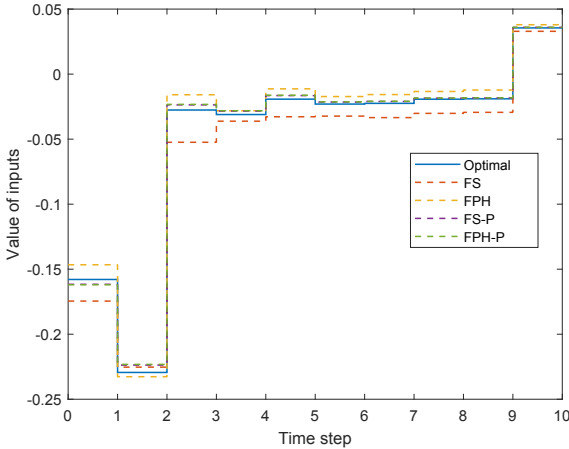


Fig. 1. Input sequence comparison of one subsystem in a test among optimal solution, FS, FPH, FS-P and FPH-P with  $N = 10$  and relative suboptimality being  $1 \times 10^{-3}$ . ( $\mathbf{y}_{(1:1)}^k \oplus \bar{\mathbf{y}}_{(2:N)}$  and  $\mathbf{y}_{e,(1:1)}^k \oplus \mathbf{y}_{e,(2:N)}^k$  are presented respectively for FS and FS-P, note that only inputs of the first time step would be applied.)

<sup>1</sup> The relative suboptimality is computed as suboptimality divided by  $\mathcal{J}^*$ .

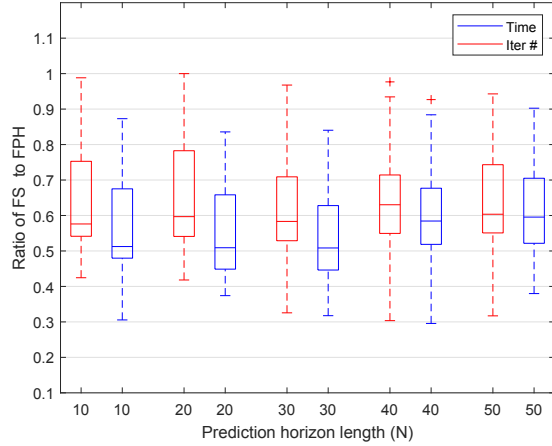


Fig. 2. Iteration number and computation time ratio of FS to FPH with relative suboptimality being  $1 \times 10^{-3}$  (sample value exceeded  $\pm 2.7\sigma$  shows as whisker, same setting for other box plots)

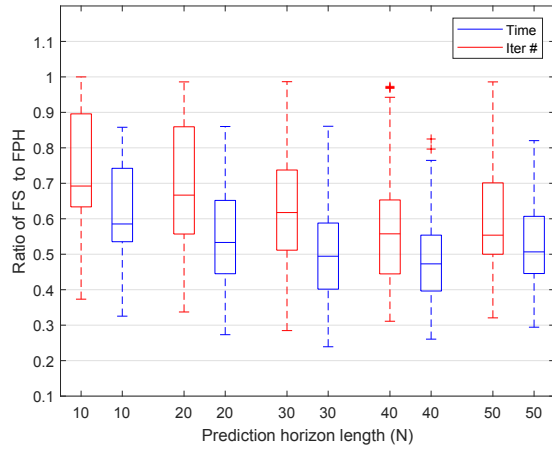


Fig. 3. Iteration number and computation time ratio of FS to FPH with relative suboptimality being  $1 \times 10^{-4}$

In Fig. 1, by projection mechanism, the trajectory of FS-P and FPH-P are closer to the optimal solution than FS and FPH. Furthermore, the latter the step appeared in input sequence of FS-P and FPH-P, the tighter the gap between them and the optimal solution.

From Fig. 2, Fig. 3 and Fig. 4, statistically, FS consumes significantly fewer iterations compared to FPH in majority tests of all  $N$  cases and of all relative suboptimality. Note that, ratio as 1 denotes that  $k_1 = k_2$  in Lemma 3, which is the worst case that could happen to FS in terms of iteration number. Generally, the ratio of FS to FPH in computation time are even smaller than that in iteration number. Observing (33) and (14), at each iteration FS only needs to calculate gradient for the first step, while the full prediction horizon calculation is required of FPH, resulting in an even larger time advantage as  $N$



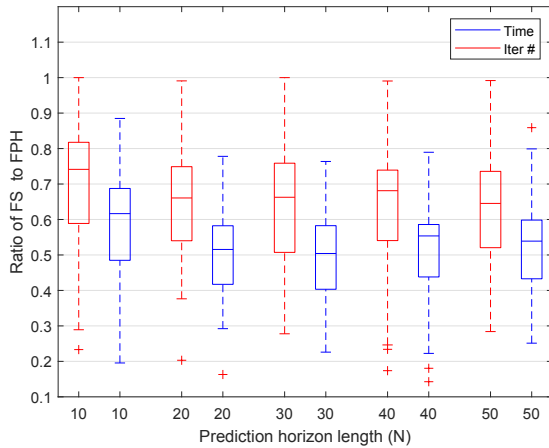


Fig. 4. Iteration number and computation time ratio of FS to FPH with relative suboptimality being  $1 \times 10^{-5}$

increases.

## 6 Conclusion

In this paper, in tackling MPC resulted optimization, we have proposed projection onto the primal feasible set with  $\epsilon$ -suboptimality guarantee based on a  $\epsilon$  dual solution. We have demonstrated the early stopping condition in the MPC context through a step-based partition technique by focusing on the first step components. To generate the system's applicable inputs, we have also introduced the first step focused projection with  $\epsilon$ -suboptimality and feasibility guarantee.

Through random numerical experiments, the  $\epsilon$ -suboptimality condition has been verified for both algorithms. Due to less demanding stopping condition and computation burden, the first step focused algorithm has generally outperformed the full prediction horizon algorithm largely in iteration number and computation time.

The technique demonstrated in this paper can be extended to MPC setting with linear inequality constraints, which could be either eliminated by barrier function [4], or transformed into equality constraints by active constraint identification technique [15] [16] or active set method [20]. Hessian approximation/factorization and eigenvalue approximation could also be adopted in solving  $L$  and  $\beta$  in large scale system or in cases containing sparse structure. Input sequence showed in Fig. 1 endows the potential for stability study.

## References

[1] A. Alessio and A. Bemporad. A survey on explicit model predictive control. In *Nonlinear Model Predictive Control*, pages 345–369. Springer, 2009.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[3] D. P. Bertsekas. Dynamic programming and suboptimal control: A survey from ADP to MPC. *European Journal of Control*, 11(4-5):310–334, 2005.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.

[5] Y. Censor, W. Chen, P. L. Combettes, R. Davidi, and G. T. Herman. On the Effectiveness of Projection Methods for Convex Feasibility Problems with Linear Inequality Constraints. *arXiv:0912.4367 [math]*, Dec. 2009.

[6] X. Dai, R. Bourdais, and H. Guéguen. Dynamic Reduction of the Iterations Requirement in a Distributed Model Predictive Control. In *IEEE 58th Conference on Decision and Control (CDC)*, 2019.

[7] P. Giselsson, M. D. Doan, T. Keviczky, B. De Schutter, and A. Rantzer. Accelerated gradient methods and dual decomposition in distributed model predictive control. *Automatica*, 49(3):829–833, 2013.

[8] P. Giselsson and A. Rantzer. On feasibility, stability and performance in distributed model predictive control. *IEEE Transactions on Automatic Control*, 59(4):1031–1036, 2014.

[9] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, volume 3. Taipei, Taiwan, 2004.

[10] D. Q. Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, 2014.

[11] A. Nedić, A. Olshevsky, and W. Shi. Achieving Geometric Convergence for Distributed Optimization Over Time-Varying Graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, Jan. 2017.

[12] A. Nedic and A. Ozdaglar. Distributed Subgradient Methods for Multi-Agent Optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, Jan. 2009.

[13] Y. Nesterov. *Lectures on Convex Optimization*, volume 137. Springer, 2018.

[14] Y. E. Nesterov. A method for solving the convex programming problem with convergence rate  $O(1/k^2)$ . In *Dokl. Akad. Nauk Sssr*, volume 269, pages 543–547, 1983.

[15] J. Nocedal. *NUMERICAL OPTIMIZATION*. Springer, 2006.

[16] C. Oberlin and S. J. Wright. Active Set Identification in Nonlinear Programming. *SIAM Journal on Optimization*, 17(2):577–605, Jan. 2006.

[17] P. Patrinos and A. Bemporad. An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *IEEE Transactions on Automatic Control*, 59(1):18–33, 2014.

[18] W. Shi, Q. Ling, G. Wu, and W. Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

[19] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. *submitted to SIAM Journal on Optimization*, 2:3, 2008.

[20] E. L. S. Wong. *Active-Set Methods for Quadratic Programming*. PhD thesis, UC San Diego, 2011.

[21] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson. A survey of distributed optimization. *Annual Reviews in Control*, 2019.

[22] K. Zhang and Y. Shi. Adaptive model predictive control for a class of constrained linear systems with parametric uncertainties. *Automatica*, 117:108974, 2020.