

Real-time proximal gradient method for linear MPC*

Ruben Van Parys¹ and Goele Pipeleers¹

Abstract—This paper presents a real-time implementation of the proximal gradient method (PGM) in a model predictive control (MPC) setting. In each control update only one iteration of the PGM is performed, while a next update is warm-started using the solution of the previous one. When applied to linear time-invariant (LTI) systems with simple input constraints, the resulting control law becomes extremely simple and offers possibilities to obtain fast control rates even on resource-constrained hardware. The paper provides a proof of closed-loop stability of the real-time PGM applied to LTI systems. A numerical simulation example validates the resulting closed-loop performance.

I. INTRODUCTION

Model predictive control (MPC) is a popular control approach that relies on the solution of an optimal control problem (OCP) with a receding horizon [1]. It allows to handle physical constraints in a systematic manner and often leads to improved control performance. However, evaluating the model predictive control law is often computationally demanding. It requires solving an optimization problem in real-time, which is challenging in case the problem is large, the solution is needed quickly, and the problem must be solved on resource-constrained embedded computing hardware such as programmable logic controllers (PLCs) or field-programmable gate arrays (FPGAs).

An intuitive approach to reduce the computational load of MPC is solving the OCP approximately by using a fixed number of optimization iterations during each control update. In a next update, the previous solution is used to warm-start the optimization algorithm to further reduce the suboptimality of the predicted trajectories. This *suboptimal* MPC idea was already investigated in the 90s [2], [3] and has known many different implementations since then. One particular example hereof is the celebrated real-time iteration scheme [4], [5], which uses only one sequential quadratic programming (SQP) iteration per control update in order to provide quick response to disturbances. Although the method shows large potential for a wide application domain, the underlying Newton-based optimization iterations are computationally demanding. Each iteration requires the solution of a linear system of equations, which is challenging to implement on resource-constrained hardware.

*This work benefits from KU Leuven-BOF PFV/10/002 Centre of Excellence: Optimization in Engineering (OPTEC); from the project G0C4515N of the Research Foundation-Flanders (FWO-Flanders); and from the KU Leuven Research project C14/15/067: B-spline based certificates of positivity with applications in engineering. Ruben Van Parys is a PhD fellow of FWO-Flanders. Flanders Make is the Flemish strategic research centre for the manufacturing industry.

¹The authors are with the MECO Research Team, Department Mechanical Engineering, KU Leuven and with the DMMS lab, Flanders Make, Leuven, Belgium. ruben.vanparys@kuleuven.be

Recently, there has been an increased interest in using first-order methods for the online solution of MPC problems [6]–[11]. In contrast to active-set or interior-point schemes, first-order methods often do not require the solution of a linear system at every iteration. Furthermore, these methods have certain features that make them amenable to fixed-point implementation and they can be efficiently parallelized. All these properties make first-order schemes the ultimate choice to obtain faster MPC on cheaper hardware [7]. Using first-order methods in suboptimal MPC schemes has already been addressed in literature [9]–[12]. These approaches mainly focus on finding lower bounds on the number of optimization iterations per sample instance in order to guarantee a given level of optimality.

This paper focuses on a real-time iteration scheme for linear MPC based on the proximal gradient method (PGM). The PGM is an extension of the (projected) gradient method and a popular first-order method for linear MPC with simple input constraints [7], [10]. In a real-time iteration scheme, only one PGM iteration is performed per control update, while the next update is warm-started using the solution of the previous one. When applying the real-time PGM scheme to linear time-invariant (LTI) systems with simple input constraints such as box constraints, the resulting control law becomes extremely simple. This offers possibilities to obtain fast control rates even on resource-constrained hardware. The paper proves closed-loop stability of both the system's state and the input trajectory, i.e. the real-time PGM scheme guarantees that the system's state is attracted to a stable equilibrium while the suboptimal input trajectory is converging towards its optimal value. In order to retrieve this outcome, the paper extends the current convergence analysis of the PGM with a result relating the monotonic decrease in objective function to the distance to the optimum. Numerical simulations of the real-time PGM analyze its closed-loop performance compared to the performance of an optimal MPC scheme.

Section II describes the problem addressed in this work, while Section III introduces the proximal gradient method. Section IV embeds this method in a real-time MPC setting and proves asymptotic stability of the resulting closed-loop system. Thereafter, the closed-loop performance is validated with a numerical example in Section V. Finally, Section VI draws concluding remarks.

II. PROBLEM FORMULATION

This paper addresses the control of a discrete-time linear time-invariant (LTI) system of the form

$$x_{k+1} = Ax_k + Bu_k, \quad (1)$$

with current and next state $x_k, x_{k+1} \in \mathbb{R}^n$ and input $u_k \in \mathbb{R}^m$. The input is confined to a convex compact set $\mathbb{U} \subset \mathbb{R}^m$ containing the origin in its interior, which typically represents actuator limitations. Given a measurement or estimate of the current state x_k , an MPC controller will compute an input trajectory $q = \{q(0), \dots, q(N-1)\}$ over a time horizon of N samples. The state trajectory following from x_k, q and the integration of (1) is indicated by $\phi(\cdot, x_k, q)$. An optimal trajectory is retrieved by solving an optimal control problem of the form

$$\min_{q \in \mathcal{Q}(x_k)} V_N(x_k, q) = \sum_{i=0}^{N-1} l(\phi(i, x_k, q), q(i)) + V_f(\phi(N, x_k, q)). \quad (2)$$

The set $\mathcal{Q}(x_k)$ represents the set of feasible trajectories:

$$\mathcal{Q}(x_k) = \{q \mid q(i) \in \mathbb{U}, i = 0, \dots, N-1, \phi(N, x_k, q) \in \mathbb{X}_f\},$$

where \mathbb{X}_f is a convex compact set representing a possible terminal state constraint to guarantee closed-loop stability. With $q^*(x_k) = \{q^*(x_k, 0), \dots, q^*(x_k, N-1)\}$ the optimal solution of (2), an optimal MPC approach will apply the control input $u_k = q^*(x_k, 0)$ to the system.

Similar as in a classical linear MPC formulation, the following assumption is made:

Assumption 1: The functions l and V_f are quadratic and of the following form:

$$l(x, u) = \frac{1}{2}x^T Qx + \frac{1}{2}u^T Ru, \\ V_f(x) = \frac{1}{2}x^T Px,$$

where $Q \in \mathbb{R}^{n \times n}$, $R \in \mathbb{R}^{m \times m}$ and $P \in \mathbb{R}^{n \times n}$ are positive definite, i.e. $Q \succ 0$, $R \succ 0$, $P \succ 0$. Furthermore, P and \mathbb{X}_f are chosen such that $\forall x \in \mathbb{X}_f, \exists \kappa_f(x) \in \mathbb{U}$:

$$(i) Ax + B\kappa_f(x) \in \mathbb{X}_f, \quad (3a)$$

$$(ii) V_f(Ax + B\kappa_f(x)) - V_f(x) \leq -l(x, \kappa_f(x)). \quad (3b)$$

After eliminating the states, the value function V_N is rewritten as a quadratic function in the input sequence q and current state x_k :

$$\min_{q \in \mathcal{Q}(x_k)} V_N(x_k, q) = \frac{1}{2}q^T Fq + \frac{1}{2}x_k^T Gx_k + q^T Hx_k. \quad (4)$$

Under Assumption 1 the closed-loop system when using an optimal MPC control law is exponentially stable with a region of attraction $\mathcal{X} = \{x \mid \mathcal{Q}(x) \neq \emptyset\}$ [1]. The goal of this paper is to prove closed-loop stability when using a suboptimal solution of (4) obtained by only one iteration of the proximal gradient method.

III. PROXIMAL GRADIENT METHOD

This section describes the proximal gradient method (PGM), which forms the underlying optimization strategy of the MPC approach presented in this paper. First, the considered problem type is given. Then, the PGM is described and a relation is derived between the monotonic decrease in objective value during each PGM step and the distance to

the optimum. This relation is a key element in the overall closed-loop stability proof of Section IV.

A. Problem description

Consider the problem

$$\min_q h(q) = f(q) + g(q), \quad (5)$$

where the objective function h is split in two parts, f and g , that satisfy the following assumption:

Assumption 2: The function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is closed and strongly convex with parameter $m > 0$ and ∇f is Lipschitz continuous with parameter L . The function $g : \mathbb{R}^d \rightarrow \mathbb{R}$ is proper, closed, convex and extended real-valued, i.e. $\mathbb{R} = \mathbb{R} \cup \{-\infty, +\infty\}$.

Strong convexity of f implies a *quadratic lower bound* on f :

$$f(p) \geq f(q) + \nabla f(q)^T(p - q) + \frac{m}{2}\|p - q\|_2^2 \quad \forall q, p \in \mathbb{R}^d,$$

and *strong monotonicity* of ∇f :

$$(\nabla f(q) - \nabla f(p))^T(q - p) \geq m\|q - p\|_2^2 \quad \forall q, p \in \mathbb{R}^d. \quad (6)$$

Using the Cauchy-Schwarz inequality, (6) implies

$$\|\nabla f(q) - \nabla f(p)\|_2 \geq m\|q - p\|_2 \quad \forall q, p \in \mathbb{R}^d. \quad (7)$$

Lipschitz continuity of ∇f implies a *quadratic upper bound* on f :

$$f(p) \leq f(q) + \nabla f(q)^T(p - q) + \frac{L}{2}\|p - q\|_2^2 \quad \forall q, p \in \mathbb{R}^d, \quad (8)$$

and *co-coercivity* of ∇f :

$$(\nabla f(q) - \nabla f(p))^T(q - p) \geq \frac{1}{L}\|\nabla f(q) - \nabla f(p)\|_2^2 \quad \forall q, p \in \mathbb{R}^d. \quad (9)$$

As g is possibly non-smooth, the analysis in Section III-C can only rely on the definition of a subgradient: $w \in \mathbb{R}^d$ is a subgradient of g at $q \in \mathbb{R}^d$ if

$$g(p) \geq g(q) + w^T(p - q) \quad \forall p \in \mathbb{R}^d. \quad (10)$$

The set of all subgradients of g at q is the subdifferential $\partial g(q)$.

B. The proximal gradient method

The proximal gradient method belongs to the set of proximal algorithms [13], a group of methods that use the *proximal mapping* as base operation. The proximal mapping prox_g of a function g is defined as

$$\text{prox}_g(q) = \arg\min_p g(p) + \frac{1}{2}\|p - q\|_2^2.$$

When g is proper, closed and convex, the minimizer exists and is unique, i.e. prox_g is single-valued. Furthermore, proximal mappings are *nonexpansive*:

$$\|\text{prox}_g(q) - \text{prox}_g(p)\|_2 \leq \|q - p\|_2. \quad (11)$$

Suppose g is the indicator of a closed convex set C , i.e.

$$g(q) = \mathcal{I}_C(q) = \begin{cases} 0 & q \in C, \\ +\infty & q \notin C, \end{cases}$$

then the proximal mapping of g represents the projection on C , i.e. $\text{prox}_{\mathcal{I}_C}(q) = \Pi_C(q)$.

The proximal gradient method solves problem (5) iteratively, by updating the estimate for q in each step as

$$\begin{aligned} q^+ &= \text{prox}_{\gamma g}(q - \gamma \nabla f(q)), \\ &= q - \gamma G_\gamma(q), \end{aligned} \quad (12)$$

where $G_\gamma(q) = \frac{q - q^+}{\gamma}$ is an analogue of the usual gradient to nonsmooth functions of the form (5) [14]. When $\gamma \in (0, \frac{2}{L})$ and when Assumption 2 holds, the method converges to the optimum q^* of (5) [14].

C. Monotonic decrease in objective value

This subsection derives a relation between the monotonic decrease in the objective function of (5) due to one PGM step (12) and the distance to the optimum. This is summarized in the following theorem.

Theorem 1: Consider a function $h = f + g$, where f and g satisfy Assumption 2. Suppose $q \in \mathbb{R}^d$ is mapped on $q^+ \in \mathbb{R}^d$ using the PGM step (12) with $\gamma \in (0, \frac{2}{L})$ and q^* is the minimizer of h . Then, there exists a $\beta > 0$ such that

$$h(q^+) - h(q) \leq -\beta \|q - q^*\|_2^2.$$

Proof: First the PGM step (12) is shown to reduce the distance to the optimum:

$$\begin{aligned} \|q^+ - q^*\|_2^2 &= \|\text{prox}_{\gamma g}(q - \gamma \nabla f(q)) \\ &\quad - \text{prox}_{\gamma g}(q^* - \gamma \nabla f(q^*))\|_2^2, \\ &\leq \|q - \gamma \nabla f(q) - q^* + \gamma \nabla f(q^*)\|_2^2, \\ &\leq \|q - q^*\|_2^2 - \gamma \left(\frac{2}{L} - \gamma\right) \|\nabla f(q) - \nabla f(q^*)\|_2^2, \\ &\leq \underbrace{\left(1 - \frac{2m^2}{L}\gamma + m^2\gamma^2\right)}_{\delta^2} \|q - q^*\|_2^2. \end{aligned} \quad (13)$$

The first inequality follows from the nonexpansiveness of the proximal mapping (11). The second inequality follows from co-coercivity of ∇f (9) and the third from strong monotonicity of ∇f (7). Since $\gamma \in (0, \frac{2}{L})$, it follows that $(1 - \frac{2m^2}{L}\gamma + \gamma^2 m^2) \in (0, 1)$.

Inequality (13) allows us to find a relation between $\|G_\gamma(q)\|_2^2$ and the distance to the optimum:

$$\begin{aligned} \gamma^2 \|G_\gamma(q)\|_2^2 &= \|(q - q^*) - (q^+ - q^*)\|_2^2, \\ &\geq (\|q - q^*\|_2 - \|q^+ - q^*\|_2)^2, \\ &\geq (1 - \delta)^2 \|q - q^*\|_2^2. \end{aligned} \quad (14)$$

The first inequality is the reverse triangle inequality and the second inequality follows from (13).

Finally a relation between the decrease in objective value and the distance to the optimum is derived:

$$\begin{aligned} h(q^+) - h(q) &= f(q^+) - f(q) + g(q^+) - g(q), \\ &\leq \nabla f(q)^T(q^+ - q) + \frac{L}{2} \|q^+ - q\|_2^2 \\ &\quad + g(q^+) - g(q), \\ &\leq \nabla f(q)^T(q^+ - q) + \frac{L}{2} \|q^+ - q\|_2^2 \\ &\quad + (G_\gamma(q) - \nabla f(q))^T(q^+ - q), \\ &= -\gamma \left(1 - \frac{L}{2}\gamma\right) \|G_\gamma(q)\|_2^2, \\ &\leq \underbrace{-\frac{1}{\gamma} \left(1 - \frac{L}{2}\gamma\right) (1 - \delta)^2}_{\beta} \|q - q^*\|_2^2. \end{aligned}$$

The first inequality expresses the quadratic upper bound (8) for f due to Lipschitz continuity of ∇f . The second inequality follows from the first-order optimality conditions of (12) which imply that $G_\gamma(q) - \nabla f(q) \in \partial g(q^+)$ and from the definition of a subgradient (10). The last inequality follows from (14). Since $\gamma \in (0, \frac{2}{L})$, it follows that $\beta > 0$. ■

IV. REAL-TIME PROXIMAL GRADIENT MPC

A. Real-time PGM algorithm

Optimal control problem (4) is rewritten as the following equivalent problem

$$\min_q h(x_k, q) = V_N(x_k, q) + \mathcal{I}_{\mathcal{Q}(x_k)}(q). \quad (16)$$

Note that by Assumption 1 $\nabla_q V_N(x_k, \cdot)$ is Lipschitz continuous with Lipschitz constant the maximum eigenvalue of F , i.e. $L = \lambda_{\max}(F)$, and $V_N(x_k, \cdot)$ is strongly convex with parameter the minimum eigenvalue of F , i.e. $m = \lambda_{\min}(F)$. The function $\mathcal{I}_{\mathcal{Q}(x_k)}$ represents the indicator function of $\mathcal{Q}(x_k)$ and is proper, closed and convex when $x_k \in \mathcal{X}$ as $\mathcal{Q}(x_k)$ is then nonempty, compact and convex. Problem (16) follows the structure of (5) where $V_N(x_k, \cdot)$ represents the smooth part and $\mathcal{I}_{\mathcal{Q}(x_k)}(\cdot)$ the non-smooth part of the objective function.

This work proposes an MPC scheme that implements a real-time version of the PGM to solve optimal control problem (16) in receding horizon. This means that only one PGM step (12) is performed per control update. The first sample of the resulting input trajectory is applied to the system (1) and a warm-start for the next update is computed. This is achieved by shifting the input trajectory over one sample and expanding its last sample using the control law $\kappa_f(\cdot)$ applied to the final state. The closed-loop dynamics are summarized as:

$$\begin{aligned} q_k^+ &= \Pi_{\mathcal{Q}(x_k)}(q_k - \gamma \nabla_q V_N(x_k, q_k)), \\ x_{k+1} &= Ax_k + Bq_k^+(0), \\ q_{k+1} &= \{q_k^+(1), \dots, q_k^+(N-1), \kappa_f(\phi(N, x_k, q_k^+))\}. \end{aligned} \quad (17)$$

This closed-loop system has x_k and q_k as states.

B. Asymptotic stability of the real-time PGM

This subsection proves asymptotic stability of the closed-loop system (17).

Theorem 2: Under Assumption 1, the set $\mathcal{X} \times \mathcal{Q}(\mathcal{X})$ is positive invariant for the closed-loop system (17) and the function $h(x, q)$ is a Lyapunov function for (17), satisfying for all $(x, q) \in \mathcal{X} \times \mathcal{Q}(\mathcal{X})$

$$\begin{aligned} h(x_{k+1}, q_{k+1}) - h(x_k, q_k) &\leq -\alpha_1(\|(x_k, q_k - q^*(x_k))\|), \\ \alpha_2(\|(x, q - q^*(x))\|) &\leq h(x, q), \\ \alpha_3(\|(x, q - q^*(x))\|) &\geq h(x, q), \end{aligned}$$

in which α_1 is a positive definite function (i.e. it is continuous, zero at zero, and positive everywhere else) and α_2, α_3 are \mathcal{K}_∞ functions (i.e. they are continuous, strictly increasing, zero at zero and unbounded). Therefore it follows that $(0, q^*(0))$ is asymptotically stable for states (x, q) of system (17) with a region of attraction $\mathcal{X} \times \mathcal{Q}(\mathcal{X})$.

Proof: Because $\mathcal{Q}(x)$ is invariant under the projection $\Pi_{\mathcal{Q}(x)}$ and because $\forall x \in \mathbb{X}_f, \exists \kappa_f(x) \in \mathbb{Q}$ such that (3a) holds, the set $\mathcal{X} \times \mathcal{Q}(x)$ is positive invariant for (17).

A decrease in h is shown as follows:

$$\begin{aligned} h(x_{k+1}, q_{k+1}) &\leq h(x_k, q_k^+) - l(x_k, q_k^+(0)), \\ &\leq h(x_k, q_k) - l(x_k, q_k^+(0)) \\ &\quad - \beta \|q_k - q^*(x_k)\|_2^2, \\ &\triangleq h(x_k, q_k) - \alpha_1(\|(x_k, q_k - q^*(x_k))\|), \end{aligned}$$

where the first inequality follows from assumption (3b). The second inequality follows from Theorem 1. Function α_1 is positive definite as l is positive definite and $\beta > 0$.

A lower bound on h is found for all (x, q) :

$$\begin{aligned} h(x, q) &\geq V_N(x, q^*(x)) + \frac{m}{2} \|q - q^*(x)\|_2^2, \\ &\geq l(x, q^*(x)) + \frac{m}{2} \|q - q^*(x)\|_2^2, \\ &\triangleq \alpha_2(\|(x, q - q^*(x))\|), \end{aligned}$$

where the first inequality follows from strong convexity of $V_N(x, \cdot)$, the subgradient inequality (10) and the optimality condition $-\nabla_q V_N(x, q^*(x)) \in \partial g(q^*(x))$. The second inequality follows from the fact that $V_N(x, q^*(x)) \geq l(x, q^*(x))$ [1]. The function α_2 is \mathcal{K}_∞ as l is positive definite and $m > 0$.

An upper bound on h is found for all $(x, q) \in \mathcal{X} \times \mathcal{Q}(\mathcal{X})$. In this set, the indicator $\mathcal{I}_{\mathcal{Q}(x)}$ evaluates to 0 and the upper bound is established as follows:

$$\begin{aligned} h(x, q) &= V_N(x, q) \\ &\leq V_N(x, q^*(x)) + \nabla_q V_N(x, q^*)^T (q - q^*) \\ &\quad + \frac{L}{2} \|q - q^*\|_2^2 \\ &\leq V_f(x) + \|\nabla_q V_N(x, q^*)\|_2 \|q - q^*\|_2 + \frac{L}{2} \|q - q^*\|_2^2 \\ &\triangleq \alpha_3(\|(x, q - q^*(x))\|), \end{aligned}$$

where the first inequality follows from Lipschitz continuity of $\nabla_q V_N$. The second inequality follows from the Cauchy-Schwarz inequality and from the fact that

Algorithm 1 Real-time PGM for linear MPC with box input constraints

Require: $\gamma \in (0, \frac{2}{\lambda_{\max}(F)})$

- 1: **Repeat** every sample k
- 2: Retrieve state estimate x_k
- 3: $q_k^+ = \text{sat}_{u_{\min}}^{u_{\max}} \{(I - \gamma F)q_k - \gamma Hx_k\}$
- 4: Apply $q_k^+(0)$ to the system
- 5: $q_{k+1} = \{q_k^+(1), \dots, q_k^+(N-1), 0\}$
- 6: **End**

$V_N(x, q^*(x)) \leq V_f(x)$ [1]. Function α_3 is \mathcal{K}_∞ as V_f is positive definite and $L > 0$.

Aforementioned derivations prove that $h(x, q)$ is a Lyapunov function for the system (17) and therefore it follows that $(0, q^*(0))$ is asymptotically stable with region of attraction $\mathcal{X} \times \mathcal{Q}(\mathcal{X})$ [1]. ■

Remark 1: Theorem 1 can be extended for Q positive semi-definite, i.e. $Q \succeq 0$, using an assumption on detectability or input/output-to-state stability [1]. Positive semi-definiteness of Q does however not harm the positive definiteness of F and the strong convexity of $V_N(x, \cdot)$.

Remark 2: Theorem 1 not only shows that the system's state x is attracted to its stable equilibrium but also guarantees that the suboptimal trajectory q converges towards the optimal trajectory $q^*(x)$.

C. Practical limitation

The real-time PGM algorithm is only useful in practice when the projection $\Pi_{\mathcal{Q}(x_k)}$ is easy to compute. The inclusion of the terminal set \mathbb{X}_f in $\mathcal{Q}(x_k)$ generally complicates this requirement. Existing approaches to omit this terminal set use soft constraining [15] or require the horizon length N to be sufficiently large [16]. Other approaches dualize the terminal state constraint and use a dual optimization strategy [7], [9], [11]. All these results however rely on solving the OCPs until optimality and are therefore unusable in a real-time, suboptimal setting.

However, when restricting to stable LTI systems, assumptions (3a) and (3b) hold for a terminal set $X_f = \mathbb{R}^n$, terminal control law $\kappa_f(\cdot) = 0$ and P the solution of the Lyapunov equation $A^T P A + Q = P$. It remains to restrict \mathbb{U} to sets that allow a simple projection. Examples are the m -dimensional Euclidean ball, the positive orthant, the simplex, box, hyperplane, halfspace, the second order cone and the positive definite cone [13]. For example, when $\mathbb{U} = [u_{\min}, u_{\max}]$, the projection on $\mathcal{Q}(x_k)$ represents the saturation function

$$\text{sat}_{u_{\min}}^{u_{\max}} \{q\}(k) = \begin{cases} u_{\min}, & q(k) < u_{\min} \\ q(k), & u_{\min} \leq q(k) \leq u_{\max} \\ u_{\max}, & q(k) > u_{\max} \end{cases}$$

As an illustration, Algorithm 1 presents the real-time PGM applied to a stable LTI system with box input constraints. The resulting algorithm consists of simple steps involving matrix-vector multiplication, addition and saturation and is therefore implementation-friendly for resource-constrained embedded system.

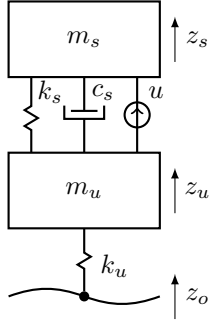


Fig. 1: Quarter-car model with an active suspension.

m_s	320 kg	k_s	18 kN/m	c_s	1 kNs/m
m_u	40 kg	k_u	200 kN/m	u_s	1.5kN

TABLE I: Numerical parameters of the quarter-car model.

V. NUMERICAL EXAMPLE

The performance of the presented real-time PGM algorithm is validated on a simulation example. The MPC strategy is used to control an active vehicle suspension system, based on a 2-DOF quarter-car model. Figure 1 presents the considered system. The quarter-car model consists of a passive suspension, represented by a stiffness k_s and damping c_s , a tire stiffness k_u , the body mass m_s and wheel mass m_u . Moreover, $z_s - z_u$ represents the suspension stroke, $z_u - z_o$ the tire deflection and z_o the vertical ground displacement caused by road unevenness. An active element such as a hydraulic actuator can generate a force u between the two masses. When choosing the states as $x = [z_s - z_u, \dot{z}_s, z_u - z_o, \dot{z}_u]^T$, the system is described by the following state-space description:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & -1 \\ -\frac{k_s}{m_s} & -\frac{c_s}{m_s} & 0 & \frac{c_s}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{k_s}{m_u} & \frac{c_s}{m_u} & -\frac{k_u}{m_u} & -\frac{c_s}{m_u} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{m_s} \\ 0 \\ -\frac{1}{m_u} \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ -1 \\ 0 \end{bmatrix} \dot{z}_o,$$

where \dot{z}_o acts as an exogenous disturbance. Numerical values for the quarter-car's parameters are adopted from [17] and are summarized in Table I. Here, u_s represents the actuator's saturation force. The system is discretized using zero-order hold with sample frequency 500 Hz.

The real-time PGM is used to generate an input sequence to steer the active element u in a proper way. When designing a vehicle suspension, the RMS body acceleration is an often used performance measure that quantifies ride comfort. Therefore the objective function is chosen to minimize $\sum_{k=0}^{N-1} \ddot{z}_{s,k}^2$, where \ddot{z}_s represents the body acceleration:

$$\ddot{z}_s = \begin{bmatrix} -\frac{k_s}{m_s} & -\frac{c_s}{m_s} & 0 & \frac{c_s}{m_s} \end{bmatrix} x + \begin{bmatrix} \frac{1}{m_s} \end{bmatrix} u.$$

As the system under control is a stable LTI system with box input constraints determined by u_s , the real-time PGM follows Algorithm 1. In all simulations the step length is chosen as $\gamma = 0.99 \frac{2}{\lambda_{\max}(F)}$.

A. Free response

In a first simulation example, the free response of the closed-loop system is examined. The initial state

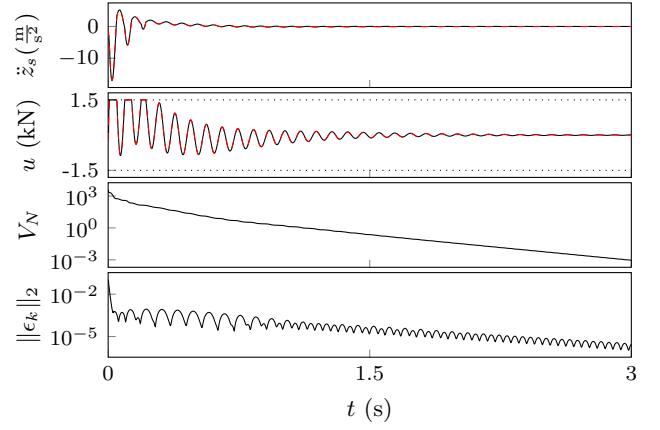


Fig. 2: Free response from $[0, 0, 0.1, 0]$ for $N = 200$. The black lines indicate the response using the real-time PGM while the dashed red lines indicate the response using an optimal MPC approach.

equals $[0, 0, 0.1, 0]^T$. The resulting body acceleration response \ddot{z}_s and control input u are depicted in Figure 2 for a horizon length $N = 200$. It is clear how the real-time PGM method generates constrained inputs u to steer the system to its equilibrium state. The RMS value of the body acceleration equals 1.6591 m/s^2 , while the RMS value of the system under an optimal MPC law is 1.6587 m/s^2 . The optimal MPC law however requires 66 PGM iterations in worst case when using a stopping criterion $\|\epsilon_k\|_2 \leq 10^{-4}$. Here, ϵ_k represents a residual for the first-order optimality condition of (16):

$$\epsilon_k = (F - \frac{1}{\gamma}I)(q_k^+ - q_k) \in \nabla_q V_N(x_k, q_k^+) + \partial g(q_k^+).$$

This expression follows from the first-order optimality conditions of (12). The \ddot{z}_s and u trajectories generated by the optimal MPC are represented in Figure 2 as dashed red lines and almost completely coincide with those of the real-time PGM. Figure 2 also presents the evolution of the value function V_N and the residual $\|\epsilon_k\|_2$ for the real-time PGM. As Theorem 2 states, V_N is monotonously decreasing over the MPC cycles. Also a decrease in optimality residual is observed which shows that the PGM steps are converging over the MPC updates.

B. Response due to road disturbances

In a second set of simulations a non-zero road disturbance z_o is applied. Similar as in [17] the response to a bump profile is examined. The bump is described by

$$z_o(t) = \begin{cases} \frac{A}{2}(1 - \cos(\frac{2\pi V}{L}t)) & 0 \leq t \leq \frac{L}{V}, \\ 0 & t > \frac{L}{V}, \end{cases}$$

where $A = 0.1 \text{ m}$ and $L = 5 \text{ m}$ represent the height and length of the bump and V is the vehicle's forward velocity.

For different values of the vehicle's forward velocity V and different horizon lengths N , the performance of the real-time PGM is compared to the performance of an optimal

$V \backslash N$	50	100	200
30 km/h	2.0e-02 (166)	2.9e-02 (157)	6.9e-02 (130)
60 km/h	2.0e-02 (189)	3.1e-02 (185)	4.0e-02 (164)
90 km/h	4.3e-03 (200)	8.3e-03 (200)	2.7e-02 (171)
120 km/h	9.8e-04 (206)	1.5e-03 (210)	6.7e-03 (181)

TABLE II: Relative deviation of the RMS body acceleration of the real-time PGM with respect to an optimal MPC approach for a bump response. The values between brackets represent the maximum iterations the optimal MPC requires to obtain $\|\epsilon_k\|_2 \leq 10^{-4}$.

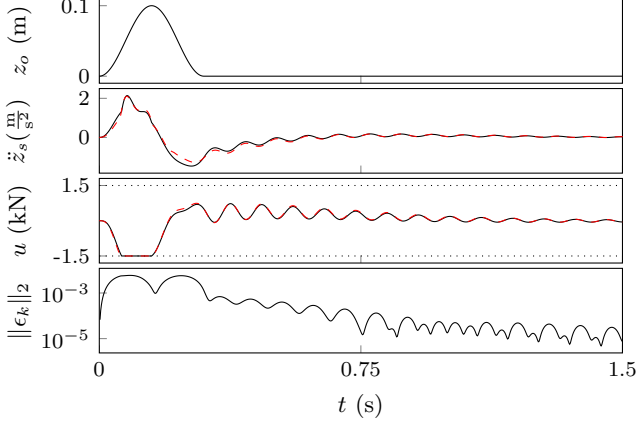


Fig. 3: Bump response for $N = 200$ and $V = 60$ km/h. The black lines indicate the response using the real-time PGM while the dashed red lines indicate the response using an optimal MPC approach.

MPC approach. The relative deviations in the RMS body acceleration are summarized in Table II as well as the maximum number of PGM iterations required for the optimal MPC to obtain $\|\epsilon_k\|_2 \leq 10^{-4}$. In general, the performance deterioration due to the suboptimal iterations is limited while the computational complexity is much lower than the optimal MPC. While the optimal MPC needs around 200 PGM iterations in worst case, the real-time PGM only executes one iteration per control update. The difference in performance is more pronounced for large N as a longer horizon allows more opportunities to optimize the system's behavior. However, when N increases, the computational complexity of an optimal MPC can get too large to be used at fast rates. In these cases, the real-time PGM could still be preferred. As an illustration, Figure 3 presents the response to a bump road disturbance respectively for $N = 200$ and $V = 60$ km/h. In dashed red lines the response due to an optimal MPC approach is presented. Again, these trajectories are very close to the real-time PGM trajectories.

In general one could conclude from this simulation example that the real-time PGM forms a computational cheap alternative for an optimal MPC approach with little loss in performance.

VI. CONCLUSION

This paper presents a real-time model predictive control strategy. Each MPC cycle one step of the proximal gradient

method is performed, while a next MPC update is warm-started using the solution of the previous update. When applied to LTI systems with input constraints, the resulting closed-loop system is shown to be asymptotically stable. When the input constraint set allows an easy projection, the real-time PGM algorithm becomes extremely simple and can be executed fastly on embedded hardware. The closed-loop performance of the real-time PGM is validated in a simulation example where it is used to control an active vehicle suspension. These results show limited deterioration in performance when comparing the real-time PGM method with an optimal MPC approach. Compared to optimal MPC, the real-time PGM however benefits from simple updates with fixed computational complexity.

REFERENCES

- [1] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Pub., 2009.
- [2] H. Michalska and D. Q. Mayne, "Robust receding horizon control of constrained nonlinear systems," *IEEE Transactions on automatic control*, vol. 38, no. 11, pp. 1623–1633, 1993.
- [3] P. O. Scokaert, D. Q. Mayne, and J. B. Rawlings, "Suboptimal model predictive control (feasibility implies stability)," *IEEE Transactions on Automatic Control*, vol. 44, no. 3, pp. 648–654, 1999.
- [4] M. Diehl, H. G. Bock, and J. P. Schlöder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM Journal on control and optimization*, vol. 43, no. 5, pp. 1714–1736, 2005.
- [5] M. Diehl, R. Findeisen, F. Allgöwer, H. G. Bock, and J. P. Schlöder, "Nominal stability of real-time iteration scheme for nonlinear model predictive control," *IEEE Proceedings-Control Theory and Applications*, vol. 152, no. 3, pp. 296–308, 2005.
- [6] G. Stathopoulos, H. Shukla, A. Szucs, Y. Pu, C. N. Jones, et al., "Operator splitting methods in control," *Foundations and Trends® in Systems and Control*, vol. 3, no. 3, pp. 249–362, 2016.
- [7] J. L. Jerez, P. J. Goulart, S. Richter, G. A. Constantinides, E. C. Kerrigan, and M. Morari, "Embedded online optimization for model predictive control at megahertz rates," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3238–3251, 2014.
- [8] M. Kögel and R. Findeisen, "Parallel solution of model predictive control using the alternating direction multiplier method," *IFAC Proceedings Volumes*, vol. 45, no. 17, pp. 369–374, 2012.
- [9] S. Richter, M. Morari, and C. N. Jones, "Towards computational complexity certification for constrained MPC based on lagrange relaxation and the fast gradient method," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 5223–5229, IEEE, 2011.
- [10] S. Richter, C. N. Jones, and M. Morari, "Computational complexity certification for real-time MPC with input constraints based on the fast gradient method," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1391–1403, 2012.
- [11] 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, 2014.
- [12] V. Nedelcu, I. Necoara, and Q. Tran-Dinh, "Computational complexity of inexact gradient augmented lagrangian methods: application to constrained MPC," *SIAM Journal on Control and Optimization*, vol. 52, no. 5, pp. 3109–3134, 2014.
- [13] N. Parikh, S. Boyd, et al., "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.
- [14] Y. Nesterov, "Gradient methods for minimizing composite functions," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, 2013.
- [15] M. N. Zeilinger, M. Morari, and C. N. Jones, "Soft constrained model predictive control with robust stability guarantees," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1190–1202, 2014.
- [16] D. Limón, T. Alamo, F. Salas, and E. F. Camacho, "On the stability of constrained MPC without terminal constraint," *IEEE transactions on automatic control*, vol. 51, no. 5, pp. 832–836, 2006.
- [17] H. Chen, P.-Y. Sun, and K.-H. Guo, "Constrained H-infinity control of active suspensions: an LMI approach," in *Control and Automation, 2002. ICCA. Final Program and Book of Abstracts. The 2002 International Conference on*, pp. 157–157, IEEE, 2002.