A Generalized Stopping Criterion for Real-Time MPC with Guaranteed Stability

Kristína Fedorová, Yuning Jiang, Juraj Oravec, Colin N. Jones, Michal Kvasnica

Abstract— Most of the real-time implementations of the stabilizing optimal control actions suffer from the necessity to provide high computational effort. This paper presents a cutting-edge approach for real-time evaluation of linear-quadratic model predictive control (MPC) that employs a novel generalized stopping criterion, achieving asymptotic stability in the presence of input constraints. The proposed method evaluates a fixed number of iterations independent of the initial condition, eliminating the necessity for computationally expensive methods. We demonstrate the effectiveness of the introduced technique by its implementation of two widely-used first-order optimization methods: the projected gradient descent method (PGDM) and the alternating directions method of multipliers (ADMM). The numerical simulation confirmed a significantly reduced number of iterations, resulting in sub-optimality rates of less than 2 %, while the effort reductions exceeded 80 %. These results nominate the proposed criterion for an efficient real-time implementation method of MPC controllers.

I. INTRODUCTION

Model Predictive Control (MPC) is an advanced and widely used control strategy that can effectively address many complex control problems in various fields, including process control [1], automotive control [2], and robotics [3]. The MPC framework operates under the paradigm of moving horizon control strategies and is executed at every control step to account for the current state measurement [4]. It builds up a mathematical model of the system to predict its behavior over a future time horizon. Then, it generates an optimal control action by solving an optimization problem subject to constraints. Therefore, MPC can handle complex system dynamics and consider input and output constraints. In practice, deploying MPC into real-time scenarios requires an efficient and reliable optimization approach to deal with the closed-loop optimal control problem [1]. To this end, two conventional approaches are used to implement the MPC control policy, parametric (explicit) MPC [5] and implicit (non-explicit) MPC [4].

The real-time implementation of explicit MPC [5] benefits from efficient division-free computation, yielding an optimal control law in the form of a piecewise affine (PWA) function for *a priori* performance certification. Despite successful implementations, see [6], and the references therein, the memory limitations and hardly-tractable construction of explicit MPC lead to the use of implicit MPC as an alternative.

Implicit MPC employs iterative methods like active set [7], interior-point [8], and first-order methods [9], [10] for online quadratic programming. Active-set methods are quick for small-medium problems but lack scalability and robustness; interior-point methods suit large problems and robustness but struggle with closed-loop warm-starting. First-order methods [11] can be easily scaled up, while its slow convergence limits its application in practice. Fortunately, their iterations are computationally cheap, and their runtimes are contingent upon the chosen stopping criterion. As a result, one can stop the iterations in advance to design a suboptimal real-time MPC scheme.

Numerous publications in this field have already been presented, where the authors have focused on establishing stopping criteria for specific optimization algorithms, ensuring asymptotic stability of suboptimal MPC. The authors of [12] introduced a stopping criterion for an interior-point algorithm called the method of centers (MC). The work [13] gained a stopping criterion formula for the dual decomposition (DD) algorithm. In [14], authors have derived a criterion for the fixed-point iteration (FPI) algorithm. The authors of [15] have presented a formula determining the iteration count for the dual accelerated gradient projection algorithm (DAGPM). The paper [16] provides a formula for projected gradient (PGM) and accelerated projected gradient method (APGM). Each of the above-mentioned papers defines a different type of stopping criterion while considering only input constraints in MPC (see Table I). The stopping criterion that guarantees asymptotical stability in the presence of state constraints can be found in [17] for the projected gradient method (PGM) and in [18] for the fast alternating minimization algorithm (FAMA) through invariant set derivation, tightening techniques and additional assumptions. To the author's knowledge, a universal stopping criterion has not yet been published to apply to various optimization algorithms.

In this paper, we formulate a generalized stopping criterion that preserves the asymptotic stability of a closed-loop system while achieving a comparable control performance to the optimal solution with a significant decrease in the number of iterations. The main contribution of this work is twofold, (i) we propose a novel generalized stopping criterion represent-

ing a maximum finite number of iterations for any first-order optimization method applied to solve the MPC problem. We prove the guarantee of the asymptotic stability of the closed-loop system using a generalized stopping criterion under mild assumptions at the cost of suboptimal control performance; (ii) we adopt the generalized stopping criterion to two widely used first-order optimization algorithms solving the MPC problems (PGDM, ADMM). The numerical case study of the well-known double-integrator benchmark is used to analyze the performance loss associated with a reduced number of iterations, accelerating the real-time evaluation.

TABLE I: Selective summary of stopping criterion for suboptimal MPC.

Reference	Constraints *possible violation	Fixed no. iterations	Algorithm
[12]	input	✓	MC
[13]	input	–	DD
[14]	input	✓	FPI
[15]	input*	✓	DAGPM
[16]	input	✓	PGM, APGM
[17]	input, state	✓	PGM
[18]	input, state	✓	FAMA

Notation: We denote the set of real numbers by $\mathbb{R}$, the set of n-dimensional real-valued vectors by $\mathbb{R}^n$, and the set of $n \times m$-dimensional real-valued matrices by $\mathbb{R}^{n \times m}$. Moreover, we denote the subspace of symmetric matrices in $\mathbb{R}^{n \times n}$ by $\mathbb{S}^n$ and the cone of positive (semi-)definite matrices by $\mathbb{S}^n_{++}(\mathbb{S}^n_+)$. For real-valed vector x and the symmetric positive definite matrix Q, $\|x\|^2_Q := x^\top Q x$.

II. Preliminaries

Consider a linear time-invariant (LTI) system in a discrete-time domain having a form
$$x(t + T_s) = Ax(t) + Bu(t), \qquad (1)$$
where $x \in \mathbb{R}^{n_x}$ is a system state vector, $u \in \mathbb{R}^{n_u}$ is a vector of control actions, $A \in \mathbb{R}^{n_x \times n_x}$ is system matrix, $B \in \mathbb{R}^{n_x \times n_u}$ is input matrix, and T_s is the sampling time. The corresponding linear-quadratic MPC problem is given by
$$V(x_0) = \min_{x,u} \|x_N\|^2_P + \sum_{k=0}^{N-1} \left(\|x_k\|^2_Q + \|u_k\|^2_R \right)$$
$$\text{s.t.} \begin{cases} x_{k+1} = Ax_k + Bu_k, \ u_k \in \mathbb{U}_k, \\ \forall k \in \{0, \ldots, N-1\} \\ x_0 = x(t), \end{cases} \qquad (2)$$
where the decision variables $x = [x_0^\top, \ldots, x_N^\top]^\top \in \mathbb{R}^{N n_x}$ and $u = [u_0^\top, \ldots, u_{N-1}^\top]^\top \in \mathbb{R}^{N n_u}$ are the sequences of the predicted system states and control actions, respectively, with $x(t) \in \mathbb{R}^{n_x}$ as a state measurement. In (2), V denotes the minimized value function, $\mathbb{U}_k \subset \mathbb{R}^{n_u}$ is the set of the constrained control inputs, the matrices $Q, P \in \mathbb{R}^{n_x \times n_x}$, $R \in \mathbb{R}^{n_u \times n_u}$ are given tuning parameters, and N is a finite prediction horizon.

Assumption 1 *We assume that for MPC design problem in (2) hold:*

- *the matrix pair (A, B) is controllable,*
- *the penalty factors Q, P, and R are symmetric positive definite matrices,*
- *terminal penalty P is computed as a solution of the matrix Riccati equation, i.e.,*
  $$P = A^\top P A - (A^\top PB)(R + B^\top PB)^{-1} + (B^\top PA) + Q,$$
- *the sets $\mathbb{U}_k$ of control inputs are convex and closed, containing origin in their strict interiors.*

Assumption 1 enforces the strong convexity of the MPC design problem in (2), leading to a unique optimal solution. Moreover, under Assumption 1, the feasible solution of the MPC design problem in (2) leads to the asymptotic stability of the closed-loop LTI system in (1). As the state constraints are not considered in (2), the recursive feasibility is satisfied by design.

The MPC design problem in (2) has the form of the optimization problem of the quadratic programming (QP) in the general form
$$V(x_0) = \min_u J(u; x_0) = \frac{1}{2} \begin{bmatrix} u \\ x_0 \end{bmatrix}^\top \begin{bmatrix} H & S \\ S^\top & D \end{bmatrix} \begin{bmatrix} u \\ x_0 \end{bmatrix} \qquad (3)$$
$$\text{s.t.} \ u \in \mathbb{U} := \mathbb{U}_0 \times \cdots \times \mathbb{U}_{N-1},$$
where $\mathbb{U} \subset \mathbb{R}^{N n_u}$ is the set of the constrained sequence of the control inputs, $H \in \mathbb{S}^{N n_u}_{++}$ is a symmetric positive definite matrix defined as
$$H = \underbrace{\begin{bmatrix} R & & \\ & \ddots & \\ & & R \end{bmatrix}}_{\widetilde{R}} + \underbrace{\begin{bmatrix} \Phi_1 \\ \vdots \\ \Phi_N \end{bmatrix}^\top \underbrace{\begin{bmatrix} Q & & \\ & \ddots & \\ & & P \end{bmatrix}}_{\widetilde{Q}} \begin{bmatrix} \Phi_1 \\ \vdots \\ \Phi_N \end{bmatrix}}_{\Phi} \qquad (4)$$
such that $\Phi_k = [A^{k-1}B, \cdots, AB, B, 0, ..., 0] \in \mathbb{R}^{n_x \times N n_u}$. Then, matrices $S \in \mathbb{R}^{n_x \times N n_u}$ and $D \in \mathbb{R}^{n_x \times n_x}$ are constructed as $S = \Psi^\top \widetilde{Q} \Phi$ and $D = \Psi^\top \widetilde{Q} \Psi$ for $\Psi^\top = [A, \ldots, A^N]^\top$.

The main idea of MPC design in receding horizon policy is to solve Problem (3) within each sampling time to determine an optimal sequence of control actions $u^\star$ for a given initial condition x_0, and then, apply the first input $u_0^\star$ to the controlled plant.

Assumption 2 *For the MPC problem in (2), the set of feasible initial conditions is (sub)set of the corresponding region of attraction. Technically, the terminal penalty in (2) is determined to satisfy Theorem 1 in [19].*

Solving problem (3) to the optimal solution under Assumptions 1, 2 leads to the Lyapunov descent [4]
$$V(x_1) \leq V(x_0) - \|x_0\|^2_Q, \qquad (5)$$
where $x_1 = Ax_0 + Bu_0^\star$, ensuring the asymptotic stability for closed-loop systems under the receding horizon MPC control policy. Note the solution of QP in (3) leads to the $u_0^\star = f(x_0^\star)$, where $f : \mathbb{U}^{n_u} \to \mathbb{R}^{n_x}$ has the form of the piecewise affine (PWA) function, see [20].

Dealing with (3), using an online solver to achieve the optimal solution $u^\star$ can be computationally intractable within the given sampling time. On the other hand, the real-time

implementation of iterative optimization procedures yields suboptimal solutions, as we need to stop the algorithm after reaching a certain number of iterations. Thereafter, (5) cannot be applied to ensure asymptotic stability in general. In the following section, we will propose a generalized stopping criterion with a fixed number of iterations for an arbitrary linearly convergent optimization algorithm to solve (3) such that an ϵ-suboptimal solution results in an asymptotic stability guarantee in (5).

III. GENERALIZED STOPPING CRITERION

We consider solving the MPC problem (3) by using the real-time implementation of the first-order optimization algorithms online. This yields a sequence of suboptimal solutions by reaching a predefined finite number of iterations m during every sampling period. Such a suboptimal solution is defined as u_0^m at the current time instant and is applied into the closed-loop LTI system

$$x_0^+ = Ax_0 + Bu_0^m. \tag{6}$$

To guarantee the asymptotic stability of the real-time MPC, we need to augment (5) by term $V(x_0^+)$ into the form

$$V(x_0^+) \le V(x_0) - \left(\|x_0\|_Q^2 - V(x_0^+) + V(x_1) \right). \tag{7}$$

Enforcing (7) to hold, we need the following property of $V(\cdot)$ representing Lipschitz continuity property

$$|V(\widetilde{x}_1) - V(\widetilde{x}_2)| \le \eta_1 \|\widetilde{x}_1 - \widetilde{x}_2\|_2 + \frac{\eta_2}{2} \|\widetilde{x}_1 - \widetilde{x}_2\|_2^2 \tag{8}$$

providing for the properly selected real-valued constants $\eta_1 \ge 0$ and $\eta_2 > 0$. Note, (8) is satisfied by design, if Assumption 1 holds.

Proposition 1 ([21]) *In the origin, such a local neighborhood $\mathcal{B}_r$ with radius r exists, that if $\widetilde{x}_1, \widetilde{x}_2 \in \mathcal{B}_r$ holds, then no constraints are active at the solution of* (3).

Proposition 1 implies that for any $x \in \mathcal{B}_r$, the value function is reduced to $V(x) = x^\top P x$ and $\eta_1 = 0$ in (8) if $\widetilde{x}_1, \widetilde{x}_2 \in \mathcal{B}_r$, i.e., holds

$$|V(\widetilde{x}_1) - V(\widetilde{x}_2)| \le \frac{\eta_2}{2} \|\widetilde{x}_1 - \widetilde{x}_2\|_2^2. \tag{9}$$

Next, we adjust (8) to express $|V(x_0^+) - V(x_1)|$ formula complemented with applying the state update resulting in

$$
\begin{aligned}
|V(x_0^+) - V(x_1)| \le &\eta_1 \|B(u_0^m - u_0^\star)\|_2 \\
&+ \eta_2 \|B(u_0^m - u_0^\star)\|_2^2.
\end{aligned}
\tag{10}
$$

Substituting $u_0^m = \Gamma u^m$ and $u_0^\star = \Gamma u^\star$ for $\Gamma = [I \ 0 \ ... \ 0] \in \mathbb{R}^{n_u \times Nn_u}$, we rewrite

$$B(u_0^m - u_0^\star) = B\Gamma(u^m - u^\star) \tag{11}$$

in (10) such that holds

$$
\begin{aligned}
|V(x_0^+) - V(x_1)| \le &\eta_1 \|B\Gamma(u^m - u^\star)\|_2 \\
&+ \eta_2 \|B\Gamma(u^m - u^\star)\|_2^2.
\end{aligned}
\tag{12}
$$

If we take the first-order optimization algorithm with a linear convergence rate, then the following applies

$$\|u^{m+1} - u^\star\|_2 \le \kappa \|u^m - u^\star\|_2, \tag{13}$$

where $\kappa < 1$ represents convergence factor, u^m stands for QP solution at m-th iteration of algorithm, and $u^\star$ is the

optimal solution of QP in (3). Consequently, for a given m iterations, we have

$$\|u^m - u^\star\|_2 \le \kappa^m \|u^0 - u^\star\|_2. \tag{14}$$

Specifically, under the assumption of $Q \in \mathbb{S}_+^{n_x}$, $R \in \mathbb{S}_{++}^{n_u}$, and $\mathbb{U}$ is a compact convex set, we have two widely-used first-order method algorithms: (i) the projected gradient descent method (PGDM) and (ii) the alternating direction method of multipliers (ADMM), achieving linear convergence.

Example 1 (Projected Gradient Descent Method)
The projected gradient descent method is an extension of the original gradient descent method by including the constraints through projection into the constraint set $\mathbb{U}$. Applying the PGDM algorithm to the MPC design problem in the form of QP (3) results in an iteration

$$u^{m+1} := \ \mathrm{Proj}_{\mathbb{U}}(u^m - \alpha \nabla J(u^m, x_0)), \tag{15}$$

where $\alpha > 0$ is the step size and u^m is the initial guess at m-th iteration. We consider $J(u, x_0)$ is μ-strongly convex and L-smooth, such that μ and L are computed by evaluating the eigenvalue of H in (3). If we determine the step size $\alpha = \frac{1}{L}$, then we have linear convergence of PGDM given by

$$\kappa = 1 - \frac{\mu}{L} \tag{16}$$

such that $\kappa < 1$, see [22].

Example 2 (Alternating Direction Method of Multipliers)
The alternating direction method of multipliers is an algorithm described in more detail in [23]. Generally, the algorithm distributes the original optimization problem into smaller-scaled problems that are solved quickly. Applying ADMM to QP problem in (3) leads to following iterations

$$u^{m+1} = \min_u \ J(u; x_0) + u^\top \lambda^m + \frac{\rho}{2} \|u - v^m\|_2^2, \tag{17a}$$

$$v^{m+1} = \min_{v \in \mathbb{U}} \ -v^\top \lambda^m + \frac{\rho}{2} \|u^{m+1} - v\|_2^2, \tag{17b}$$

$$\lambda^{m+1} = \lambda^m + \rho(u^{m+1} - v^{m+1}), \tag{17c}$$

for some initial guesses of global coordination variable $v^m \in \mathbb{R}^{Nn_u}$, dual variable $\lambda^m \in \mathbb{R}^{Nn_u}$, and tuning parameter $\rho > 0$. The evaluation of the linear convergence rate of ADMM for QPs refers to Section IV of [24], resulting in

$$\kappa = \frac{1}{2} \|2M - I\|, \tag{18}$$

where the matrix M depends on the formulation of the MPC problem in (3) and has the form

$$M = \widetilde{G} - \widetilde{G}(I + \widetilde{G})^{-1}\widetilde{G} \tag{19}$$

for $\widetilde{G} = \rho G H^{-1} G^\top$. The matrix $G \in \mathbb{R}^{2Nn_u \times Nn_u}$ is determined by the matrix representation $Gu \le w$ of input constraints from MPC problem in (3), where vector $w \in \mathbb{R}^{2Nn_u}$.

Then, combining (12) with (14) for Examples 1, 2 yields inequality

$$
\begin{aligned}
|V(x_0^+) - V(x_1)| \le &\bar{\eta}_1 \|u^m - u^\star\|_2 + \bar{\eta}_2 \|u^m - u^\star\|_2^2 \\
\le &\bar{\eta}_1 \kappa^m \|u^0 - u^\star\|_2 + \bar{\eta}_2 \kappa^{2m} \|u^0 - u^\star\|_2^2
\end{aligned}
\tag{20}
$$

for the real-valued constants $\bar{\eta}_1 = \eta_1\sqrt{\nu_{\max}(\Gamma^\top B^\top B\Gamma)}$ and $\bar{\eta}_2 = \frac{\eta_2}{2}\nu_{\max}(\Gamma^\top B^\top B\Gamma)$. Here, $\nu_{\max}(\cdot)$ defines the maximal eigenvalue for a given matrix.

Assumption 3 *We have the real-valued constant $\gamma > 0$ that for any feasible solution x_0 of* (3) *satisfies*

$$\|u^\star\|_2 \le \gamma\|x_0\|_Q. \tag{21}$$

Since $u^\star$ is a piecewise affine function of x_0, the constant γ in (21) can be determined offline, i.e., before running the real-time MPC control (see e.g. [17]). Furthermore, the initialization also satisfies $\|u^0\|_2 \le \gamma\|x_0\|_Q$ enforced by re-scale in the algorithm, if necessary. Then, (20) reads

$$
\begin{aligned}
|V(x_0^+) - V(x_1)| &\le 2\bar{\eta}_1\gamma\kappa^m\|x_0\|_Q \\
&\quad + 2\bar{\eta}_2\gamma^2\kappa^{2m}\|x_0\|_Q^2.
\end{aligned} \tag{22}
$$

Assumption 4 *We assume that if we have non-empty convex set $\mathbb{Q} := \{x_0 \in \mathbb{R}^{n_x} \mid \|x_0\|_Q \le 1\}$, then $\mathbb{Q} \subseteq \mathcal{B}_r$ holds.*

Note that Assumption 4 can be enforced by adjusting the scaling matrices Q and R of MPC problem (2) as shown in Figure 1.

Consequently, a finite integer-valued parameter $\overline{m}$ exists such that the inequality (22) is satisfied. Finally, this finite number of iterations $\overline{m}$ guarantees the asymptotic stability of the real-time implementation of the MPC designed by (2).

Theorem 1 (Generalized stopping criterion) *Let Assumptions 1–4 hold. If constant $\overline{m} \in \mathbb{N}$ satisfies*

$$\overline{m} > \log\left(2\bar{\eta}_1\gamma + 2\bar{\eta}_2\gamma^2\right) / \log\left(1/\kappa\right), \tag{23}$$

then the control action $u_0 := \Gamma u^{\overline{m}}$ implemented to the LTI system (1) *ensures the asymptotic stability of the MPC in* (3).

Proof: The proof is established in two steps. First, if $\|x_0\|_Q > 1$ holds, the upper bound on (22) is given by

$$|V(x_0^+) - V(x_1)| \le \underbrace{\left(2\bar{\eta}_1\gamma + 2\bar{\eta}_2\gamma^2\right)\kappa^{\overline{m}}}_{\beta}\|x_0\|_Q^2, \tag{24}$$

for $\|x_0\|_Q^2 \ge \|x_0\|_Q$ and for $\kappa < 1$ leading to $\kappa^{\overline{m}} > \kappa^{2\overline{m}}$. Next, by combining (7) with (24), we have

$$V(x_0^+) \le V(x_0) - \left((1-\beta)\|x_0\|_Q^2\right), \tag{25}$$

where $(1-\beta) > 0$ is sufficient to achieve asymptotic stability. Therefore, the minimum number of iterations $\overline{m}$ is determined from following inequality

$$\beta = \left(2\bar{\eta}_1\gamma + 2\bar{\eta}_2\gamma^2\right)\kappa^{\overline{m}} < 1, \tag{26}$$

that straightforwardly implies (23) to hold.

Secondly, if $\|x_0\|_Q \le 1$, using Proposition 1 and Assumption 4, we have (22) in form

$$|V(x_0^+) - V(x_1)| \le 2\bar{\eta}_2\gamma^2\kappa^{2\overline{m}}\|x_0\|_Q^2 \le \beta\|x_0\|_Q^2, \tag{27}$$

resulting into

$$\overline{m} > \frac{\log(2\bar{\eta}_2\gamma^2)}{2\log(1/\kappa)}. \tag{28}$$

Consequently, the following inequality holds

$$\overline{m} > \frac{\log\left(2\bar{\eta}_1\gamma + 2\bar{\eta}_2\gamma^2\right)}{\log\left(1/\kappa\right)} > \frac{\log(2\bar{\eta}_2\gamma^2)}{2\log(1/\kappa)}, \tag{29}$$

that concludes the proof. ∎

Even though the algorithm to solve MPC problem (3) is stopped after $\overline{m}$ iterations determined by the generalized criterion in (23), enforcing the asymptotic stability by this approach leads to the conservative control performance. The performance loss originates in (24) representing the upper bound on the original requirement formulated in (22), i.e., represents just the necessary condition. Therefore, there may exist fewer iterations guaranteeing asymptotic stability of the closed-loop LTI system in (1) under receding horizon MPC control policy.
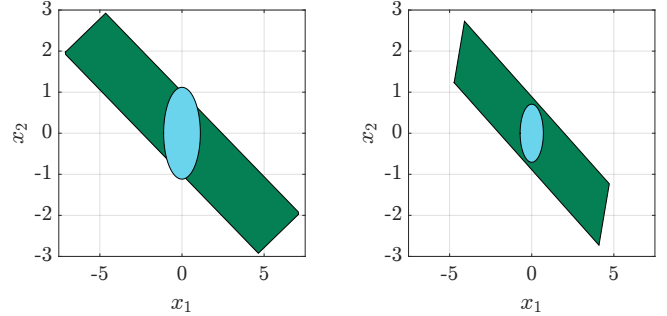
(a) Assumption 4 violated. (b) Assumption 4 holds.

Fig. 1: Example of set $\mathbb{Q}$ (blue) and LQR-based control invariant set (green) for two setups of matrix pairs (Q, R) of MPC design problem (2): a) $\mathbb{Q}$ is not the subset of the LQR-based control invariant set, b) LQR-based set contains set $\mathbb{Q}$.

IV. NUMERICAL CASE STUDY

To analyze the properties of the proposed stopping criterion, we adopted the well-known benchmark system of the double integrator system, which has the system matrices

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix},$$

and the model predictive controller in (3) is designed with weight matrices

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad P = \begin{bmatrix} 2.367 & 1.118 \\ 1.118 & 2.588 \end{bmatrix}, \quad R = 1,$$

where P is computed as a solution to the matrix Riccati equation. Furthermore, we define the input constraints in (3) as $\mathbb{U} = \{u \in \mathbb{R} \mid -1 \le u \le 1\}$, and finite prediction horizon as $N = 10$.

The simulations were executed in MATLAB. For solving the double integrator problem with ADMM-based MPC and nominal MPC, we incorporated the QUADPROG solver. Note that the PGDM method can be implemented without the need for any external optimization tools. Furthermore, we have designed the ADMM-based MPC algorithm in a way where local step (17a) was completely distributed in the time domain to N computing units. Both algorithms disposed of two stopping criteria: either satisfying the convergence condition or meeting a maximum number of iterations.

The results for PGDM-based MPC were obtained using the following setup of the parameters $L = 3\,200$ and $\mu =$

2 leading to $\kappa_{\text{PGDM}} = 0.9992$ based on (16). Then, the maximum number of iterations was calculated according to Theorem 1 by (23) as $\overline{m}_{\text{PGDM}} = 172$ using parameters $\gamma = 1$, $\eta_1 = 0.4$, $\eta_2 = 0.1$. The convergence condition was defined as $\|\nabla J\| \leq \varepsilon$ with $\varepsilon = 10^{-3}$.

The parameters for the ADMM-based MPC algorithm were set based on (18) as $\kappa_{\text{ADMM}} = 0.9980$ and formula from Section IV of [24] as $\rho = 3.1231$. The maximum number of iterations was determined according to Theorem 1 as $\overline{m}_{\text{ADMM}} = 14$ considering $\gamma = 1$, $\eta_1 = 0.2$, $\eta_2 = 0.3$. Furthermore, convergence condition was specified as $\|\lambda^{l+1} - \lambda^l\| < \varepsilon$ for $\varepsilon = 10^{-3}$.
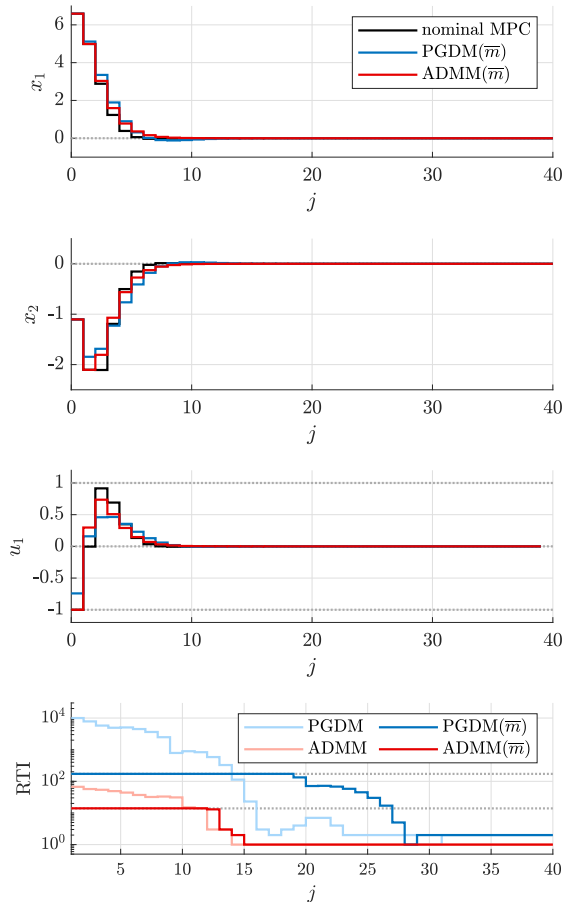

Fig. 2: Control performance of the nominal MPC (black), PGDM-based MPC (blue), and ADMM-based MPC (red) after $\overline{m}$ iterations. The bottom graph depicts the number of real-time iterations (RTI) for ADMM and PGDM for unbounded stopping criterion (light blue, light red) and for predefined $\overline{m}$ (dark blue, dark red) iterations.

The results of the numerical simulations of the closed-loop control for the proposed real-time approach are illustrated in Figure 2, where we can see the comparison of control profiles for nominal MPC, $\overline{m}$-bounded ADMM-based MPC and $\overline{m}$-bounded PGDM-based MPC. To validate their accuracy, the ADMM-based and PGDM-based MPC algorithms were also evaluated for a maximum number of iterations $m = 15\,000$

(we will address it as "unbounded"). However, these control performance results are not depicted in Figure 2 due to over-lapping with the control profiles generated by the nominal MPC that serves as the reference profile without any performance loss. As shown in Figure 2, the ADMM-based and PGDM-based MPC approaches lead to suboptimal solutions for $\overline{m}$. Nevertheless, each approach preserves the asymptotic stability and drives the system states into its origin. The last graph of Figure 2 shows the number of real-time iterations (RTI) per simulation step j of unbounded ADMM-based and unbounded PGDM-based MPC and with $\overline{m}$-bounded (suboptimal) approaches as defined above. As visualized, the number of iterations for unbounded PGDM-based MPC is extremely higher than for $\overline{m}$-bounded PGDM-based MPC. Numerically $47\,794$ vs. $3\,620$ iterations per visualized 40-steps simulation resulting in $92\,\%$ decrease in the number of iterations. In comparison, the unbounded ADMM-based MPC executes 466 iterations altogether while $\overline{m}$-bounded ADMM-based MPC performs 198 iterations ($67\,\%$ decrease). To conclude, even a drastically lower number of iterations yields the desired result with stability guarantees at the cost of suboptimal performance.

TABLE II: Comparison of suboptimality rate and a number of iterations for unbounded and $\overline{m}$-bounded methods.

Method	m_{avg} [–]	m_{max} [–]	δ_{avg} [%]	δ_{max} [%]
ADMM	39	$2\,331$	0	0
ADMM($\overline{m}$)	6	14	0.7	0.9
PGDM	$1\,873$	$9\,786$	0	0
PGDM($\overline{m}$)	99	172	0.5	1.2

The stability validation of our proposed approach involves checking the inequality (22) at each simulation step. Figure 3 depicts real-time $|V_j(x_j^+) - V_j(x_j)|$ values alongside an upper bound derived as $2\bar{\eta}_1\gamma\kappa^m\|x_0\|_Q + 2\bar{\eta}_2\gamma^2\kappa^{2m}\|x_0\|_Q^2$. The plot demonstrates that the real-time values of $|V_j(x_j^+) - V_j(x_j)|$ remain under the upper bound in each step j.

A comparative summary is provided in Table II, which analyzes suboptimality rates and iteration counts for unbounded and $\overline{m}$-bounded methods. The investigation covers control performance and evaluated iterations using metrics including the average number of iterations per simulation step m_{avg}, the maximum number of iterations per step m_{max}, the average suboptimality rate per step δ_{avg}, and the maximum suboptimality rate per step δ_{max}. The analysis involves 201 initial conditions x_0 representing distinct segments of the PWA control law. Suboptimality rates $\delta_j(x_0(i))$ for step j and initial condition i are calculated as $\delta_j(x_0(i)) = \left(|V_j(x_j^+) - V_j(x_j)|\right) / \left(|V_{j+1}(x_1)|\right)$.

For the unbounded methods, both ADMM-based and PGDM-based MPCs exhibit zero average and maximum suboptimality rates due to $\delta_{\text{avg}} < 10^{-6}$ and $\delta_{\text{max}} < 10^{-6}$. Comparable suboptimality rates are observed for $\overline{m}$-bounded ADMM and PGDM methods. While ADMM has

slightly higher suboptimality, it demonstrates lower variance than PGDM. Analysis of m_{avg} and m_{max} highlights that unbounded methods require substantially more iterations for optimal solutions, whereas $\overline{m}$-bounded techniques execute significantly fewer iterations (around 94% decrease for PGDM and 84% decrease for ADMM) while ensuring asymptotic stability.
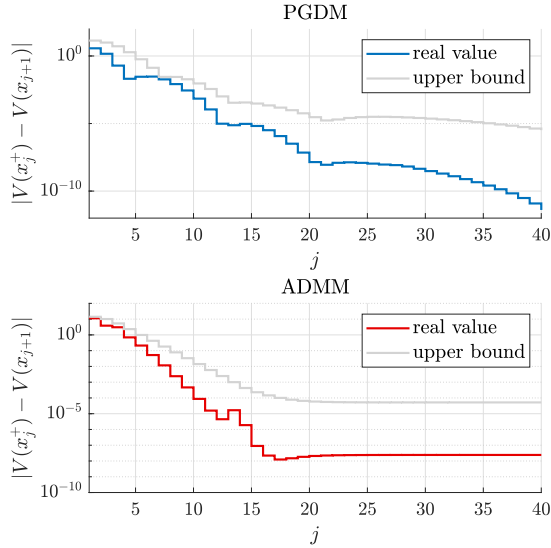

Fig. 3: Evolution of $|V(x_j^+) - V(x_{j+1})|$ for PGDM-based MPC (blue) and ADMM-based MPC (red) with $\overline{m}$ iterations supplemented by the respective upper bound for each method from (22) (grey).

V. Conclusions

This paper has presented a novel method for the real-time evaluation of linear-quadratic MPC with input constraints, offering a generalized stopping criterion representing a fixed number of iterations for first-order optimization algorithms. The proposed method establishes the asymptotic stability of the real-time solution. This approach significantly reduces the maximum number of iterations required for real-time evaluation of MPC while preserving the acceptable suboptimality rates. The proposed method was analyzed using a benchmark double-integrator problem, demonstrating an average reduction in the number of iterations by up to 80 % leading to less than 2 % suboptimality rate for the worst-case control scenarios. The analyzed results indicate that using a fixed number of iterations as a generalized stopping criterion has the potential to significantly decrease the run times of real-time MPC for the systems with fast dynamics and/or in the framework of the remote battery-supplied control platforms. Our future work will be focused on the elimination of two major drawbacks of our approach: (i) parameters gained by exhaustive offline numerical computations, (ii) state constraints were not considered.

References

[1] S. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[2] D. Hrovat, S. Di Cairano, E. Tseng, and I. Kolmanovsky, "The development of model predictive control in automotive industry: A survey," *Proceedings of the IEEE International Conference on Control Applications*, pp. 295–302, 10 2012.

[3] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, "Review on model predictive control: an engineering perspective," *The International Journal of Advanced Manufacturing Technology*, vol. 117, pp. 1327 – 1349, 2021.

[4] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*. Nob Hill Publishing Madison, WI, 2017, vol. 2.

[5] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadric regulator for constrained systems," *Automatica*, vol. 38, pp. 3–20, 2002.

[6] I. Pappas, D. Kenefake, B. Burnak, S. Avraamidou, H. S. Ganesh, J. Katz, N. A. Diangelakis, and E. N. Pistikopoulos, "Multiparametric programming in process systems engineering: Recent developments and path forward," *Frontiers in Chemical Engineering*, vol. 2, 1 2021.

[7] A. Bemporad, "A quadratic programming algorithm based on nonnegative least squares with applications to embedded model predictive control," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 1111–1116, 2015.

[8] S. J. Wright, *Primal-dual interior-point methods*. SIAM, 1997.

[9] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk, "Fast alternating direction optimization methods," *SIAM Journal on Imaging Sciences*, vol. 7, no. 3, pp. 1588–1623, 2014.

[10] P. Patrinos and A. Bemporad, "An accelerated dual gradient-projection algorithm for embedded linear model predictive control," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 18–33, 2013.

[11] A. Beck, *First-Order Methods in Optimization*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2017.

[12] L. McGovern and E. Feron, "Closed-loop stability of systems driven by real-time, dynamic optimization algorithms," in *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, vol. 4, 1999, pp. 3690–3696 vol.4.

[13] P. Giselsson and A. Rantzer, "Distributed model predictive control with suboptimality and stability guarantees," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 7272–7277.

[14] K. Graichen, "A fixed-point iteration scheme for real-time model predictive control," *Automatica*, vol. 48, no. 7, pp. 1300–1305, 2012.

[15] M. Rubagotti, P. Patrinos, and A. Bemporad, "Stabilizing linear model predictive control under inexact numerical optimization," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1660–1666, 2014.

[16] D. Liao-McPherson, T. Skibik, J. Leung, I. Kolmanovsky, and M. M. Nicotra, "An analysis of closed-loop stability for linear model predictive control based on time-distributed optimization," *IEEE Transactions on Automatic Control*, vol. 67, no. 5, pp. 2618–2625, 2022.

[17] J. Leung, D. Liao-McPherson, and I. V. Kolmanovsky, "A computable plant-optimizer region of attraction estimate for time-distributed linear model predictive control," in *2021 American Control Conference (ACC)*, 2021, pp. 3384–3391.

[18] Y. Yang, Y. Wang, C. Manzie, and Y. Pu, "Sub-optimal mpc with dynamic constraint tightening," *IEEE Control Systems Letters*, vol. 7, pp. 1111–1116, 2023.

[19] D. Limon, T. Alamo, F. Salas, and E. Camacho, "On the stability of constrained mpc without terminal constraint," *IEEE Transactions on Automatic Control*, vol. 51, no. 5, pp. 832–836, 2006.

[20] F. Borrelli, *Constrained Optimal Control of Linear and Hybrid Systems*. Springer Berlin Heidelberg, 2017.

[21] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[22] S. Bubeck, "Convex optimization: Algorithms and complexity," *Foundations and Trends® in Machine Learning*, vol. 8, no. 3-4, pp. 231–357, 2015.

[23] 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.*, vol. 3, no. 1, pp. 1–122, 2011.

[24] E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson, "Optimal Parameter Selection for the Alternating Direction Method of Multipliers (ADMM): Quadratic Problems," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 644–658, 2015.