The Discrete-time Internal Model Principle of Time-varying Optimization

Gianluca Bianchin Bryan Van Scoy

Abstract— **Time-varying optimization problems arise in a variety of engineering applications. The available information about how the problem changes in time dictates the types of algorithms that are applicable to a particular problem as well as the types of convergence guarantees that may be proven. In this paper, we explore the fundamental properties shared by the entire class of gradient-based optimization algorithms for time-varying optimization. By casting the design of such algorithms as an output regulation problem for dynamical systems, we provide necessary and sufficient conditions for the existence of an algorithm that asymptotically tracks an optimizer of the problem of interest. When these conditions hold, we provide a design procedure to construct such an algorithm. As a fundamental limitation, we show that any algorithm that achieves exact tracking needs to incorporate an internal model of the temporal variation, which we refer to as the *internal model principle of time-varying optimization*.**

I. Introduction

Time-varying optimization arises in many engineering problems where parameters evolve over time [1]. Applications include optimal power flow with renewables, robotic obstacle avoidance, model predictive control, video feature extraction, high-resolution MRI, and real-time industrial optimization; see [1, Sections IV–V] and references therein.

The algorithms available to solve a time-varying optimization problem depend on the available information about how the problem changes in time. For instance, suppose one has access to the optimization problem at each point in time, but has no foreknowledge as to how the problem will change at the next iteration. In this case, any method from static optimization (e.g., [1], [2]) may be applied directly to the problem at each point in time; unfortunately, such algorithms can only achieve convergence to a neighborhood of an optimizer, where the size of the neighborhood depends on the convergence properties of the algorithm as well as how quickly the problem varies in time [3], [4].

The variability of the optimization problem, however, is not always entirely unknown (see, e.g., [5], [6]). Instead, suppose one has access to an oracle for the optimization problem (such as the gradient of the objective function) along with a model for how the optimization problem varies in time. In this case, the algorithm may exploit this information to asymptotically track an optimizer. This is the approach proposed in [7] for discrete-time problems and [8] for continuous-time ones. In [7], the algorithm has access to the gradient of the objective function along with knowledge of the poles of the z-transform of the time-varying parameter. Based on the internal model principle, for quadratic problems, this model of the time variation is then incorporated in the algorithm to achieve exact asymptotic tracking of the optimal trajectory. Preliminary observations on internal models for time-varying optimization have appeared in our previous work [8], which is limited to continuous-time optimization problems. This paper focuses on discrete-time methods, which align more closely with traditional optimization approaches [2], [5], yet inherently introduce new challenges and require distinct forms of characterization.

In this paper we study time-varying optimization problems in discrete time, and we pose the following questions:

1) What is the minimal amount of information needed to design an algorithm that asymptotically tracks an optimizer of a time-varying optimization problem?
2) When these conditions hold, how does one design such an algorithm?

To address these questions, we cast the analysis and design of a time-varying optimization algorithm as a nonlinear output regulation problem [9], which can be studied using tools from center manifold theory for maps [10]–[12]. Our main contributions are as follows:

1) We provide necessary and sufficient conditions for a discrete-time gradient-based algorithm to asymptotically track an optimizer of a time-varying optimization problem (Theorems 2 and 3).
2) We provide a design procedure to construct such an algorithm (Algorithm 1). The algorithm consists of an observer for the temporal variability combined with a function that zeros the gradient (see Definition 2).

We show that: i) when the dependence between the optimization problem and temporal variability is known or measurable, exact asymptotic tracking can be achieved without knowledge of how the problem changes with time (see Thm. 1), and ii) when the dependence between the optimization problem and the temporal variability is unknown, asymptotic tracking can be achieved only if the algorithm has knowledge of how the problem changes with time (see Thm. 2). As a result, the algorithm must embed an internal model of the temporal variation—a feature we term the *discrete-time internal model principle of time-varying optimization*, akin to its counterpart in control theory [13].

This material is based upon work supported in part by the National Science Foundation under Award No. 2347121 and in part by the WEL-T Investigator Programme. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

G. Bianchin is with the ICTEAM institute and the Department of Mathematical Engineering (INMA) at the University of Louvain, Belgium. Email: gianluca.bianchin@uclouvain.be

B. Van Scoy is with the Dept. of Electrical and Computer Engineering, Miami University, OH 45056, USA. Email: bvanscoy@miamioh.edu

II. Problem formulation

We consider the time-varying optimization problem:

$$\min_{x\in\mathbb{R}^n} f(x,\theta_k), \tag{1}$$

where $k \in \mathbb{N}_{\geq 0}$ denotes time or iteration, and $f : \mathbb{R}^n \times \Theta \to \mathbb{R}$, with $\Theta \subseteq \mathbb{R}^p$, is a loss function that is parametrized by the time-varying parameter vector $\theta : \mathbb{N}_{\geq 0} \to \Theta$. We make the following assumption on the loss function.

Assumption 1 (Properties of the objective function)**.** The map $x \mapsto f(x,\theta)$ is convex and $x \mapsto \nabla_x f(x,\theta)$ is Lipschitz continuous in $\mathbb{R}^n$, for each $\theta \in \Theta$. □

Convexity and smoothness are standard assumptions for this class of problems [14]. In (1), the parameter θ_k is used to model the temporal variability of the problem. We will require that θ_k belongs to a certain class of temporal variabilities, as specified next.

Assumption 2 (Class of temporal variabilities)**.** There exists a smooth (i.e., C^∞) vector field $s : \Theta \to \Theta$ and initial condition $\theta_0 \in \Theta$ such that θ_k satisfies

$$\theta_{k+1} = s(\theta_k), \tag{2}$$

for all $k \in \mathbb{N}_{\geq 0}$. Moreover, $\theta = 0$ is an equilibrium of (2) and the trajectories of (2) are bounded. □

Assumption 2 defines the class of temporal variations considered. It is a mild assumption, requiring only that θ_k is deterministic, sufficiently smooth, and exhibits bounded trajectories. We stress that, a priori, we do not assume that $s(\theta)$ nor θ_0 are known (see Problem 1 below). In line with [15], we will call (2) the *exosystem*.

For simplicity of the presentation, we assume that Θ is some neighborhood of the origin of $\mathbb{R}^p$. We put no restrictions on the size of this neighborhood (which is, e.g., allowed to be the entire space $\Theta = \mathbb{R}^p$), and thus on the size of θ_k. Moreover, there is no restriction with asking that Θ contains the origin because, if θ_k takes values in the neighborhood of any other point, such a point can be shifted to the origin through a time-invariant change of variables.

We are interested in designing an optimization algorithm that computes and tracks a *critical trajectory* $x_k^\star$ of (1), which is defined[1] implicitly as:

$$0 = \nabla_x f(x_k^\star, \theta_k), \quad \forall k \in \mathbb{N}_{\geq 0}. \tag{3}$$

Moreover, we will denote by $x_\circ^\star \in \mathbb{R}^n$ a point such that[2]

$$0 = \nabla_x f(x_\circ^\star, 0), \tag{4}$$

and assume that $x_\circ^\star \in \mathbb{R}^n$ is locally unique.

In line with the existing literature [1], [14], [17]–[19], we focus on gradient-type algorithms to solve (1); that is, algorithms that have access to functional evaluations of:

$$(k, x) \mapsto \nabla_x f(x, \theta_k). \tag{5}$$

[1]Existence of a critical trajectory can be ensured under standard assumptions on the optimization; for example, coercivity of the cost [16].

[2]We stress that $x_k^\star$ and $x_\circ^\star$ are distinct quantities. While $x_k^\star$ is a sequence, $x_\circ^\star$ is a constant vector.

More precisely, we consider the class of optimization algorithms described by a dynamic state $z_k \in \mathcal{Z} \subseteq \mathbb{R}^{n_c}$, $n_c \in \mathbb{N}_{>0}$. The algorithm generates a sequence of points $x_k \in \mathbb{R}^n$ (called *exploration signal*) and has access to functional evaluations $y_k = \nabla_x f(x_k, \theta_k)$ of (5) at these points (called *gradient feedback signal*). Mathematically, the optimization algorithm is described by:

$$z_{k+1} = F_c(z_k, y_k), \qquad x_k = G_c(z_k), \tag{6a}$$

together with the *gradient-feedback signal*:

$$y_k = \nabla_x f(x_k, \theta_k), \tag{6b}$$

where $F_c : \mathcal{Z} \times \mathbb{R}^n \to \mathcal{Z}$ and $G_c : \mathcal{Z} \to \mathbb{R}^n$ are functions to be designed. In the remainder, we refer to (6) as a *dynamic gradient-feedback optimization algorithm*. We will assume that the functions $F_c(z, y)$ and $G_c(z)$ are such that[3]

$$F_c(z_\circ^\star, 0) = z_\circ^\star, \qquad x_\circ^\star = G_c(z_\circ^\star), \tag{7}$$

for some $z_\circ^* \in \mathcal{Z}$ locally unique. This ensures that the optimization algorithm (6) has an equilibrium at $z = z_\circ^\star$, with corresponding gradient feedback signal equal to zero.

The dynamics of the optimization algorithm (6), coupled with the time-variability generator (2), have the form:

$$\begin{aligned} z_{k+1} &= F_c(z_k, y_k), && \text{(8a)}\\ \theta_{k+1} &= s(\theta_k), && \text{(8b)}\\ y_k &= \nabla_x f(G_c(z_k), \theta_k). && \text{(8c)} \end{aligned}$$

Definition 1. We say that (8) *asymptotically tracks a critical trajectory of* (1) *with respect to initializations in the set* $\Theta_\circ \subseteq \Theta$ if, for each initial condition (z_0, θ_0) with z_0 in some neighborhood of $z_\circ^\star$ and $\theta_0 \in \Theta_\circ$, the solution of (8) satisfies $y_k \to 0$ as $k \to \infty$. □

In practice, the initial condition θ_0 to the exosystem (2) may not be known; Definition 1 accounts for such uncertainty by allowing θ_0 to be anywhere in the set $\Theta_\circ$. We also stress that the tracking notion of Definition 1 is of local nature; the reason being that we allow for optimizations that admit multiple local minima and exosystems that are nonlinear and whose trajectories can have arbitrary asymptotic behaviors (converge to equilibrium points, limit cycles, chaotic motion, etc.). We are now ready to make our objective formal.

Problem 1 (Minimal knowledge for asymptotic tracking)**.** Consider the class of optimization algorithms (6). Determine the minimal necessary knowledge (beyond (5)) concerning the exosystem and the optimization problem, needed to design an algorithm as in (6) that tracks a critical trajectory of (1) with respect to initializations in some set $\Theta_\circ$. In addition, provide a method to design $F_c(z, y)$, $G_c(z)$, and n_c, requiring the minimal necessary knowledge. □

[3]Notice that this is without loss of generality since $F_c(z, y)$ and $G_c(z)$ are to be designed and $x_\circ^\star$ is known through (4).

III. THE PARAMETER-FEEDBACK PROBLEM

In many practical cases, having access to functional evaluations as in (5) (and hence to the signal y_k) is a byproduct of having access to both the function $\nabla_x f(x,\theta)$ and knowledge or measurements of θ_k. In this section, we analyze this scenario (hence, we will assume that $\nabla_x f(x,\theta)$ and θ_k are known at each k). We anticipate that the results derived in this section will also serve as an intermediate step to tackle the more challenging problem where only oracle evaluations of (5) are available, which is the focus of Section IV.

When the algorithm has access to both θ_k at each k and to the function $\nabla_x f(x,\theta)$, the measurements y_k do not provide additional information. Hence, the algorithm (6) can be replaced by the algebraic relationship[4]:

$$x_k = H_c(\theta_k), \tag{9}$$

where $H_c : \Theta \to \mathbb{R}^n$ is a mapping to be designed; we will require that H_c is of class C^0 and, in line with (7), that $x^\star_\circ = H_c(0)$ (cf. (7)). Because of the explicit dependence on θ_k, we will refer to (9) as a *static parameter-feedback* optimization algorithm. Our objective is to design the map H_c so that the composition of (2), (6b), and (9):

$$y_k = \nabla_x f(H_c(\theta_k),\theta_k), \qquad \theta_{k+1} = s(\theta_k), \tag{10}$$

tracks, with zero asymptotic error, a critical trajectory of (1).

Solvability of the static parameter-feedback problem will depend on the existence of a function that zeros the gradient.

Definition 2 (Mapping zeroing the gradient)**.** A mapping $H_c : \Theta \to \mathbb{R}^n$ *zeros the gradient* at the point $\theta \in \Theta$ if

$$0 = \nabla_x f(H_c(\theta),\theta). \tag{11}$$

Moreover, H_c zeros the gradient *on a set* $\Theta_\circ \subseteq \Theta$ if (11) holds for all $\theta \in \Theta_\circ$. □

The following definition is instrumental.

Definition 3 (Limit point and limit set)**.** A point $\theta_\omega \in \Theta$ is called a *limit point with respect to initialization* $\theta_\circ \in \Theta$ if there exists a sequence $\{k_i\}_{i\in\mathbb{N}_{\geq 0}}$, with $k_i \to \infty$ as $i \to \infty$, such that the trajectory of (2) with $\theta_0 = \theta_\circ$ satisfies $\theta_{k_i} \to \theta_\omega$ as $i \to \infty$. For $\theta_\circ \in \Theta$, let $\Omega(\theta_\circ)$ denote the set of all limit points (i.e., for all sequences $\{k_i\}_{i\in\mathbb{N}_{\geq 0}}$) of (2) with respect to the initialization $\theta_\circ$. Given $\Theta_\circ \subseteq \Theta$, the set $\Omega(\Theta_\circ) := \cup_{\theta_\circ\in\Theta_\circ}\Omega(\theta_\circ)$ is called the *limit set with respect to initializations in* $\Theta_\circ$ [20]. □

Intuitively, $\Omega(\Theta_\circ)$ denotes the set of all limit points (equilibria, limit cycles, etc.) that can be reached by the exosystem when initialized at points in $\Theta_\circ$. Notice also that, by Assumption 2, $\Omega(\Theta_\circ)$ is contained in some neighborhood of the origin of $\mathbb{R}^p$. For example, if the exosystem (2) is linear and the origin is a Lyapunov stable equilibrium, then $\Omega(\Theta_\circ)$ is some neighborhood of the origin, whose radius depends on the radius of the initialization set $\Theta_\circ$. The following result characterizes all parameter-feedback optimization algorithms that achieve asymptotic tracking of a critical trajectory.

Theorem 1 (Parameter-feedback algorithm characterization)**.** Let Assumptions 1–2 hold, and let $\Theta_\circ \subseteq \Theta$. The parameter-feedback algorithm (10) asymptotically tracks a critical trajectory of (1) with respect to initializations in $\Theta_\circ$ if and only if the mapping $H_c(\theta)$ zeros the gradient on $\Omega(\Theta_\circ)$. □

Proof. (Only if) Suppose $y_k \to 0$ as $k \to \infty$ for initializations $\theta_0 \in \Theta_\circ$; we will show that H_c zeros the gradient on $\Omega(\Theta_\circ)$. By Assumption 2, the trajectories of (2) are bounded, and thus, by the Bolzano–Weierstrass theorem, there exists an increasing subsequence $\{k_i\}_{i\in\mathbf{N}_{\geq 0}}$ such that the trajectory θ_{k_i} converges to some limit point $\theta_\omega \in \Omega(\theta_0)$ as $i \to \infty$. Then,

$$\lim_{i\to\infty} y_{k_i} = \lim_{i\to\infty} \nabla_x f(H_c(\theta_{k_i}),\theta_{k_i}) = \nabla_x f(H_c(\theta_\omega),\theta_\omega), \tag{12}$$

where the second equality follows from the continuity of the gradient (Assumption 1) and that of H_c. Since $y_k \to 0$ as $k \to \infty$, the left-hand side of (12) is zero, which implies that H_c zeros the gradient on θ_ω. Since this holds for any limit point $\theta_\omega \in \Omega(\Theta_\circ)$, H_c zeros the gradient on $\Omega(\Theta_\circ)$.

(If) Suppose $\theta_0 \in \Theta_\circ$ and that H_c zeros the gradient on $\Omega(\Theta_\circ)$. The right-hand side of (12) is then zero, which implies the existence of a sequence k_i such that $y_{k_i} \to 0$ as $i \to \infty$. Since this holds for any limit point $\theta_\omega \in \Omega(\Theta_\circ)$, any convergent subsequence of y_k converges to zero. Moreover, y_k is bounded due to Lipschitz continuity of the gradient, so y_k also converges to zero as $k \to \infty$. By iterating the reasoning for all $\theta_0 \in \Theta_\circ$, it follows that $y_k \to 0$ for all initializations $\theta_0 \in \Theta_\circ$, and thus the statement follows. □

Intuitively, the theorem states that the parameter-feedback algorithm asymptotically tracks a critical trajectory if and only if the mapping H_c zeros the gradient on the limit set of the exosystem. We summarize this result as follows.

Solution to Problem 1 with θ_k measurable. There exists a parameter-feedback algorithm (9) that achieves exact asymptotic tracking if and only if there exists a mapping $H_c(\theta)$ that zeros the gradient on the limit set $\Omega(\Theta_\circ)$. When this holds, the minimal knowledge needed to design such an algorithm is the parameter θ_k at each iteration k and the gradient function $\nabla_x f(x,\theta)$. Moreover, a procedure to design such an algorithm is to first determine the limit set $\Omega(\Theta_\circ)$ and then find a mapping $H_c(\theta)$ that zeros the gradient on the limit set, in which case such an algorithm is (9). □

Remark 1 (Knowledge of $\Omega(\Theta_\circ)$)**.** Designing a parameter-feedback algorithm as in (9) can be accomplished without exact knowledge of the limit set $\Omega(\Theta_\circ)$. Indeed, it follows from the boundedness of the trajectories (cf. Assumption 2) and the sufficiency part of the proof of Theorem 1 that if H_c zeros the gradient on some subset of $\mathbb{R}^p$ containing $\Omega(\Theta_\circ)$, then the choice (9) ensures that $y_k \to 0$ as $k \to \infty$. □

We conclude this section by illustrating the design procedure for parameter-feedback algorithms on a quadratic problem.

[4]While one could consider a dynamic optimization algorithm of the form $z_{k+1} = F_c(z_k,\theta_k)$ and $x_k = G_c(z_k)$, we will prove in Theorem 1 that a dynamic structure is unnecessary.

Example 1. Consider an instance of (1) with quadratic cost and time-variability that depends linearly on θ_k:

$$f(x, \theta_k) = \tfrac{1}{2} x^\mathsf{T} R x + x^\mathsf{T} Q \theta_k, \tag{13}$$

with $R = R^\mathsf{T} \in \mathbb{R}^n$ and $Q \in \mathbb{R}^{n\times p}$. Notice that (13) admits a critical point for arbitrary θ_k if and only if $\operatorname{Im} Q \subseteq \operatorname{Im} R$, in which case $x_k^\star$ is unique. In this case, designing an optimization algorithm amounts to finding x_k such that we regulate to zero the signal: $y_k = \nabla_x f(x_k, \theta_k) = Rx_k + Q\theta_k$. Applying Theorem 1 requires finding a linear transformation $H_c(\theta) = H_c \theta, H_c \in \mathbb{R}^{n\times p}$, such that $0 = (RH_c + Q)\theta$ for all θ in some neighborhood of the origin. Using $\operatorname{Im} Q \subseteq \operatorname{Im} R$, we can choose $H_c = -R^\dagger Q$, where $R^\dagger$ is the pseudo-inverse of R; this choice for $H_c(\theta)$ zeros the gradient globally in $\mathbb{R}^p$. By substituting into (10), we have $y_k = RH_c\theta_k + Q\theta_k = 0$ for all $k \in \mathbb{N}_{\geq 0}$. Namely, the gradient is identically zero at all times. Interestingly, this behavior originates for two reasons: (i) θ_k is measurable at each k, and (ii) $H_c(\theta)$ obtained for this particular problem zeros the gradient on the entire $\mathbb{R}^p$ (not just some limit set of the trajectories of θ). When one of these two properties fails (as in Section IV, shortly below), this behavior can no longer be expected. □

Remark 2 (Existence of a mapping that zeros the gradient)**.** The implicit function theorem [21] provides sufficient conditions for the existence of a function H_c that zeros the gradient. Under Assumptions 1 and 2, for instance, there exists a function that zeros the gradient on a neighborhood $\Theta_\circ$ of the origin of $\mathbb{R}^p$ if, for some neighborhood $X_\circ$ of $x_\circ$,

(i) the loss function f is C^1 on $X_\circ \times \Theta_\circ$,

(ii) $x \mapsto f(x, \theta)$ is C^2 on $X_\circ$ for each $\theta \in \Theta_\circ$, and

(iii) the Hessian $\nabla^2_{xx} f(x, \theta)|_{x=x_\circ^\star, \theta=0}$ is positive definite.

Beyond existence of such a mapping, the implicit function theorem also provides the linearization of such a function about the origin; see [21, Ch. 9]. □

IV. The dynamic gradient-feedback problem

In the previous section, we analyzed algorithms based on the assumption that the gradient evaluations y_k are derived from explicit knowledge of the function $\nabla_x f(x, \theta)$ and the signal θ_k. In this section, we relax these assumptions and require only that y_k is available through oracle evaluations of (5). For this reason, we will shift our attention back to the general class of dynamic gradient-feedback algorithms (6).

A. Characterization of gradient-feedback algorithms

We begin with the following instrumental characterization.

Theorem 2 (Gradient-feedback algorithm characterization)**.** Suppose Assumptions 1–2 hold, assume that $F_c(z, y)$ and $G_c(z)$ are such that the equilibrium $z = z_\circ^\star$ of

$$z_{k+1} = F_c(z_k, \nabla_x f(G_c(z_k), 0)) \tag{14}$$

is locally exponentially stable. The gradient-feedback optimization algorithm (8) asymptotically tracks a critical trajectory of (1) with respect to initializations in $\Theta_\circ$ if and only if there exists a C^2 mapping $z = \sigma(\theta)$ with $\sigma(0) = z_\circ^\star$, defined on $\Omega(\Theta_\circ)$, which satisfies:

$$\sigma(s(\theta_\omega)) = F_c(\sigma(\theta_\omega), 0), \tag{15a}$$

$$0 = \nabla_x f(G_c(\sigma(\theta_\omega)), \theta_\omega), \tag{15b}$$

for all limit points $\theta_\omega \in \Omega(\Theta_\circ)$. □

Proof. (Only if) The coupled dynamics (8) have the form:

$$\begin{aligned} z_{k+1} &= (A_c + B_c RM) z_k + B_c Q \theta_k + \chi(z_k, \theta_k), \\ \theta_{k+1} &= S\theta_k + \psi(\theta_k), \end{aligned} \tag{16}$$

for some mappings $\chi(z, \theta)$ and $\psi(\theta)$ that vanish at the fixed point along with their first-order derivatives, where the following matrices are Jacobians evaluated at the fixed point:

$$\begin{aligned} A_c &= \left[\frac{\partial F_c}{\partial z}\right]_{(z,y)=(z_\circ^\star, 0)}, & B_c &= \left[\frac{\partial F_c}{\partial y}\right]_{(z,y)=(z_\circ^\star, 0)}, \\ R &= \left[\frac{\partial \nabla_x f}{\partial x}\right]_{(x,\theta)=(x_\circ^\star, 0)}, & M &= \left[\frac{\partial G_c}{\partial z}\right]_{z=z_\circ^\star}, \\ Q &= \left[\frac{\partial \nabla_x f}{\partial \theta}\right]_{(x,\theta)=(x_\circ^\star, 0)}, & S &= \left[\frac{\partial s}{\partial \theta}\right]_{\theta=0}. \end{aligned} \tag{17}$$

By assumption, the eigenvalues of the matrix $A_c + B_c RM$ are located in the open unit disk. Then by [10, Thm. 6], (16) has a center manifold at $(z_\circ^\star, 0)$, which is the graph of a mapping $z = \sigma(\theta)$ with $\sigma(\theta)$ satisfying (see [10, Eq. (2.8.4)])

$$\sigma(s(\theta)) = F_c(\sigma(\theta), \nabla_x f(G_c(\sigma(\theta)), \theta)). \tag{18}$$

This proves that (15a) holds. The proof that (15b) holds follows by iterating the *(Only if)* part of the proof of Theorem 1.

(If) We now prove that (15) implies $y_k \to 0$. Suppose there exists a C^2 mapping $z = \sigma(\theta)$ such that (15) holds for all $\theta_\omega \in \Omega(\Theta_\circ)$. It follows from (15) that the graph of σ (i.e., $(\sigma(\theta), \theta)$) is a center manifold for the coupled dynamics (8). Moreover, by[5] [10, Lem. 1], this manifold is locally attractive, meaning that $z_k \to \sigma(\theta_k)$ as $t \to \infty$. Then, the fulfillment of (15b) guarantees that the right-hand side of (12) satisfies $y_k \to 0$. The conclusion then follows by iterating the *(If)* part of the proof of Theorem 1. □

The two conditions in (15) fully characterize the class of optimization algorithms that achieve exact asymptotic tracking. In other words, an algorithm of the class (6) tracks a critical trajectory if and only if, for some mapping σ, the composite function $G_c \circ \sigma$ zeros the gradient on the limit set of the exosystem (see (15b)), and the controller $F_c(z, y)$ is algebraically related to the exosystem $s(\theta)$ as given by (15a). Notice that, by combining (15b) with Theorem 1,

$$x_k = G_c(\sigma(\theta_k)) := H_c(\theta_k), \tag{19}$$

is a parameter-feedback algorithm for (1) (cf. (9)).

[5] Notice that attractivity of the center manifold is proven for ordinary differential equations in [10, Lem. 1]; a proof of the discrete-time counterpart follows by using an analogous reasoning.

Remark 3 (The internal model principle)**.** We interpret (15a) as the *(discrete-time) internal model principle of time-varying optimization,* as it expresses the requirement that any optimization algorithm that achieves asymptotic tracking must include an internal model of the exosystem. □

Remark 4 (Tracking accuracy vs internal model fidelity)**.** In general, the tracking accuracy of the optimization algorithm will depend on the fidelity of the internal model as well as the asymptotic behavior of the exosystem. This property is discussed in detail in [8, §5] for quadratic problems. We stress that this is not a limitation of our approach, but of *any* algorithm seeking exact tracking of a critical trajectory. In this sense, the internal model principle from Theorem 2 (see also Remark 3) provides a *fundamental limitation* that should be carefully considered when designing optimization algorithms for time-varying problems. □

It is important to note that, by Theorem 2, the exosystem state θ and that of the optimization z must be related, in the limit set of the exosystem, by the relationship:

$$z_k = \sigma(\theta_k). \tag{20}$$

Intuitively, (20) is interpreted as the existence of a change of coordinates between the state of the exosystem and that of the optimization algorithm.

Remark 5 (Special cases)**.** An important special case is obtained when $\sigma(\theta)$ is the identity operator; in this case, the internal model condition (15a) simplifies to $s(\theta) = F_c(\theta, 0)$, which states that the controller vector field $F_c(z, y)$ must coincide with that of the exosystem $s(\theta)$ on the limit set of y and the exosystem. In this case, (20) gives $z_k = \theta_k$, meaning that the controller state z_k and that of the exosystem θ_k must coincide on the limit set. □

B. Detectability of the exosystem

While Theorem 2 provides a full characterization of all gradient-feedback algorithms that achieve tracking, it remains to address under what conditions on the optimization problem such an algorithm exists. To state our results, for simplicity, we require that the exosystem description (2) is non-redundant, in the sense that it does not include states that do not affect the optimization. To formalize this requirement, the following technical notion is needed.

Definition 4 (Exponential detectability [22])**.** The exosystem (2) is called *exponentially detectable* from the gradient feedback signal (6b) if there exists a dynamical system

$$\hat{\theta}_{k+1} = g(\hat{\theta}_k, y_k), \tag{21}$$

where g is a smooth mapping with $g(0, 0) = 0$ such that: (i) if $\hat{\theta}_0 = \theta_0$, then $\hat{\theta}_k = \theta_k$ for all $k \in \mathbb{N}_{\geq 0}$, and (ii) there exist an open neighborhood Θ_1 of the origin of $\mathbb{R}^p$ and positive constants M and $0 < a < 1$ such that, if $\hat{\theta}_0 - \theta_0 \in \Theta_1$, then $\|\hat{\theta}_k - \theta_k\| \leq M a^k \|\hat{\theta}_0 - \theta_0\|$ for all $k \in \mathbb{N}_{\geq 0}$. In this case, (21) is called a *local exponential observer* [22]. □

In other words, detectability refers to the ability to infer the internal state of the exosystem from measurements of the gradient signal y_k. We then make the following assumption.

Assumption 3 (Detectability of the exosystem)**.** The exosystem (2) is exponentially detectable from the gradient-feedback signal (6b). □

Lack of detectability corresponds to situations where certain modes of θ_k do not affect the gradient y_k, and hence the critical points. If this were the case, these modes could be removed from θ_k without altering the optimization problem. From a technical standpoint, it can be shown that exponential detectability is necessary for exact asymptotic tracking [22, Thm. 4.3]. For this reason, Assumption 3 is without loss of generality, and is made here for simplicity[6].

C. Existence of gradient-feedback algorithms

We are now able to characterize when there exists a dynamic gradient-feedback that achieves exact asymptotic tracking.

Theorem 3 (Existence of gradient-feedback algorithms)**.** Suppose Assumptions 1–3 hold. There exists an algorithm as in (6a) that achieves exact asymptotic tracking if and only if there exists a mapping $H_c : \Theta \to \mathbb{R}^n$ that zeros the gradient on the limit set of (2) with respect to its initializations. □

Proof. (Only if) By Theorem 2, there exists a mapping $z = \sigma(\theta)$ such that (15b) holds. Then, the gradient condition (11) holds with $H_c(\theta) = G_c(\sigma(\theta))$.

(If) We will prove this claim by constructing a gradient-feedback algorithm that achieves $y_k \to 0$ as $k \to \infty$. By Assumption 3, there exists a neighborhood Θ_1 of the origin of $\mathbb{R}^p$ and a dynamical system

$$\hat{\theta}_{k+1} = g(\hat{\theta}_k, y_k), \tag{22}$$

such that $\hat{\theta}_k \to \theta_k$ exponentially, for any $\hat{\theta}_0 - \theta_0 \in \Theta_1$. Consider then the optimization algorithm

$$F_c(z, y) = g(z, y), \qquad G_c(z) = H_c(z), \tag{23}$$

where $H_c(z)$ is as in (11). The claim then follows by application of Theorem 2 with σ the identity operator. □

Interestingly, this result shows that existence of a dynamic gradient-feedback algorithm is equivalent to that of a static parameter-feedback one. We thus conclude the following.

Solution to Problem 1 with y_k measurable. There exists a dynamic gradient-feedback algorithm (6) that achieves exact asymptotic tracking if and only if there exists a mapping $H_c(\theta)$ that zeros the gradient on the limit set $\Omega(\Theta_\circ)$. When this holds, the minimal knowledge needed to design such an algorithm is the exosystem mapping $s(\theta)$ and the gradient function $\nabla_x f(x, \theta)$. Moreover, a procedure to design such an algorithm is given in Section IV-D. □

We illustrate the applicability of the results next.

Example 2. Consider the quadratic problem studied in Example 1, and assume that the exosystem follows the linear model $\theta_{k+1} = S\theta_k$ for some $S \in \mathbb{R}^{p \times p}$. According to Theorem 3, an optimization algorithm given by

$$z_{k+1} = A_c z_k + B_c y_k, \quad x_k = G_c z_k, \quad y_k = R x_k + Q\theta,$$

[6]Note that exponential detectability does *not* require the gradient to depend explicitly on the entire exosystem state θ_k; see the example in §V.

where $A_c \in \mathbb{R}^{n_c \times n_c}$, $B_c \in \mathbb{R}^{n_c \times n}$, $G_c \in \mathbb{R}^{n \times n_c}$, achieves asymptotic tracking if and only if there exists a linear transformation $\Sigma \in \mathbb{R}^{n_c \times p}$ such that

$$0 = (\Sigma S - A_c \Sigma)\theta, \qquad 0 = (RG_c\Sigma + Q)\theta, \tag{24a}$$

for all θ in the limit set of the exosystem. □

D. Design of gradient-feedback optimization algorithms

Theorem 3 exactly characterizes the existence of a gradient-feedback algorithm. Synthesizing such an algorithm, however, is challenging as one needs to design an exponential observer for the exosystem. We next show that this process can be accomplished by having access only to first-order information on the exosystem. We begin by presenting an instrumental lemma; its statement hinges on the notation:

$$Q = \left[\frac{\partial \nabla_x f}{\partial \theta}\right]_{(x,\theta)=(x_\circ^\star,0)}, \qquad S = \left[\frac{\partial s}{\partial \theta}\right]_{\theta=0}. \tag{25}$$

Lemma 4 (First-order detectability of exosystem)**.** There is an exponential observer for the exosystem (2) if and only if the pair (Q, S) is detectable. □

Proof. The claim follows directly from [22, Cor. 3.4]. □

Harnessing this tool, a technique to design an exponential observer of the exosystem is presented in Algorithm 1. Here, a linear Luenberger observer is used to estimate the exosystem state (see line 4), and a parameter feedback algorithm is then applied to the estimated state to regulate the gradient to zero – precisely, $G_c(z)$ is designed following the approach of Theorem 1 (see line 3).

Algorithm 1: Gradient-feedback algorithm design

Data: $s(\theta)$, $\nabla_x f(x,\theta)$, $H_c(\theta)$ satisfying (15), Jacobian matrices Q and S in (25)

1. $n_c \leftarrow n$;
2. $L \leftarrow$ any matrix such that $S - LQ$ is Schur stable;
3. $G_c(z) \leftarrow H_c(z)$;
4. $F_c(z,y) \leftarrow s(z) + L(y - \nabla_x f(H_c(z), z))$;

Result: $F_c(z,y)$, $G_c(z)$, and n_c that solve Problem 1

We illustrate the applicability of (15) on a quadratic problem in the following example.

Example 3. Consider the quadratic problem in Example 2. Direct application of Algorithm 1 gives: $A_c = S$, $B_c = L$, and $G_c = -R^\dagger Q$, where L is such that $S - LQ$ is Schur stable; notice that this choice satisfies (24) with $\Sigma = I$. □

V. Simulation Results

In this section, we illustrate our approach through numerical simulations. We consider the following instance of (1):

$$\min_{x\in\mathbb{R}} f(x,\theta_k) := \frac{1}{2}\left(x - \theta_k^{(1)}\right)^2 + \kappa \log(1 + e^{\mu x}), \tag{26}$$

where $f : \mathbb{R} \times \Theta \to \mathbb{R}, \Theta = \mathbb{R}^2$, and we utilized the vector notation $\theta_k = (\theta_k^{(1)}, \theta_k^{(2)})$ (the choice to use $\Theta = \mathbb{R}^2$ instead of $\Theta = \mathbb{R}^1$ will be discussed shortly below). Eq. (26) models a logistic regression problem with a time-varying regularization term. Intuitively, an optimizer of (26) is a point that tracks the time-varying signal $\theta_k^{(1)}$, while seeking to avoid large values of x, which are penalized by the logistic term. For our experiments, we choose $\mu = 0.5$ and $\kappa = 1$. The function $f(x,\theta)$ satisfies Assumption 1; in particular, the cost is strongly convex in x (since $\nabla_{xx} f(x,\theta) = 1 + \kappa\mu^2 \frac{\exp(\mu x)}{[1+\exp(\mu x)]^2} \geq 1$), and thus the optimizer is unique for each θ. We let $\theta_k^{(1)} = \cos(\omega k)$, which can be generated by a two-dimensional linear exosystem $\theta_{k+1} = S\theta_k$ (hence the choice $\Theta = \mathbb{R}^2$). For our simulations, we generate matrix S by discretizing a continuous-time linear system with state matrix $S_{\text{ct}} = [0, 1; -\omega^2, 0]$ with $\omega = 0.2$, yielding $S = [0.9801, 0.9933; -0.0397, 0.9801]$. Notice that, in this case, Assumption 3 is satisfied with $\Theta_1 = \Theta$.

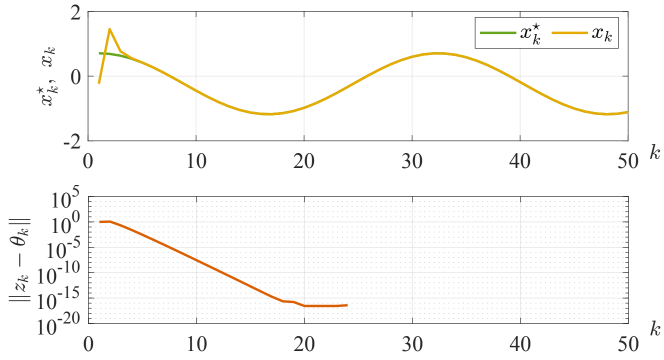

Fig. 1: Simulation results illustrating the behavior of an algorithm synthesized using Algorithm 1 for the problem (26). The proposed algorithm successfully computes the time-varying optimizer of (26) with exponential rate of convergence. Absence of a line means that the value of the timeseries is numerically zero. See Section V for a discussion.

We applied Algorithm 1 with $Q = [-1, 0]$; we chose L such that the spectral radius of $S - LQ$ is 0.1. Moreover, a mapping zeroing the gradient was computed numerically, yielding $H_c(\theta) = (0.9819 \cdot \theta^{(1)} - 0.2469, 0)$. From the simulations, we can conclude the following: from Fig. 1 (bottom), we see that $z_k \to \theta_k$ exponentially, and thus z_k is a local exponential observer for θ_k; from Fig. 1 (top), we see that $\|x_k - x_k^\star\| \to 0$, and thus the algorithm converges to the time-varying optimizer of (26).

In Fig. 2, we plot the error $\|x_k - x_k^\star\|$ of the algorithm proposed here and compare it with the prediction-correction algorithm [3]. Here, we used $\omega = 0.02$. The prediction-correction algorithm was implemented following [3, Alg. 1] with stepsize $\gamma = 0.2$. Numerically, we are led to conclude that, for this problem, our approach outperforms the prediction-correction algorithm both in convergence rate and in asymptotic precision. The difference in performance can be further appreciated by varying the spectral radius of $S - LQ$ for the exogenous signal observer in the set $\{0.1, 0.01\}$ and by varying the horizon of prediction $\tau \in \{1, 5\}$ in [3]. As expected, reducing the spectral radius of the observer and increasing the prediction horizon improve both the rate of convergence and the asymptotic precision of the two

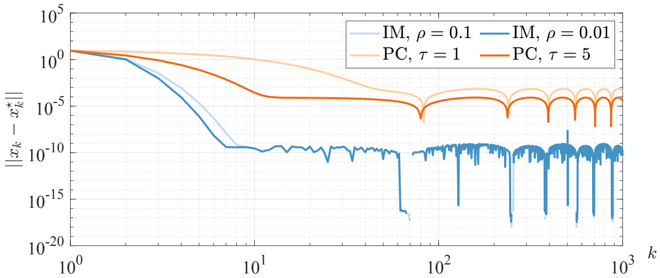

Fig. 2: Comparison between the approach proposed here (labeled IM for Internal Model in the plot) and the Prediction Correction (labeled PC in the plot) algorithm [3, Alg. 1] for the problem (26). In the plot, ρ denotes the spectral radius of the observer for θ, and τ the horizon of the prediction step [3]. Even by employing large prediction horizons, the approach proposed here outperformed [3] for this problem.

algorithms. In both cases, however, the prediction-correction method is outperformed by the approach in this work.

VI. Conclusions

We prove a fundamental result for time-varying optimization, which states that any algorithm that asymptotically tracks a critical trajectory must embed an internal model of the time variation. We exploited this result to provide a design procedure to construct algorithms for time-varying optimization. The proposed approach relies on an exponential observer to estimate the temporal variability of the problem, combined with an algorithm design that zeros the gradient. Possible extensions include the use of other observers to influence the properties of the resulting algorithm, and application of the methodology to structured time-varying problems arising from particular applications.

References

[1] A. Simonetto, E. Dall'Anese, S. Paternain, G. Leus, and G. B. Giannakis, "Time-varying convex optimization: Time-structured algorithms and applications," *Proceedings of the IEEE*, vol. 108, no. 11, pp. 2032–2048, 2020.

[2] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Machine Learning*, vol. 69, no. 2, pp. 169–192, 2007.

[3] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, "A class of prediction-correction methods for time-varying convex optimization," *IEEE Transactions on Signal Processing*, vol. 64, no. 17, pp. 4576–4591, 2016.

[4] T. Ni, X.-Z. Xie, and W.-Z. Gu, "Discrete-time Euler-smoothing methods for time-varying convex constrained optimization," *Journal of the Franklin Institute*, vol. 361, no. 10, p. 106898, 2024.

[5] A. Simonetto, E. Dall'Anese, J. Monteil, and A. Bernstein, "Personalized optimization with user's feedback," *Automatica*, vol. 131, p. 109767, 2021.

[6] G. Bianchin, J. Cortés, J. I. Poveda, and E. Dall'Anese, "Time-varying optimization of LTI systems via projected primal-dual gradient flows," *IEEE Transactions on Control of Network Systems*, vol. 9, no. 1, pp. 474–486, Mar. 2022.

[7] N. Bastianello, R. Carli, and S. Zampieri, "Internal model-based online optimization," *IEEE Transactions on Automatic Control*, vol. 69, no. 1, pp. 689–696, 2024.

[8] G. Bianchin and B. Van Scoy, "The internal model principle of time-varying optimization," *arXiv preprint*, Aug. 2024, arXiv:2407.08037.

[9] B. Castillo, S. Di Gennaro, S. Monaco, and D. Normand-Cyrot, "Nonlinear regulation for a class of discrete-time systems," *Systems & Control Letters*, vol. 20, no. 1, pp. 57–65, 1993.

[10] J. Carr, *Applications of centre manifold theory*. Springer-Verlag, 1981, vol. 35.

[11] S. Wiggins, *Ordinary Differential Equations: A Dynamical Point of View*. World Scientific, 2023.

[12] G. Osipenko, *Center Manifolds*. New York, NY: Springer New York, 2011, pp. 48–62.

[13] B. A. Francis and W. M. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, no. 5, pp. 457–465, 1976.

[14] E. Hazan, "Introduction to online convex optimization," *Foundations and Trends in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2016.

[15] E. Davison, "The robust control of a servomechanism problem for linear time-invariant multivariable systems," *IEEE Transactions on Automatic Control*, vol. 21, no. 1, pp. 25–34, 1976.

[16] R. K. Sundaram, *A first course in optimization theory*. Cambridge university press, 1996.

[17] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, "Prediction-correction interior-point method for time-varying convex optimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1973–1986, 2017.

[18] R. Raveendran, A. D. Mahindrakar, and U. Vaidya, "Fixed-time dynamical system approach for solving time-varying convex optimization problems," in *American Control Conference*, 2022, pp. 198–203.

[19] N. Bastianello, R. Carli, and S. Zampieri, "Internal model-based online optimization," *IEEE Transactions on Automatic Control*, vol. 69, no. 1, pp. 689–696, 2024.

[20] G. D. Birkhoff, *Dynamical systems*. American Mathematical Society: New York, 1927.

[21] W. Rudin, *Principles of Mathematical Analysis*, 3rd ed., ser. International Series in Pure and Applied Mathematics. McGraw-Hill, 1976.

[22] W. Lin and C. I. Byrnes, "Design of discrete-time nonlinear control systems via smooth feedback," *IEEE Transactions on Automatic Control*, vol. 39, no. 11, pp. 2340–2346, 1994.