Policy Optimization for Unknown Systems using Differentiable Model Predictive Control

Riccardo Zuliani RZULIANI@ETHZ.CH
Automatic Control Laboratory, ETH Zürich

Efe C. Balta EFE.BALTA@INSPIRE.CH
Inspire AG, Zürich and Automatic Control Laboratory, ETH Zürich

John Lygeros JLYGEROS@ETHZ.CH
Automatic Control Laboratory, ETH Zürich

Editors: G. Sukhatme, L. Lindemann, S. Tu, A. Wierman, N. Atanasov

Abstract

Model-based policy optimization often struggles with inaccurate system dynamics models, leading to suboptimal closed-loop performance. This challenge is especially evident in Model Predictive Control (MPC) policies, which rely on the model for real-time trajectory planning and optimization. We introduce a novel policy optimization framework for MPC-based policies combining differentiable optimization with zeroth-order optimization. Our method combines model-based and model-free gradient estimation approaches, achieving faster transient performance compared to fully data-driven approaches while maintaining convergence guarantees, even under model uncertainty. We demonstrate the effectiveness of the proposed approach on a nonlinear control task involving a 12-dimensional quadcopter model.

Keywords: Model Predictive Control, Policy Optimization, Zeroth-order Optimization.

1. Introduction

Policy optimization is the problem of designing a control policy that minimizes a prescribed performance objective, typically by searching over a parameterized policy class (Sutton et al., 2002). A growing line of work studies policy optimization problems where the policy is a model predictive controller (MPC) (Amos et al., 2018; Gros and Zanon, 2019; Agrawal et al., 2020; Drgoňa et al., 2022; Zuliani et al., 2025b,c). MPC-based policies generate predictions of future state trajectories using a system model and naturally incorporate constraints into their decision-making process, offering stronger safety guarantees and greater interpretability compared to model-free approaches.

Existing MPC-based policy optimization schemes typically assume that the dynamics model is exact, and convergence results rely on this assumption (Zuliani et al., 2025b); obtaining convergence guarantees when the model is inaccurate remains an open problem. In this paper, we propose a novel policy optimization algorithm with convergence guarantees inspired by He et al. (2024) blending model-based and zeroth-order gradient information that is robust to inexact models. A key feature of our method is that it can smoothly trade off between model-based and zeroth-order components, putting more weight on the model whenever it is trusted and relying more on model-free information otherwise. To handle the nonsmoothness of MPC policies, we leverage the tools of Bolte and Pauwels (2021). We validate our approach on a 12-dimensional nonlinear quadcopter.

This work was supported as a part of NCCR Automation, a National Centre of Competence in Research, funded by the Swiss National Science Foundation (grant number 51NF40_225155).

Related work: Zeroth-order optimization addresses the problem of minimizing an objective function when exact gradient information is unavailable. The foundations of the approach used in this paper trace back to the seminal work of Flaxman et al. (2004), which introduced a smoothing-based approximation technique enabling gradient-free optimization for possibly nonsmooth functions. Subsequent research extended these ideas to convex settings: see Duchi et al. (2012) and Nesterov and Spokoiny (2017) for one-point gradient estimators, or Shamir (2017) for a two-point estimator. More recently, Lin et al. (2022) generalized the two-point approach to nonconvex problems, demonstrating its effectiveness beyond the convex regime. Closest to our work is He et al. (2024), which combines a one-point zeroth-order estimator with model-based gradient information to improve convergence speed, a direction we further build upon in this paper.

Notation: $\mathcal{X}^n$ denotes the n-fold Cartesian product of the set $\mathcal{X}$. Given a path-differentiable function g of two arguments x and y, $\mathcal{J}_{g,x}$ and $\mathcal{J}_{g,y}$ denote the projection of the conservative Jacobian $\mathcal{J}_g$ (defined in Section 2.1) onto the x and y entries. $\mathbb{B}$ and $\mathbb{S}$ are the unit ball and sphere in the Euclidean norm. $U(\mathbb{B})$ and $U(\mathbb{S})$ denote uniform distributions over $\mathbb{B}$ and $\mathbb{S}$. $\mathcal{N}_{\mathcal{X}}$ is the Clarke tangent cone of the set $\mathcal{X}$. $\mathrm{dist}_p(a, B)$ is the distance between point a and set B in the p-norm. We use $\mathbf{0}_n, \mathbf{1}_n \in \mathbb{R}^n$ to denote the vector of zeros and ones, respectively.

Outline: We graphically outline the rest of this paper graphically in Figure 1.

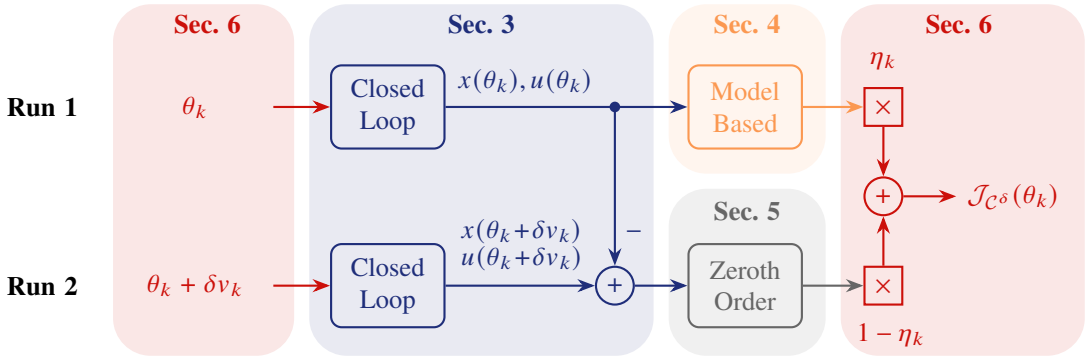

Figure 1: Graphical outline of the paper.

2. Preliminaries

Since MPC-based policy optimization is a fundamentally nonsmooth problem, we adopt the framework of definable functions to ensure existence of well-defined generalized Jacobians. We formally define the generalized Jacobian in Section A and report the definition of definability in Appendix 2.1. The derivation of generalized Jacobians for MPC solution maps is reported in Appendix B. Finally, to account for both the nonsmooth landscape and unknown dynamics, we characterize the convergence of our algorithm using generalized critical points, defined in Section 2.2.

2.1. Conservative Jacobians

The notion of Conservative Jacobians, introduced in Bolte and Pauwels (2021), extends the concept of derivatives to locally Lipschitz, almost everywhere differentiable functions.

Definition 1 (Bolte and Pauwels (2021)) *The map $\mathcal{J}_\phi : \mathbb{R}^n \rightrightarrows \mathbb{R}^{m\times n}$ is a conservative Jacobian of the locally Lipschitz function $\phi : \mathbb{R}^n \to \mathbb{R}^m$ if it is nonempty-valued, outer semicontinuous, locally*

bounded, and for all absolutely continuous curves $x : [0,1] \to \mathbb{R}^n$ *and almost all* $t \in [0,1]$

$$\frac{d}{dt}\phi(x(t)) = \langle v, \dot{x}(t)\rangle, \ \forall v \in \mathcal{J}_\phi(x(t)). \tag{1}$$

Excluding a set of measure zero, $\mathcal{J}_\phi$ coincides almost everywhere with the gradient $\nabla\phi$. Unlike other nonsmooth Jacobians, such as Clarke Jacobians, conservative Jacobians obey the chain rule of differentiation and a nonsmooth version of the implicit function theorem (Bolte et al., 2021), making them particularly suitable for the sensitivity analysis of solution maps of optimization problems, which are typically nonsmooth functions implicitly defined by optimality conditions.

A function admitting a conservative Jacobian is called *path-differentiable*. As demonstrated in Lemma 3, Bolte et al. (2021), locally Lipschitz definable functions are always path-differentiable.

2.2. Stationary points in nonsmooth optimization

The *Clarke Jacobian* of a locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$ is the outer-semicontinuous map

$$\partial_c f(x) := \operatorname{co}\{g \in \mathbb{R}^n : g = \lim_{y \to x} \nabla f(y), \ y \in D_f\},$$

where $D_f \subset \mathbb{R}^n$ is the full-measure set on which f is differentiable, and co denotes the convex hull. There is a close connection between Clarke and conservative Jacobians (Bolte and Pauwels, 2021), where $\partial_c f(x) \subseteq \operatorname{co} \mathcal{J}_f(x)$ for all x, and $\partial_c f(x) = \mathcal{J}_f(x)$ for almost every x.

If x is a local minimizer of f, then $0 \in \partial_c f(x)$ (and similarly $0 \in \operatorname{co} \mathcal{J}_f(x)$). Hence, in nonsmooth optimization, one typically searches for *Clarke stationary points*, that is, points x with $0 \in \partial_c f(x)$. In our setting, identifying Clarke stationary points is impossible to due lack of full information regarding the objective function. Consequently, we adopt a weaker notion of stationarity, namely, that of a *Goldstein δ-critical point*, defined as any x such that $0 \in \partial_\delta f(x)$, where

$$\partial_\delta f(x) := \operatorname{co}\left\{\bigcup_{y \in \delta\mathbb{B}} \partial_c f(x+y)\right\}$$

is the Goldstein δ-subdifferential of f. As shown in Zhang et al. (2020), $\lim_{\delta \downarrow 0} \partial_\delta f(x) = \partial_c f(x)$, making Goldstein δ-stationarity a meaningful optimality condition for nonsmooth problems.

For constrained problems, such as minimizing $f(x)$ subject to $h(x) = 0$, $x \in \mathcal{X}$, we consider a generalized stationarity concept adapted from Grimmer and Jia (2025). A point x is said to satisfy the *Goldstein Fritz-John δ-critical condition* if there exist multipliers $\lambda_0, \lambda_1 \geq 0$ such that $0 \in \lambda_0 \partial_\delta f(x) + \lambda_1 \partial_\delta h(x) + \mathcal{N}_\mathcal{X}(x)$ and $\lambda_0 + \lambda_1 = 1$. This condition parallels the classical Fritz-John optimality conditions, with the gradients replaced by Goldstein δ-subdifferentials. If $\lambda_0 > 0$, the point is called a *Goldstein KKT δ-critical point*.

3. Problem Formulation

Consider an unknown system evolving over $t \in \mathbb{Z}_{[0,T]}$ from an initial condition $x_0 \in \mathbb{R}^{n_x}$

$$x_{t+1} = f(x_t, u_t), \tag{2a}$$
$$u_t = \mathrm{MPC}(x_t, \theta), \tag{2b}$$

where $\theta \in \Theta$ is a tunable parameter that determines the behavior of the policy and $\Theta \subset \mathbb{R}^{n_\theta}$ is a parameter set. Despite not knowing the dynamics in (2a) exactly, we assume existence of a model

$g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ such that $g(x,u) \approx f(x,u)$ for all x and u. For example, g may be a linear model obtained via system identification of the nonlinear plant f, or a first-principles model that neglects higher-order effects. We consider the constraints

$$x_t \in \mathcal{X}, \quad u_t \in \mathcal{U}, \tag{3}$$

where $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ and $\mathcal{U} \subseteq \mathbb{R}^{n_u}$ are known convex sets. The input $u_t = \text{MPC}(x_t, \theta)$ depends on the state x_t (last constraint in (2b)) and the parameters θ, and it is computed by solving

$$\begin{aligned} \underset{x_{\cdot|t}, u_{\cdot|t}}{\text{minimize}} \quad & \ell_{N,\theta}(x_{N|t}) + \sum_{k=0}^{N-1} \ell_\theta(x_{k|t}, u_{k|t}) \\ \text{subject to} \quad & x_{k+1|t} = g(x_{k|t}, u_{k|t}), \quad x_{0|t} = x_t, \quad k \in \mathbb{Z}_{[0,N-1]}, \\ & x_{k|t} \in \mathcal{X}, \quad k \in \mathbb{Z}_{[0,N]}, \\ & u_{k|t} \in \mathcal{U}, \quad k \in \mathbb{Z}_{[0,N-1]}, \end{aligned} \tag{4}$$

where ℓ_θ and $\ell_{N,\theta}$ are parameterized cost functions, and setting $u_t = u_{0|t}$. Throughout, we assume that (4) is a quadratic program (QP) meeting the conditions of Theorem 10 in Appendix B. If g is a nonlinear function, one can use the linearization techniques in Section VI-A of Zuliani et al. (2025b), to obtain an MPC that can be expressed as a quadratic program.

Our emphasis is on QP-based MPC policies, which often deliver strong performance even on nonlinear control problems. Nonetheless, the framework can easily be extended to fully nonlinear policies using the differentiation methods in Zuliani et al. (2025a). Similarly, this method can accommodate nonconvex upper-level constraints in (3) as long as the MPC satisfies Assumption 1. Note additionally that while in this work we restrict θ to parameters appearing in the cost of (2b), extending it to also affect the constraints or dynamics of the MPC is straightforward.

Our goal is to obtain an MPC design θ^* that minimizes a known objective function $C(x,u)$, where $x = (x_0, \ldots, x_T)$ and $u = (u_0, \ldots, u_{T-1})$, typically with $T \gg N$, are the closed-loop trajectories obtained by combining (2a) and (2b), while satisfying (3) for all t.

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & C(x,u) = C(x_0, \ldots, x_T, u_0, \ldots, u_{T-1}) \\ \text{subject to} \quad & x_{t+1} = f(x_t, u_t), \quad x_0 \text{ given}, \quad t \in \mathbb{Z}_{[0,T-1]}, \\ & u_t = \text{MPC}(x_t, \theta), \quad t \in \mathbb{Z}_{[0,T-1]}, \\ & x_t \in \mathcal{X}, \quad t \in \mathbb{Z}_{[0,T]}, \\ & u_t \in \mathcal{U}, \quad t \in \mathbb{Z}_{[0,T-1]}. \end{aligned} \tag{5}$$

Since f is unknown, problem (5) cannot be solved directly. Our approach combines model-based gradient estimation, leveraging the available model g, with zeroth-order techniques.

4. Model-based Gradient Estimation

4.1. Solving the problem when dynamics are known

Even with perfect model knowledge, (5) is difficult to solve due to the nonsmooth constraints imposed by the MPC function. Zuliani et al. (2025b) introduced a gradient-based method specifically designed for such problems, which solves an unconstrained reformulation of (5). To cast (5) as an

unconstrained problem, let $x : \Theta \to \mathbb{R}^{(T+1)n_x}$ and $u : \Theta \to \mathbb{R}^{Tn_u}$ be the closed-loop trajectories obtained by combining (2b) and (2a) over the entire horizon $t \in \mathbb{Z}_{[0,T]}$. Then (5) becomes

$$\begin{aligned} \underset{\theta \in \Theta}{\text{minimize}} \quad & C(x(\theta), u(\theta)) \\ \text{subject to} \quad & x(\theta) \in \mathcal{X}^{T+1}, \;\; u(\theta) \in \mathcal{U}^T. \end{aligned} \tag{6}$$

Since (4) enforces $u_t(\theta) \in \mathcal{U}$ for all $t \in \mathbb{Z}_{[0,T-1]}$, the input constraints in (6) can be dropped. The state constraints can be incorporated in the cost through a continuous penalty function $P(\theta) = \bar{P}(x(\theta))$ satisfying $P(\theta) = 0$ whenever $x(\theta) \in \mathcal{X}^{T+1}$ and $P(\theta) > 0$ otherwise

$$\underset{\theta \in \Theta}{\text{minimize}} \;\; C(x(\theta), u(\theta)) + P(\theta) =: \mathcal{C}(\theta). \tag{7}$$

Under appropriate assumptions, which we detail in Section 6.1, $\mathcal{C}$ is path-differentiable and the update law $\theta_{k+1} = \Pi_\Theta[\theta_k - \alpha_k d_k]$, where $d_k \in \mathcal{J}_\mathcal{C}(\theta_k)$, and $\Pi_\Theta : \mathbb{R}^{n_\theta} \to \Theta$ is the projector onto the set Θ, converges to a minimizer of (6).

4.2. Imperfect Gradient Information using an Inexact Model

Since (2a) is unknown, the exact Jacobian $\mathcal{J}_\mathcal{C}$ cannot be computed, and the exact update cannot be applied directly. Instead, we approximate $\mathcal{J}_\mathcal{C}$ using the available model g of the true dynamics f. First, we recursively build approximations $J_x(\theta)$ and $J_u(\theta)$ of $\mathcal{J}_x(\theta)$ and $\mathcal{J}_u(\theta)$ via

$$J_{x_{t+1}}(\theta) = J_{g,x}(x_t, u_t) J_{x_t}(\theta) + J_{g,u}(x_t, u_t) J_{u_t}(\theta), \tag{8a}$$

$$J_{u_t}(\theta) = J_{\text{MPC},x}(x_t, \theta) J_{x_t}(\theta) + J_{\text{MPC},\theta}(x_t, \theta), \tag{8b}$$

where $x_t = x_t(\theta)$, $u_t = u_t(\theta)$, and $J_g(x, u)$ and $J_{\text{MPC}}(x, \theta)$ are elements of the conservative Jacobians $\mathcal{J}_g(x, u)$ and $\mathcal{J}_{\text{MPC}}(x, \theta)$, respectively. We then combine $J_x(\theta)$ and $J_u(\theta)$ to obtain an estimate $J_\mathcal{C}(\theta)$ of $\mathcal{J}_\mathcal{C}(\theta)$ using the chain rule

$$J_\mathcal{C}(\theta) = J_{C,x}(x, u) J_x(\theta) + J_{C,u}(x, u) J_u(\theta), \;\; J_C(x, u) \in \mathcal{J}_C(x, u). \tag{8c}$$

Replacing $\mathcal{J}_\mathcal{C}(\theta)$ with $J_\mathcal{C}(\theta)$ yields the implementable update $\theta_{k+1} = \Pi_\Theta[\theta_k - \alpha_k J_\mathcal{C}(\theta_k)]$.

5. Model-free Gradient Estimation

The update of Section 4.2 is appealing because it only relies on the approximate model. However, without further assumptions on the model accuracy, it is impossible to derive convergence guarantees. To address this issue, we construct gradient-like directions purely from data, for which convergence can be established under mild conditions.

For any $\delta > 0$, consider the *randomized smoothing approximation* $\mathcal{C}^\delta(\theta) : \Theta_\delta \to \mathbb{R}$ of $\mathcal{C}$

$$\mathcal{C}^\delta(\theta) = \mathbb{E}_{w \sim U(\mathbb{B})}[\mathcal{C}(\theta + \delta w)],$$

where $\Theta_\delta := \Theta + \delta\mathbb{B}$. $\mathcal{C}^\delta$ is a smooth approximation of $\mathcal{C}$: as long as $\mathcal{C}$ is Lipschitz with constant $L_\mathcal{C}$, $\mathcal{C}^\delta$ is continuously differentiable and Lipschitz with constant $L_{\mathcal{C}^\delta} = c\sqrt{n_\theta}\delta^{-1} L_\mathcal{C}$ for some $c > 0$ (Lin et al., 2022, Proposition 2.3). Working with $\mathcal{C}^\delta$ instead of $\mathcal{C}$ has two key benefits: (i)

the Lipschitz continuity of $\mathcal{C}$ implies smoothness of $\mathcal{C}^\delta$, and (ii) one can estimate $\nabla\mathcal{C}^\delta$ solely using function evaluations. For this, we use the following one-point estimator

$$J_{\mathcal{C}^\delta}(\theta_k, v_k) = \frac{n_\theta}{\delta}[\mathcal{C}(\theta_k + \delta v_k) - \mathcal{C}(\theta_k)]v_k, \tag{9}$$

where v_k is sampled i.i.d. from $U(\mathbb{S})$ for each $k \in \mathbb{N}$. We have the following.

Lemma 2 *For any $\theta \in \Theta$, $\mathbb{E}_v[J_{\mathcal{C}^\delta}(\theta, v)] = \nabla\mathcal{C}^\delta(\theta)$.*

Proof By Lemma 1 in Flaxman et al. (2004), it holds that $\mathbb{E}_v[\mathcal{C}(\theta + \delta v)v] = \delta/n_\theta \nabla\mathcal{C}^\delta(\theta)$. Since v is zero-mean, we immediately have $\mathbb{E}_v[(\mathcal{C}(\theta + \delta v) - \mathcal{C}(\theta))v] = \mathbb{E}_v[\mathcal{C}(\theta + \delta v)v] = \delta/n_\theta \nabla\mathcal{C}^\delta(\theta)$. ■

Lemma 2 shows that the zeroth-order estimator (9) provides an unbiased estimate of $\nabla\mathcal{C}^\delta$. To relate this to the original nonsmooth objective, we recall the following result from Lin et al. (2022).

Lemma 3 *Suppose $\mathcal{C}$ is $L_\mathcal{C}$-Lipschitz in Θ. Then $\nabla\mathcal{C}^\delta(\theta) \in \partial_\delta\mathcal{C}(\theta)$ for any $\theta \in \Theta$.*

This means that estimating $\nabla\mathcal{C}^\delta$ yields an element of the Goldstein δ-subdifferential of the true objective $\mathcal{C}$, thus providing a second implementable update law $\theta_{k+1} = \Pi_\Theta[\theta_k - \alpha_k J_{\mathcal{C}^\delta}(\theta_k, v_k)]$.

6. Proposed Algorithm

Our algorithm combines the model-based update of Section 4.2 with the model-free update of Section 5, following the *gray-box* scheme of He et al. (2024). At iteration k, we update θ^k as follows

$$\begin{aligned} d_{k,1} &= J_\mathcal{C}(\theta_k), \\ d_{k,2} &= J_{\mathcal{C}^\delta}(\theta_k, v_k),\ v_k \sim U(\mathbb{S}), \\ d_k &= \eta_k d_{k,1} + (1 - \eta_k) d_{k,2}, \\ \theta_{k+1} &= \Pi_\Theta[\theta_k - \alpha_k d_k], \end{aligned} \tag{10}$$

where $\{\alpha_k\}_{k\in\mathbb{N}} \subset \mathbb{R}_{>0}$ is a sequence of vanishing stepsizes, and $\{\eta_k\}_{k\in\mathbb{N}} \subset [0, 1]$ weights the relative contribution of the two update directions. Choosing $\eta_k \approx 1$ prioritizes the model-based direction $J_\mathcal{C}(\theta_k)$, whereas $\eta_k \approx 0$ makes the update closer to zeroth order via $J_{\mathcal{C}^\delta}(\theta_k, v_k)$. Generally, selecting η_k is a design choice that should reflect the trustworthiness of the model. We showcase how the converge speed is affected by different choices of η_k in simulation in Section 7.

6.1. Convergence to a Goldstein δ-Critical Point

Our convergence analysis builds on Lemma 2 and standard results on stochastic projected gradient methods for nonsmooth, nonconvex objectives (Davis et al., 2020). We first impose regularity assumptions ensuring that the closed-loop map and the objective are locally Lipschitz and definable.

Assumption 1 *The true dynamics f, the model g, the cost function C, the penalty P, and the MPC function* MPC : $\mathbb{R}^{n_x} \times \Theta \to \mathbb{R}^{n_u}$ *are locally Lipschitz and definable in an o-minimal structure.*

Under Assumption 1, the closed-loop trajectories $x(\theta)$ and $u(\theta)$ are locally Lipschitz and definable in θ, allowing the definition of conservative Jacobians. Assumption 1 is not restrictive, as definable functions cover almost all functions of interest in control and optimization, and the MPC function satisfies Assumption 1 under the mild conditions laid out in Appendix B. To ensure feasibility of the MPC, one can resort to the technique in Section VI-D of Zuliani et al. (2025b).

Our last technical requirement simplifies the analysis by ensuring boundedness of the gradients.

Assumption 2 *The set Θ is convex, compact and definable in an o-minimal structure.*

To ensure convergence, we require the following conditions on α_k and η_k

$$\alpha_k > 0, \qquad \sum_{k\in\mathbb{N}}\alpha_k = +\infty, \;\; \sum_{k\in\mathbb{N}}\alpha_k^2 < +\infty, \tag{11a}$$

$$\eta_k \in [0,1], \qquad \sum_{k\in\mathbb{N}}\eta_k\alpha_k < +\infty. \tag{11b}$$

These conditions hold, for example, if $\alpha_k = 1/(k+1)^\gamma$ and $\eta_k = 1/(k+1)^{\beta-\gamma}$, with $\gamma \in (0.5, 1]$ and $\beta > 1$. This allows for a wide range of possible stepsizes and, crucially, for different decrease rates for η_k. This last feature, in particular, allows us to accomodate situations where the model $J_{\mathcal{C}}$ is deemed trustworthy, and thus η_k should have larger values, but also situations where $J_{\mathcal{C}}$ is less trusted and the zeroth-order estimation is preferred. Observe that $\eta_k \downarrow 0$, meaning that eventually the information obtained using the model is discarded and the algorithm relies solely on data.

Theorem 4 *Under Assumptions 1, and 2, if α_k and η_k satisfy (11), then θ_k as obtained through (10) converges to a Goldstein Fritz-John δ-critical point of the problem*

$$\underset{\theta\in\Theta}{\text{minimize}} \quad \mathcal{C}(\theta) \quad \textit{subject to} \;\; P(\theta) = 0. \tag{12}$$

The proof of Theorem 4 requires several preliminary results. First, we prove that under Assumption 1 the approximation $\mathcal{C}^\delta$ of $\mathcal{C}$ retains definability and Lipschitz continuity.

Lemma 5 *Under Assumptions 1 and 2, $\mathcal{C}^\delta$ is Lipschitz continuous and definable.*

Proof Under Assumptions 1 and 2, the functions $x(\theta)$ and $u(\theta)$ are definable and locally Lipschitz since both these properties are preserved by composition (Coste, 1999, Exercise 1.11). Next, the function $y \mapsto \max\{y_1, y_2\}$ is the pointwise maximum of two linear functions, and it is therefore locally Lipschitz and definable (it is, in fact, semialgebraic). This proves that $\mathcal{C}$ is locally Lipschitz and definable. Since integration (and therefore expectation) preserves definability (Speissegger, 1999), $\mathcal{C}^\delta$ is definable for every $\delta > 0$. Moreover, since $\mathcal{C}$ is locally Lipschitz and therefore Lipschitz if restricted to Θ, by (Lin et al., 2022, Proposition 2.3), $\mathcal{C}^\delta$ is Lipschitz for any $\delta > 0$. ■

Next, we show that the zeroth-order update dominates the model-based one for all k large enough.

Lemma 6 *Under Assumptions 1 and 2, if α_k and η_k satisfy (11), then $\sum_{k\in\mathbb{N}}\alpha_k\eta_k\|J_{\mathcal{C}}(\theta_k)\| < +\infty$.*

Proof Since $\sum_{k\in\mathbb{N}}\alpha_k\eta_k < +\infty$ by (11b), it suffices to prove that $\|J_{\mathcal{C}}(\theta_k)\|$ is bounded for all θ_k. To prove this, observe that $\|x(\theta)\| \le C_x$ and $\|u(\theta)\| \le C_u$ for all $\theta \in \Theta$ for some (unknown) $C_x, C_u < +\infty$ since both $x(\theta)$ and $u(\theta)$ are locally Lipschitz and Θ is compact by Assumption 2. Since J_g in (8) is an element of the conservative Jacobian of the locally Lipschitz definable function g, its value is almost surely equal to ∇g, and it is therefore almost surely bounded above by the Lipschitz constant of g on Θ. The same goes for J_{MPC}. Since J_x and J_u are constructed through the recursion (8) involving bounded quantities, and the horizon T of the problem is finite, J_{x_t} and J_{u_t} are bounded for all t, and therefore so are J_x and J_u. Finally, since C is Lipschitz on the set $C_x\mathbb{B} \times C_u\mathbb{B}$, $J_{\mathcal{C}}$ is bounded. This completes the proof. ■

Our final technical result is about proving the finiteness of the variance of $J_{\mathcal{C}^\delta}$ for each k.

Lemma 7 *Under Assumptions 1 and 2, we have for all $k \in \mathbb{N}$ that*

$$\mathbb{E}_{v_k}[J_{\mathcal{C}^\delta}(\theta_k, v_k) - \nabla\mathcal{C}^\delta(\theta_k)] = 0, \;\; \mathbb{E}_{v_k}[\|J_{\mathcal{C}^\delta}(\theta_k, v_k) - \nabla\mathcal{C}^\delta(\theta_k)\|^2] < +\infty.$$

Proof The first equation is trivial since $\mathbb{E}[J_{\mathcal{C}^\delta}(\theta_k, v_k)] = \nabla\mathcal{C}^\delta(\theta_k)$ by Lemma 2. Next, we have

$$\begin{aligned}
\mathbb{E}_{v_k}[\|J_{\mathcal{C}^\delta}(\theta_k, v_k) - \nabla\mathcal{C}^\delta(\theta_k)\|^2] &= -\|\nabla\mathcal{C}^\delta(\theta_k)\|^2 + \mathbb{E}_{v_k}[\|J_{\mathcal{C}^\delta}(\theta_k, v_k)\|^2] \\
&= -\|\nabla\mathcal{C}^\delta(\theta_k)\|^2 + \mathbb{E}_{v_k}\left[\frac{n_\theta^2\|v_k\|^2}{\delta^2}(\mathcal{C}(\theta_k + \delta v_k) - \mathcal{C}(\theta_k))^2\right] \\
&\leq -\|\nabla\mathcal{C}^\delta(\theta_k)\|^2 + \frac{n_\theta^2 L_\mathcal{C}}{\delta^2}\mathbb{E}_{v_k}[\|v_k\|^2\|\theta_k + \delta v_k - \theta_k\|^2], \\
&\leq -\|\nabla\mathcal{C}^\delta(\theta_k)\|^2 + n_\theta^2 L_\mathcal{C}\mathbb{E}_{v_k}[\|v_k\|^4],
\end{aligned}$$

Since $v_k \sim U(\mathbb{S})$, the term on the right is always finite for finite θ_k. Combining this with the continuity of $\|\nabla\mathcal{C}^\delta(\cdot)\|$, we conclude the existence of a function $p : \Theta \to \mathbb{R}_{>0}$ bounded on bounded sets such that $\mathbb{E}_{v_k}[\|J_{\mathcal{C}^\delta}(\theta_k) - \nabla\mathcal{C}^\delta(\theta_k)\|^2] \leq p(\theta_k)$, concluding the proof. ■

Proof of Theorem 4. We follow Section A in Davis et al. (2020). Note that since $\mathcal{C}^\delta$ is continuously differentiable, we can take its gradient $\nabla\mathcal{C}^\delta$ as a conservative field. First, observe that

$$\theta_{k+1} = \Pi_\Theta[\theta_k - \alpha_k\nabla\mathcal{C}^\delta(\theta_k)] + \alpha_k\xi_k \tag{13}$$

where $\alpha_k\xi_k = \Pi_\Theta[\theta_k - \alpha_k(1-\eta_k)J_{\mathcal{C}^\delta}(\theta_k, v_k) - \alpha_k\eta_k J_\mathcal{C}(\theta_k)] - \Pi_\Theta[\theta_k - \alpha_k\nabla\mathcal{C}^\delta(\theta_k)]$. By leveraging the convexity of Θ and the triangle inequality, we have $\|\xi_k\| \leq \eta_k[\|J_\mathcal{C}(\theta_k)\| + \|J_{\mathcal{C}^\delta}(\theta_k, v_k)\|] + \|J_{\mathcal{C}^\delta}(\theta_k, v_k) - \nabla\mathcal{C}^\delta(\theta_k)\|$. Since $\mathcal{C}^\delta$ is Lipschitz and definable by Lemma 5, and both $\theta_k \in \Theta$ and $v_k \in \mathbb{S}$ take on finite values, there exists a constant $C_J > 0$ such that $\|J_\mathcal{C}(\theta_k)\| + \|J_{\mathcal{C}^\delta}(\theta_k, v_k)\| \leq C_J$ for all $k \in \mathbb{N}$, which combined with (11b) gives $\sum_{k\in\mathbb{N}}\alpha_k\eta_k[\|J_\mathcal{C}(\theta_k)\| + \|J_{\mathcal{C}^\delta}(\theta_k, v_k)\|] < +\infty$. Next, leveraging Lemma 6, (11a), and the compactness of Θ, we conclude that $\sum_{k\in\mathbb{N}}\alpha_k\|J_{\mathcal{C}^\delta}(\theta_k) - \nabla\mathcal{C}^\delta(\theta_k)\| < +\infty$ by (Davis et al., 2020, Lemma 4.1), as the summability condition coincides with Assumption A.4 in Davis et al. (2020). This proves that $\sum_{k\in\mathbb{N}}\alpha_k\xi_k < +\infty$. Next, letting $G_k(\theta) = -\nabla\mathcal{C}^\delta(\theta_k) - \alpha_k^{-1}[\theta - \alpha_k\nabla\mathcal{C}^\delta(\theta) - \Pi_\Theta[\theta - \alpha_k\nabla\mathcal{C}^\delta(\theta)]]$, the update in (13) can be written as

$$\theta_{k+1} = \theta_k + \alpha_k[g_k + \xi_k], \quad g_k \in G_k(\theta_k).$$

The final argument of this proof relies on Theorem 3.2 of Davis et al. (2020), which we will invoke to prove convergence to a critical point of (6). To utilize Theorem 3.2 of Davis et al. (2020) we require all items in Assumption A of Davis et al. (2020) to hold true. First, observe that items 1-4 hold thanks to Assumption 2 and $\sum_{k\in\mathbb{N}}\alpha_k\xi_k < +\infty$. It only remains to show that item 5 holds, that is, that given any unbounded subset $\mathcal{K}$ of $\mathbb{N}$ for which $\theta_j \to \bar{\theta}$, $j \in \mathcal{K}$, we have $\text{dist}(1/k\sum_{j=0}^k g_j, G(\bar{\theta})) \to 0$, where $G(\bar{\theta}) = -\nabla\mathcal{C}^\delta(\bar{\theta}) - \mathcal{N}_\Theta(\bar{\theta})$ and $\mathcal{C}^\delta(\theta) = \mathbb{E}_v[\mathcal{C}(\theta + \delta v)]$. Since $G(\bar{\theta})$ is a convex set, we have $\text{dist}(1/k\sum_{j=0}^k g_j, G(\bar{\theta})) \leq 1/k\sum_{j=0}^k \text{dist}(g_j, G(\bar{\theta}))$, meaning that it suffices to show that $\text{dist}(g_j, G(\bar{\theta})) \to 0$ as $j \to \infty$, $j \in \mathcal{K}$. Since for each j we have $\Pi_\Theta[\theta_j - \alpha_j\nabla\mathcal{C}^\delta(\theta_j)] \in \theta_j - \alpha_j\nabla\mathcal{C}^\delta(\theta_j) - \mathcal{N}_\Theta(\Pi_\Theta[\theta_j - \alpha_j\nabla\mathcal{C}^\delta(\theta_j)])$, we have by definition that $g_j = -\nabla\mathcal{C}^\delta(\theta_j) - \alpha_j^{-1}z_j$, for some $z_j \in \mathcal{N}_\Theta(\Pi_\Theta[\theta_j - \alpha_j\nabla\mathcal{C}^\delta(\theta_j)])$, and therefore $g_j - G(\bar{\theta}) = -\nabla\mathcal{C}^\delta(\theta_j) - \alpha_j^{-1}z_j + \nabla\mathcal{C}^\delta(\bar{\theta}) + \mathcal{N}_\Theta(\bar{\theta})$. Due to the outer semicontinuity of

the normal cone $\mathcal{N}_\Theta$ of a convex set Θ, and that $\theta_{j+1} = \Pi_\Theta[\theta_j - \alpha_j \nabla \mathcal{C}^\delta(\theta_j)] \to \bar{\theta}$, we have that in the limit $\alpha_j^{-1} z_j \in \mathcal{N}_\Theta(\bar{\theta})$. Moreover, by continuity of $\nabla \mathcal{C}^\delta$, $\nabla \mathcal{C}^\delta(\theta_j) \to \nabla \mathcal{C}^\delta(\bar{\theta})$. This proves that all items in Assumption A of Davis et al. (2020) are satisfied. Since Assumption B of Davis et al. (2020) is also satisfied thanks to Lemma 5 and Theorem 5.8 of Davis et al. (2020), we conclude that (10) converges to a point satisfying $0 \in \nabla \mathcal{C}^\delta(\bar{\theta}) + \mathcal{N}_\Theta(\bar{\theta})$. By Lemma 3, this means that $0 \in \partial_\delta[C(\bar{\theta}) + P(\bar{\theta})] + \mathcal{N}_\Theta(\bar{\theta})$. Since $\bigcup_{\vartheta \in \delta\mathbb{B}} [\partial_c C(\theta + \vartheta) + \partial_c P(\theta + \vartheta)] \subseteq \bigcup_{\vartheta \in \delta\mathbb{B}} [\partial_c C(\theta + \vartheta)] + \bigcup_{\vartheta \in \delta\mathbb{B}} [\partial_c P(\theta + \vartheta)]$, and this inclusion is preserved if we consider the convex hull of both sets, we have that $\partial_\delta[C(\bar{\theta}) + P(\bar{\theta})] \subseteq \partial_\delta C(\bar{\theta}) + \partial_\delta P(\bar{\theta})$, and therefore there exist $\lambda_0, \lambda_1 \geq 0$ with $\lambda_0 + \lambda_1 = 1$ such that

$$0 \in \lambda_0 \partial_\delta C(\bar{\theta}) + \lambda_1 \partial_\delta P(\bar{\theta}) + \mathcal{N}_\Theta(\bar{\theta}), \quad \lambda_0, \lambda_1 \geq 0, \quad \lambda_0 + \lambda_1 = 1, \tag{14}$$

proving that the algorithm converges to a Goldstein Fritz-John δ-critical point of (12). ■

Under stronger assumptions on the constraint $P(\theta) = 0$ (for example the constraint qualification given in Grimmer and Jia (2025)), one can additionally prove that any feasible solution $\bar{\theta}$ satisfies (14) with $\lambda_0 > 0$, that is, that $\bar{\theta}$ is a KKT point of (7). We leave this as a direction for future work.

7. Simulation Results

We evaluate our approach on the 12-dimensional quadcopter model from Abdulkareem et al. (2022) whose state vector comprises the position (p_x, p_y, p_z), velocity (v_x, v_y, v_z), Euler angles (ϕ, ϑ, ψ), and angular velocity (p, q, r) in the body frame.[1] The control inputs are the rotation speeds ω_i of the four rotors. The system is subject to the constraints $\omega_i \in [0, 630]$, $v_x, v_y, v_z \in [-2, 2]$, $\phi, \vartheta, \psi \in [-\pi/4, \pi/4]$, and $p, q, r \in [-\pi/8, \pi/8]$. We implement (4) with $N = 12$ and

$$\ell_\theta(x, u) = \|x - x_{\text{ref}}\|_Q^2 + \|u - u_{\text{ref}}\|_R^2, \quad \ell_{N,\theta}(x) = \|x - x_{\text{ref}}\|_P^2,$$

where $x_{\text{ref}} = (-6, -3.5, 0, \mathbf{0}_9)$ and u_{ref} is the input required to maintain the drone at a hovering state. The parameter $\theta = (p_Q, p_R, p_P)$ defines the stage cost matrices $Q = \text{diag}(p_Q^2) + 10^{-6} I$ and $R = \text{diag}(p_R^2) + 10^{-6} I$, as well as the terminal cost $P = LL^\top + 10^{-6} I$, where L is a lower-triangular matrix containing the entries of p_P. To handle constraint violations, we relax the state constraints using slack variables, which are penalized using both a quadratic and a linear penalty (scaled by a factor of 25). We model the system as linear choosing $g(x, u) = Ax + Bu$, where the matrices A and B are identified via least-squares regression on 100 closed-loop trajectories collected near the target point under a stabilizing MPC controller. In practice, if such a controller is not available, these trajectories could instead be generated by a human pilot. The upper-level cost is

$$C(x, u) = \|x_T - x_{\text{ref}}\|_{\mathcal{P}}^2 + \textstyle\sum_{t=0}^{T-1} \|x_t - x_{\text{ref}}\|_{\mathcal{Q}}^2 + \|u_t - u_{\text{ref}}\|_{\mathcal{R}}^2$$

where $T = 200$, $\mathcal{Q} = \text{diag}(\mathbf{1}_6, 0.1 \cdot \mathbf{1}_6)$, $\mathcal{R} = 0.01 \cdot I$, and $\mathcal{P}$ obtained by solving the DARE with the identified dynamics and cost given by $\mathcal{Q}$ and $\mathcal{R}$. The penalty term is $P(\theta) = 10^4 \, \text{dist}_1(x(\theta), \mathcal{X}^{T+1})$.

We choose $\delta = 10^{-5}$, $\alpha_k = 2 \cdot 10^{-4} \log(k + 2)/(k + 1)^{0.8}$ and $\eta_k = 1/(k + 1)^\gamma$, with $\gamma \in \{0.25, 0.5, 0.75\}$, and train for 300 iterations starting with $Q = \mathcal{Q}$, $R = \mathcal{R}$, and $P = \mathcal{P}$. To isolate the contribution of each optimization component, we additionally train using the same

[1] The code will be made available at https://github.com/RiccardoZuliani98/bpmpc

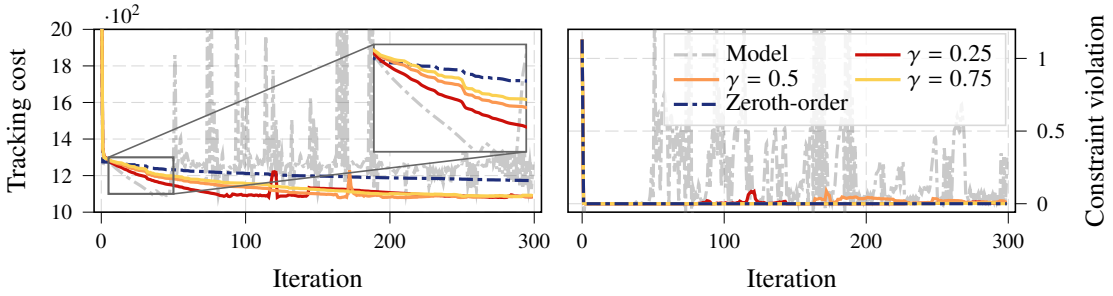

Figure 2: Tracking cost and constraint violation across iterations.

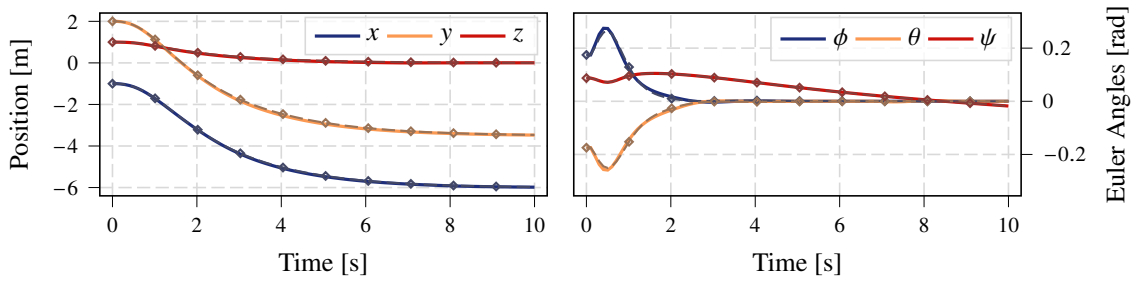

Figure 3: Comparison of position (left) and attitude (right) trajectories obtained with the trained controller (solid) and the controller tuned using the exact model (marked dash-dotted).

stepsizes and initial conditions with $\eta_k = 1$ and $\eta_k = 0$. The behavior of the tracking cost and the constraint violation across iterations can be seen in Figure 2. The proposed approach demonstrates the most efficient convergence, whereas the purely model-based method exhibits rapid improvement during the initial iterations but later becomes unstable and fails to converge. The purely zeroth-order method, on the other hand, achieves consistent progress, but at a slower rate. The proposed hybrid strategy benefits from fast initial transients, enabled by the model, and improved long-term convergence, supported by zeroth-order correction. Notably, smaller values of γ, corresponding to greater reliance on the model, lead to faster initial convergence, but also to more pronounced instability.

To further assess policy quality, we compare the trajectory generated by the trained controller with that obtained from tuning using the exact nonlinear model. As shown in Figure 3, the two trajectories exhibit strong qualitative agreement across all states. Quantitatively, the trained controller achieves a final cost of 1084.91, compared to 1081.3 for the controller optimized with perfect model knowledge. This corresponds to less than 1% suboptimality, indicating that the learned parameters are nearly optimal despite using an approximate model and noisy gradients.

More simulation results are reported in Appendix C.

8. Conclusions and Future Work

We introduced a policy optimization framework for MPC policies that combines model-based information with zeroth-order gradient estimation achieving fast learning transients while guaranteeing convergence to a critical point even under imperfect models. This approach is well-suited to settings where accurate modeling is difficult, offering robustness without sacrificing efficiency. We demonstrated the effectiveness of our algorithm on a nonlinear quadcopter task, showing that it achieves near-optimal performance while outperforming purely model-based and model-free baselines. Future work will focus on strengthening the safety guarantees of the approach.

References

Ademola Abdulkareem, Victoria Oguntosin, Olawale M Popoola, and Ademola A Idowu. Modeling and nonlinear control of a quadcopter for stabilization and trajectory tracking. *Journal of Engineering*, 2022(1):2449901, 2022.

Akshay Agrawal, Shane Barratt, Stephen Boyd, and Bartolomeo Stellato. Learning convex optimization control policies. In *Learning for Dynamics and Control*, pages 361–373. PMLR, 2020.

Brandon Amos, Ivan Jimenez, Jacob Sacks, Byron Boots, and J Zico Kolter. Differentiable mpc for end-to-end planning and control. *Advances in neural information processing systems*, 31, 2018.

Jérôme Bolte and Edouard Pauwels. Conservative set valued fields, automatic differentiation, stochastic gradient methods and deep learning. *Mathematical Programming*, 188(1):19–51, 2021.

Jérôme Bolte, Tam Le, Edouard Pauwels, and Tony Silveti-Falls. Nonsmooth implicit differentiation for machine-learning and optimization. *Advances in neural information processing systems*, 34: 13537–13549, 2021.

Michel Coste. *Introduction to o-minimal geometry*. Institut de recherche mathématique de Rennes (IRMAR), Rennes, France, 1999.

Damek Davis, Dmitriy Drusvyatskiy, Sham Kakade, and Jason D Lee. Stochastic subgradient method converges on tame functions. *Foundations of computational mathematics*, 20(1):119–154, 2020.

Ján Drgoňa, Karol Kiš, Aaron Tuor, Draguna Vrabie, and Martin Klaučo. Differentiable predictive control: Deep learning alternative to explicit model predictive control for unknown nonlinear systems. *Journal of Process Control*, 116:80–92, 2022.

John C Duchi, Peter L Bartlett, and Martin J Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674–701, 2012.

Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. *arXiv preprint cs/0408007*, 2004.

Benjamin Grimmer and Zhichao Jia. Goldstein stationarity in lipschitz constrained optimization. *Optimization Letters*, 19(2):425–435, 2025.

Sébastien Gros and Mario Zanon. Data-driven economic NMPC using reinforcement learning. *IEEE Transactions on Automatic Control*, 65(2):636–648, 2019.

Zhiyu He, Saverio Bolognani, Michael Muehlebach, and Florian Dörfler. Gray-box nonlinear feedback optimization. *arXiv preprint arXiv:2404.04355*, 2024.

Tianyi Lin, Zeyu Zheng, and Michael Jordan. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems*, 35: 26160–26175, 2022.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(52):1–11, 2017.

Patrick Speissegger. The pfaffian closure of an o-minimal structure. 1999.

Richard S Sutton, Andrew G Barto, and Ronald J Williams. Reinforcement learning is direct adaptive optimal control. *IEEE control systems magazine*, 12(2):19–22, 2002.

Jingzhao Zhang, Hongzhou Lin, Stefanie Jegelka, Suvrit Sra, and Ali Jadbabaie. Complexity of finding stationary points of nonconvex nonsmooth functions. In *International Conference on Machine Learning*, pages 11173–11182. PMLR, 2020.

Riccardo Zuliani, Efe Balta, and John Lygeros. Differentiable-by-design Nonlinear Optimization for Model Predictive Control. *arXiv preprint arXiv:2509.12692*, 2025a.

Riccardo Zuliani, Efe C Balta, and John Lygeros. BP-MPC: Optimizing the closed-loop performance of MPC using BackPropagation. *IEEE Transactions on Automatic Control*, 2025b.

Riccardo Zuliani, Efe C Balta, and John Lygeros. Closed-loop performance optimization of model predictive control with robustness guarantees. *European Journal of Control*, page 101319, 2025c.

Appendix A. Functions Definable in an o-minimal Geometry

Definition 8 (Definitions 1.4 and 1.5, Coste (1999)) *A collection $\mathcal{O} = (\mathcal{O}_n)_{n\in\mathbb{N}}$, where $\mathcal{O}_n \subset 2^{\mathbb{R}^n}$ for each $n \in \mathbb{N}$, is an o-minimal structure on $(\mathbb{R}, +, \cdot)$ if: 1) all semialgebraic subsets of $\mathbb{R}^n$ belong to $\mathcal{O}_n$; 2) $\mathcal{O}_1$ is the set of all finite unions of points and intervals; 3) $\mathcal{O}_n$ is a boolean subalgebra of $2^{\mathbb{R}^n}$; 4) $A \times B \in \mathcal{O}_{n+m}$ for all $A, B \in \mathcal{O}_n \times \mathcal{O}_m$; 5) the set $\{v \in \mathbb{R}^n : (v, w) \in A \text{ for some } w \in \mathbb{R}\}$, for any $A \in \mathcal{O}_{n+1}$, belongs to $\mathcal{O}_n$. A subset of $\mathbb{R}^n$ which belongs to $\mathcal{O}$ is said to be definable.*

A function $\varphi : \mathbb{R}^n \to \mathbb{R}^p$ is *definable* if its graph $\{(x, v) : v = \varphi(x)\}$ is definable. Definable functions are ubiquitous in control and optimization, containing, for example, all semialgebraic and globally subanalytic functions. Moreover, they are stable under most common operations, such as composition, differentiation, and affine transformations.

Appendix B. Differentiating Solutions of Optimization Problems

We focus on parametric quadratic programs in standard form following Zuliani et al. (2025b)

$$\begin{aligned} \min_{x} \quad & \tfrac{1}{2}x^\top Q(\theta)x + q(\theta)^\top x \\ \text{s.t.} \quad & F(\theta)x = f(\theta), \\ & G(\theta)x \le g(\theta). \end{aligned} \tag{15a}$$

$$\begin{aligned} \min_{z=(\lambda,\mu)} \quad & \tfrac{1}{2}z^\top H(p)z + h(p)^\top z, \\ \text{s.t.} \quad & \lambda \ge 0, \end{aligned} \tag{15b}$$

where θ is a parameter and, assuming $Q(\theta) \succ 0$, $H(p)$ and $h(p)$ are defined as

$$H(\theta) = \begin{bmatrix} G(\theta)Q(\theta)^{-1}G(\theta)^\top & G(\theta)Q(\theta)^{-1}F(\theta)^\top \\ F(\theta)Q(\theta)^{-1}G(\theta)^\top & F(\theta)Q(\theta)^{-1}F(\theta)^\top \end{bmatrix}, \; h(\theta) = \begin{bmatrix} G(\theta)Q(\theta)^{-1}q(\theta) + g(\theta) \\ F(\theta)Q(\theta)^{-1}q(\theta) + f(\theta) \end{bmatrix}.$$

MPC problems with linear (or affine) dynamics and polytopic constraints can always be expressed as in (15a). For systems with nonlinear dynamics, we can obtain a linear prediction model by linearizing at the current state or along the trajectory obtained as solution of the previous MPC problem. We discussed these ideas thoroughly in (Zuliani et al., 2025b, Section VI-A).

Given a dual optimizer $z(\theta)$, that is, a solution of (15b), one can retrieve the primal optimizer as

$$x(\theta) = \mathcal{G}(z, \theta) := -Q(\theta)^{-1}([F(\theta)^\top \; G(\theta)^\top]z + q(\theta)).$$

To ensure path-differentiability, we need the following definition.

Definition 9 *A minimizer of* (15a) *satisfies the* linear independence constraint qualification *(LICQ) if the rows of $G(\theta)$ associated to active constraints and the rows of $F(\theta)$ are linearly independent.*

Theorem 10 (Theorem 1, Zuliani et al. (2025b)) *Suppose $x(\theta)$ is a minimizer of* (15a) *satisfying LICQ and $Q(\theta) \succ 0$. Then the mapping $\theta \mapsto x(\theta)$ is locally unique, Lipschitz and definable. Moreover, we have that*

$$W + Q(\theta)^{-1}[G(\theta)^\top \; F(\theta)^\top]Z \in \mathcal{J}_x(\theta), \quad W \in \mathcal{J}_{\mathcal{G},\theta}(z, \theta)$$

where z solves (15b)*, and $Z = U^{-1}V$, with $U \in J_{P_C}(I - \gamma H(\theta)) - I$, $V \in -\gamma J_{P_C}(Az + B)$, is an element of the conservative Jacobian of the dual problem* (15b)*, with $J_{P_C} = \operatorname{diag}(\operatorname{sign}(\lambda), \mathbb{1}_{n_{eq}})$, and $A \in \mathcal{J}_H(\theta)$, $B \in \mathcal{J}_h(\theta)$.*

Fulfilling the assumptions of Theorem 10 is not difficult: strong convexity depends on the parameterization of the cost, and it is therefore possible to satisfy this condition by design. LICQ holds in many situations, for example if the constraints are box constraints on the state and input.

Appendix C. Ablation Studies

We repeated the experiments in Section 7 with different values of δ and over multiple random seeds (that determine the value of the random perturbations v_k needed to form the zeroth-order estimation). The results are reported in Table 1 (where we run for 1000 iterations and modified the stepsize to $\rho =$ `1e-4`), and in Table 2. All costs denote the best attained cost in the last 50 iterations. These randomized tests highlight the robustness of the approach.

Table 1: Ablation study on δ.

δ	`1e-04`	`5e-05`	`1e-05`	`5e-06`	`1e-06`	`5e-07`	`1e-07`
Cost	1092.9	1093.1	1098.1	1088.9	1097.1	1092.4	1093.8

Table 2: Ablation study on the random seed.

Seed	0	1	2	3	4	5	6	7	8	9
Cost	1094.1	1094.1	1092.2	1090.8	1092.8	1090.7	1093.4	1092.5	1104.9	1097.3

We report the following computation times (in seconds) for a laptop running Windows 11 (32GB RAM, i7-1165G7 2.80GHz): the average time to solve each QP was 0.0030 (variance: 2.8227×10^{-7}), the time to compute the Jacobian was 0.0061 (variance: 4.4733×10^{-7}), and the total time for each closed-loop iteration was 1.6032 (variance: 0.0121).