SIAM J. OPTIM.
Vol. 28, No. 3, pp. 2654–2689

ANALYSIS OF OPTIMIZATION ALGORITHMS VIA INTEGRAL QUADRATIC CONSTRAINTS: NONSTRONGLY CONVEX PROBLEMS*

MAHYAR FAZLYAB†, ALEJANDRO RIBEIRO†, MANFRED MORARI†, AND VICTOR M. PRECIADO†

Abstract. In this paper, we develop a unified framework capable of certifying both exponential and subexponential convergence rates for a wide range of iterative first-order optimization algorithms. To this end, we construct a family of parameter-dependent nonquadratic Lyapunov functions that can generate convergence rates in addition to proving asymptotic convergence. Using integral quadratic constraints (IQCs) from robust control theory, we propose a linear matrix inequality (LMI) to guide the search for the parameters of the Lyapunov function in order to establish a rate bound. Based on this result, we develop a semidefinite programming (SDP) framework whose solution yields the best convergence rate that can be certified by the class of Lyapunov functions under consideration. We illustrate the utility of our results by analyzing the gradient method, proximal algorithms, and their accelerated variants for (strongly) convex problems. We also develop the continuous-time counterpart, whereby we analyze the gradient flow and the continuous-time limit of Nesterov's accelerated method.

Key words. convex optimization, first-order methods, Nesterov's accelerated method, proximal gradient methods, integral quadratic constraints, linear matrix inequality, semidefinite programming

AMS subject classifications. 90C22, 90C25, 90C30, 93C10, 93D99, 93C15

DOI. 10.1137/17M1136845

1. Introduction.

The analysis and design of iterative optimization algorithms is a well-established research area in optimization theory. Due to their computational efficiency and global convergence properties, first-order methods are of particular interest, especially in large-scale optimization arising in current machine learning applications. However, these algorithms can be very slow, even for moderately well-conditioned problems. In this direction, accelerated variants of first-order algorithms, such as Polyak's heavy-ball algorithm [25] or Nesterov's accelerated method [22], have been developed to speed up the convergence in ill-conditioned and nonstrongly convex problems.

In numerical optimization, convergence analysis is an integral part of algorithm tuning and design. This task, however, is often pursued on a case-by-case basis, and the analysis techniques heavily depend on the particular algorithm under study as well as the underlying assumptions. However, by interpreting iterative algorithms as feedback dynamical systems, it is possible to incorporate tools from control theory to analyze and design these algorithms in a more systematic and unified manner [15, 31, 12, 30]. Moreover, control techniques can be exploited to address more complex tasks, such as analyzing robustness against uncertainties, deriving nonconservative worst-

*Received by the editors June 30, 2017; accepted for publication (in revised form) June 20, 2018; published electronically September 20, 2018. Parts of this work appeared in conference paper [11].
http://www.siam.org/journals/siopt/28-3/M113684.html
Funding: This work was supported in part by the NSF under grants CAREER-ECCS-1651433 and IIS-1447470.
†Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 (mahyarfa@seas.upenn.edu, aribeiro@seas.upenn.edu, morari@seas.upenn.edu, preciado@seas.upenn.edu).

case bounds, and providing convergence guarantees under less restrictive assumptions [6, 15, 19].

A universal approach to analyzing the stability of dynamical systems is to construct a Lyapunov function that decreases along the trajectories of the system, proving asymptotic convergence. In the context of iterative optimization algorithms, it is of particular importance to certify a nonconservative rate bound in addition to proving asymptotic convergence. Construction of Lyapunov functions that can achieve this goal is not straightforward, especially for nonstrongly convex problems, in which the convergence rate is subexponential. It is important to note that in a considerable number of applications in machine learning, the underlying optimization problem is not strongly convex [3].

The goal of the present work is to develop a semidefinite programming (SDP) framework for the construction of Lyapunov functions that can characterize both exponential and subexponential convergence rates for iterative first-order optimization algorithms. The main pillars of our framework are time-varying Lyapunov functions, originally proposed in [27] for analyzing gradient-based momentum methods [32, 33], and integral quadratic constraints (IQCs) from robust control theory [34, 20], which have recently been adapted by Lessard, Recht, and Packard [19] in the context of optimization algorithms. Specifically, we propose a family of nonquadratic Lyapunov functions equipped with time-dependent parameters that can establish both exponential and subexponential convergence rates. We then develop a linear matrix inequality (LMI) to guide the search for the parameters of the Lyapunov function in order to generate analytical/numerical convergence rates. Based on this result, we formulate a semidefinite program to compute the fastest convergence rate that can be certified by the class of Lyapunov functions under consideration. In this semidefinite program, the properties of the objective function (e.g., convexity, Lipschitz continuity, etc.) can be systematically encoded into the program, providing a modular approach to obtaining convergence rates under various regularity assumptions, such as quasi-convexity [14], weak quasi-convexity [13], quasi-strong convexity [21], quadratic growth [21], and the Polyak–Łojasiewicz condition [17]. Furthermore, we extend our framework to continuous-time settings, in which we analyze the continuous-time limits (by taking infinitesimal stepsizes) of relevant iterative optimization algorithms. We will illustrate the generality of our framework by analyzing several first-order optimization algorithms, namely, unconstrained (accelerated) gradient methods, gradient methods with projection, and (accelerated) proximal methods.

Finally, we consider algorithm design. Specifically, we develop a robust counterpart of the developed LMI whose feasibility provides the algorithm with an additional stability margin in the sense of Lyapunov. As a design experiment, we use the LMI to tune the stepsize and momentum coefficient of Nesterov's accelerated method applied to strongly convex functions, considering robustness as a design criterion.

1.1. Related work. There is a host of results in the literature using semidefinite programs to analyze the convergence of first-order optimization algorithms [10, 29, 28, 18]. The first among them is [10], in which Drori and Teboulle developed a semidefinite program to derive analytical/numerical bounds on the worst-case performance of the unconstrained gradient method and its accelerated variant. An extension of this framework to the proximal gradient method—for the case of strongly convex problems—has been recently proposed in [28]. These SDP formulations, despite being able to yield new performance bounds, are highly algorithm dependent. Departing from classical algorithmic view, Lessard, Recht, and Packard [19] developed an SDP

framework based on quadratic Lyapunov functions and IQCs to derive sufficient conditions for exponential stability of an algorithm when the objective function is strongly convex [19, Theorem 4]. Specifically, they formulate a small semidefinite program whose feasibility verifies exponential convergence at a specified rate. It is important to note that Lessard's framework is specifically tailored to analyze strongly convex problems with exponential convergence [19, 24], and subexponential rates cannot be captured. Finally, another related work is that of Hu and Lessard [16], in which they proposed an LMI framework based on quadratic Lyapunov functions and dissipativity theory to analyze Nesterov's accelerated method. In contrast, the present work, inspired by [19], develops (1) an IQC framework using time-dependent nonquadratic Lyapunov functions for the analysis of a broader family of functionals, and (2) algorithms involving projections and proximal operators, including the proximal variant of Nesterov's method.

1.2. Notation and preliminaries. We denote the set of real numbers by $\mathbb{R}$, the set of real n-dimensional vectors by $\mathbb{R}^n$, the set of $m \times n$-dimensional matrices by $\mathbb{R}^{m\times n}$, and the n-dimensional identity matrix by I_n. We denote by $\mathbb{S}^n$, $\mathbb{S}^n_+$, and $\mathbb{S}^n_{++}$ the sets of n-by-n symmetric, positive semidefinite, and positive definite matrices, respectively. For $M \in \mathbb{R}^{n\times n}$ and $x \in \mathbb{R}^n$, we have that $x^\top Mx = \frac{1}{2}x^\top(M + M^\top)x$. The p-norm ($p \geq 1$) is displayed by $\|\cdot\|_p \colon \mathbb{R}^n \to \mathbb{R}_+$. For two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ of arbitrary dimensions, their Kronecker product is given by

$$A \otimes B = \begin{bmatrix} A_{11}B & \cdots & A_{1n}B \\ \vdots & \ddots & \vdots \\ A_{m1}B & \dots & A_{mn}B \end{bmatrix}.$$

Further, we have that $(A \otimes B)^\top = A^\top \otimes B^\top$ and $(AC) \otimes (BD) = (A \otimes B)(C \otimes D)$ for matrices of appropriate dimensions. Let $f \colon \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a closed proper function. The effective domain of f is denoted by $\operatorname{dom} f = \{x \in \mathbb{R}^n \colon f(x) < \infty\}$. The indicator function $\mathbb{I}_{\mathcal{X}} \colon \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ of a closed nonempty convex set $\mathcal{X} \subset \mathbb{R}^n$ is defined as $\mathbb{I}_{\mathcal{X}}(x) = 0$ if $x \in \mathcal{X}$ and as $\mathbb{I}_{\mathcal{X}}(x) = +\infty$ otherwise. The Euclidean projection of $x \in \mathbb{R}^n$ onto a set $\mathcal{X}$ is denoted by $[x]_{\mathcal{X}} = \operatorname{argmin}_{y\in\mathcal{X}} \|y - x\|_2$.

DEFINITION 1.1 (smoothness). *A differentiable function $f \colon \mathbb{R}^d \to \mathbb{R}$ is L_f-smooth on $\mathcal{S} \subseteq \operatorname{dom} f$ if*

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L_f \|x - y\|_2 \quad \text{for all } x, y \in \mathcal{S}. \tag{1.1}$$

Lipschitz continuity implies that

$$f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{L_f}{2}\|y - x\|_2^2 \quad \text{for all } x, y \in \mathcal{S}. \tag{1.2}$$

DEFINITION 1.2 (strong convexity). *A differentiable function $f \colon \mathbb{R}^d \to \mathbb{R}$ is m_f-strongly convex on $\mathcal{S} \subseteq \operatorname{dom} f$ if*

$$m_f\|x - y\|_2^2 \leq (x - y)^\top (\nabla f(x) - \nabla f(y)) \quad \text{for all } x, y \in \mathcal{S}. \tag{1.3}$$

An equivalent definition is that

$$f(x) + \nabla f(x)^\top (y - x) + \frac{m_f}{2}\|y - x\|_2^2 \leq f(y) \quad \text{for all } x, y \in \mathcal{S}. \tag{1.4}$$

We denote the class of L_f-smooth and m_f-strongly convex functions by $\mathcal{F}(m_f, L_f)$. Note that, by setting $m_f = 0$, we recover convex functions. For the class $\mathcal{F}(m_f, L_f)$, we denote the condition number by $\kappa_f = L_f/m_f \geq 1$.

2. Algorithm representation. Iterative algorithms can be represented as linear dynamical systems interacting with one or more static nonlinearities [19]. The linear part describes the algorithm itself, while the nonlinear components depend exclusively on the first-order oracle of the objective function. In this paper, we consider first-order algorithms that have the state-space representation

$$\begin{aligned} \xi_{k+1} &= A_k \xi_k + B_k u_k, \\ y_k &= C_k \xi_k, \\ u_k &= \phi(y_k), \\ x_k &= E_k \xi_k, \end{aligned} \tag{2.1}$$

where at each iteration index k, $\xi_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^d$ is the input ($d \leq n$), $y_k \in \mathbb{R}^d$ is the feedback output that is transformed by the nonlinear map $\phi\colon \mathbb{R}^d \to \mathbb{R}^d$ to generate u_k, and $x_k \in \mathbb{R}^d$ is the output at which the suboptimality will be evaluated for convergence analysis. See Figure 1 for a block diagram representation.[1]

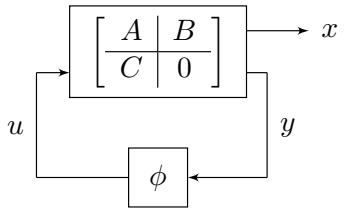

FIG. 1. *Block diagram representation of a first-order algorithm in state-space form.*

A broad family of first-order algorithms can be represented in the canonical form (2.1), where the matrices (A_k, B_k, C_k, E_k) differ for each algorithm. In this representation, the nonlinear feedback component ϕ depends on the oracle of the objective function. For instance, in unconstrained smooth minimization problems, we have that $\phi = \nabla f$, where f is the objective function. In composite optimization problems, ϕ is the generalized gradient mapping of the composite function, which we will describe in section 5. As an illustration, consider the following recursion defined on the two sequences $\{x_k\}$ and $\{y_k\}$:

$$\begin{aligned} x_{k+1} &= x_k + \beta_k(x_k - x_{k-1}) - h_k \nabla f(y_k), \\ y_k &= x_k + \gamma_k(x_k - x_{k-1}), \end{aligned} \tag{2.2}$$

where h_k, β_k, and γ_k are nonnegative scalars, $\{x_k\}$ is the primary sequence, and $\{y_k\}$ is the sequence at which the gradient is evaluated. By defining the state vector $\xi_k = [x_{k-1}^\top \; x_k^\top]^\top \in \mathbb{R}^{2d}$, we can represent (2.2) in the canonical form (2.1), where the matrices (A_k, B_k, C_k) are given by

$$\left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & 0 \end{array}\right] = \left[\begin{array}{cc|c} 0 & I_d & 0 \\ -\beta_k I_d & (\beta_k + 1) I_d & -h_k I_d \\ \hline -\gamma_k I_d & (\gamma_k + 1) I_d & 0 \end{array}\right]. \tag{2.3}$$

[1] Since the input $u = \phi(y)$ is an explicit function of the output, we set the feedforward matrix D to zero in the representation of the linear dynamics to ensure the explicit dependence of the feedback input on the output; i.e., the feedback system is well-posed.

Notice that, depending on the selection of β_k and γ_k, (2.2) describes various existing algorithms. For example, the gradient method corresponds to the case $\beta_k = \gamma_k = 0$. In Nesterov's accelerated method, we have $\beta_k = \gamma_k$. Finally, we recover the heavy-ball method by setting $\gamma_k = 0$.

For an algorithm represented in the canonical form (2.1), its fixed points (if they exist) are characterized by

$$\xi_\star = A_k\xi_\star + B_k u_\star, \quad y_\star = C_k\xi_\star, \quad u_\star = \phi(y_\star), \quad x_\star = E_k\xi_\star \quad \text{for all } k. \tag{2.4}$$

For well-designed algorithms, the fixed-point equation must coincide with the optimality conditions of the underlying optimization problem.

3. Main results. In this paper, we are concerned with the convergence analysis of first-order algorithms designed to solve optimization problems of the form

$$\mathcal{X}_\star = \operatorname{argmin}_{x\in\mathbb{R}^d}\{F(x) = f(x) + g(x)\}, \tag{3.1}$$

where $f\colon \mathbb{R}^d \to \mathbb{R}$ is closed, proper, and differentiable, while $g\colon \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ is closed convex proper (CCP) and possibly nondifferentiable. Depending on the choice of f and g, (3.1) describes various specialized optimization problems. For instance, when $g(x) = \mathbb{I}_{\mathcal{X}}(x)$ is the indicator function of a nonempty, closed, convex set $\mathcal{X} \subseteq \mathbb{R}^d$, (3.1) is equivalent to constrained smooth programming; when $g(x) \equiv 0$, we obtain unconstrained smooth programming; and, when $f(x) \equiv 0$, (3.1) simplifies to an unconstrained nonsmooth optimization problem. In all cases, we assume that the optimal solution set $\mathcal{X}_\star$ is nonempty and closed, and the optimal value $F_\star = \inf_{x\in\mathbb{R}^d} F(x)$ is finite.

Consider an iterative first-order algorithm represented in the state-space form (2.1) that, under appropriate initialization, solves (3.1) asymptotically; that is, the sequence of outputs $\{x_k\}$ satisfies $\lim_{k\to\infty} F(x_k) = F(x_\star)$, where $x_\star \in \mathcal{X}_\star$. We assume that the fixed point $y_\star$ of the sequence $\{y_k\}$, defined in (2.4), satisfies $y_\star = x_\star$. In other words, both $\{x_k\}$ and $\{y_k\}$ are convergent to the same optimal point $x_\star$. To establish a rate bound for the algorithm under study, we propose the Lyapunov function

$$V_k(x,\xi) = a_k(F(x) - F(x_\star)) + (\xi - \xi_\star)^\top P_k(\xi - \xi_\star), \tag{3.2}$$

where $a_k \geq 0$, $P_k \in \mathbb{S}^n_+$ for all k, and are to be determined. The first term is the suboptimality of x scaled by a_k, and the second term quantifies the suboptimality of the state ξ with respect to the optimal state $\xi_\star$. Notice that by this definition, we have that $V_k(x,\xi) \geq 0$ for all k, and $V_k(x_\star,\xi_\star) = 0$; i.e., the Lyapunov function is nonnegative everywhere and zero at optimality. Suppose we select $\{a_k\}$ and $\{P_k\}$ such that the Lyapunov function becomes nonincreasing along the trajectories of (2.1); i.e., the following condition holds:

$$V_{k+1}(x_{k+1},\xi_{k+1}) \leq V_k(x_k,\xi_k) \quad \text{for all } k. \tag{3.3}$$

Then, we can conclude that $a_k(F(x_k) - F(x_\star)) \leq V_k(x_k,\xi_k) \leq V_0(x_0,\xi_0)$ or, equivalently,

$$0 \leq F(x_k) - F(x_\star) \leq \frac{V_0(x_0,\xi_0)}{a_k} = \mathcal{O}\left(\frac{1}{a_k}\right) \quad \text{for all } k. \tag{3.4}$$

In other words, the sequence $\{a_k\}$ generates an upper bound on the suboptimality or, equivalently, a lower bound on the convergence rate. As a result, the task of certifying

a convergence rate for the algorithm translates into finding sufficient conditions to guarantee (3.3). In the following theorem, we develop an LMI whose feasibility is sufficient for (3.3) to hold.

THEOREM 3.1 (main result). *Let $x_\star \in \operatorname{argmin}_{x\in\mathbb{R}^d} F(x)$ be a minimizer of $F\colon \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ with a finite optimal value $F(x_\star)$. Consider an iterative first-order algorithm in the state-space form* (2.1).

1. *Suppose the fixed points $(\xi_\star, u_\star, y_\star, x_\star)$ of* (2.1) *satisfy*

$$\xi_\star = A_k\xi_\star + B_k u_\star, \quad y_\star = C_k\xi_\star, \quad u_\star = \phi(y_\star), \quad x_\star = E_k\xi_\star = y_\star \quad \text{for all } k. \tag{3.5}$$

2. *Suppose there exist symmetric matrices M_k^1, M_k^2, M_k^3 such that the following inequalities hold for all k:*

$$F(x_{k+1}) - F(x_k) \le e_k^\top M_k^1 e_k, \tag{3.6a}$$

$$F(x_{k+1}) - F(x_\star) \le e_k^\top M_k^2 e_k, \tag{3.6b}$$

$$0 \le e_k^\top M_k^3 e_k, \tag{3.6c}$$

where $e_k = [(\xi_k - \xi_\star)^\top \ (u_k - u_\star)^\top]^\top \in \mathbb{R}^{n+d}$ and M_k^3 is either zero or indefinite.

3. *Suppose there exist a nonnegative and nondecreasing sequence of reals $\{a_k\}$, a sequence of nonnegative reals $\{\sigma_k\}$, and a sequence of $n\times n$ positive semidefinite matrices $\{P_k\}$ satisfying*

$$M_k^0 + a_k M_k^1 + (a_{k+1} - a_k)M_k^2 + \sigma_k M_k^3 \preceq 0 \quad \text{for all } k, \tag{3.7}$$

where

$$M_k^0 = \begin{bmatrix} A_k^\top P_{k+1} A_k - P_k & A_k^\top P_{k+1} B_k \\ B_k^\top P_{k+1} A_k & B_k^\top P_{k+1} B_k \end{bmatrix}. \tag{3.8}$$

Then the sequence $\{x_k\}$ satisfies

$$F(x_k) - F(x_\star) \le \frac{a_0(F(x_0) - F(x_\star)) + (\xi_0 - \xi_\star)^\top P_0 (\xi_0 - \xi_\star)}{a_k} \quad \text{for all } k. \tag{3.9}$$

Before proving Theorem 3.1, we briefly discuss the assumptions made in the statement of the theorem. The first inequality in (3.6) bounds the difference between two consecutive iterates. In particular, if M_k^1 is negative semidefinite for all k, then the sequence $\{F(x_k)\}$ is monotone. The second inequality in (3.6) bounds the suboptimality, and finally, the third inequality in (3.6) is a quadratic constraint on the input-output pairs (ξ_k, u_k) that are related via the rule $u_k = \phi(C_k\xi_k)$. These bounds are required to satisfy condition (3.3) and will feature heavily throughout the paper. Note that the matrices (M_k^1, M_k^2, M_k^3) in (3.6) depend on the algorithm parameters, i.e., the matrices (A_k, B_k, C_k, E_k) that define the algorithm (see (2.1)), as well as on the assumptions about the objective function F.

Proof of Theorem 3.1. First, by (2.1) and (3.5), we can write

$$\xi_{k+1} - \xi_\star = A_k(\xi_k - \xi_\star) + B_k(u_k - u_\star).$$

Using the above identity, we can write

$$(\xi_{k+1} - \xi_\star)^\top P_{k+1}(\xi_{k+1} - \xi_\star) - (\xi_k - \xi_\star)^\top P_k(\xi_k - \xi_\star) = e_k^\top M_k^0 e_k. \tag{3.10a}$$

Multiply (3.6a) by a_k and (3.6b) by $(a_{k+1} - a_k)$ and add both sides of the resulting inequalities to obtain

$$a_{k+1}(F(x_{k+1}) - F(x_\star)) - a_k(F(x_k) - F(x_\star)) \leq 0. \tag{3.10b}$$

By adding both sides of the inequalities in (3.10) and recalling the definition of $V_k(x_k, \xi_k)$ in (3.2), we can write

$$V_{k+1}(x_{k+1}, \xi_{k+1}) - V_k(x_k, \xi_k) \leq e_k^\top \left(M_k^0 + a_k M_k^1 + (a_{k+1} - a_k) M_k^2 \right) e_k. \tag{3.11}$$

Suppose the matrix inequality in (3.7) holds. By multiplying this inequality from the left and right by $e_k^\top$ and e_k, respectively, we obtain

$$e_k^\top \left(M_k^0 + a_k M_k^1 + (a_{k+1} - a_k) M_k^2 + \sigma_k M_k^3 \right) e_k \leq 0. \tag{3.12}$$

Finally, adding both sides of (3.11) and (3.12) yields

$$V_{k+1}(x_{k+1}, \xi_{k+1}) - V_k(x_k, \xi_k) \leq -\sigma_k e_k^\top M_k^3 e_k \leq 0, \tag{3.13}$$

where the second inequality follows from (3.6c). Hence, the sequence $\{V_k(x_k, \xi_k)\}$ is nonincreasing, implying $a_k(F(x_k) - F(x_\star)) \leq V_k(x_k, \xi_k) \leq V_0(x_0, \xi_0)$. The proof becomes complete by dividing both sides of the last inequality by a_k. ∎

Some remarks are in order regarding Theorem 3.1.

1. We do not make the assumption that the algorithm under consideration is a descent method. In other words, the sequence $\{F(x_k)\}$ of function values is not necessarily monotone, which is a hallmark of accelerated algorithms [23]. In contrast, we require the sequence $\{V_k(x_k, \xi_k)\}$ of "energy" values to be monotonically decreasing. From this perspective, the LMI (3.7) provides a guideline for the construction energy functions with this property.
2. There is no restriction on the sequence $\{a_k\}$ other than nonnegativity and monotonicity. Hence, we can characterize both exponential ($a_k = \rho^{-k}, 0 \leq \rho < 1$) and subexponential ($a_k = k^p,\ p > 0$, for example) convergence rates.
3. We have made no explicit assumptions about the objective function in Theorem 3.1 other than the quadratic bounds in (3.6). In fact, the matrices M_k^1, M_k^2, M_k^3 that characterize these bounds depend on the parameters of the algorithm (e.g., stepsize, momentum coefficient, etc.) and on the assumptions about F. In sections 4 and 5, we will describe a general procedure for deriving these matrices for a wide range of algorithms and assumptions.

3.1. Time-invariant algorithms with exponential convergence. In this subsection, we specialize the results of Theorem 3.1 to time-invariant algorithms with exponential convergence. Under these assumptions, we can precondition a_k and P_k to simplify the LMI in (3.7). Explicitly, suppose the matrices (A_k, B_k, C_k, E_k) that define the algorithm do not change with k. By the particular selection

$$a_k = \rho^{-2k} a_0, \quad a_0 > 0, \quad P_k = \rho^{-2k} P_0, \quad P_0 \succeq 0, \quad 0 < \rho \leq 1 \quad \text{for all } k, \tag{3.14}$$

the Lyapunov function in (3.2) reads

$$V_k(\xi) = \rho^{-2k} \left(a_0(F(x) - F(x_\star)) + (\xi - \xi_\star)^\top P_0 (\xi - \xi_\star) \right). \tag{3.15}$$

The unknown parameters of the Lyapunov function are now $a_0 > 0, P_0 \succeq 0$, and $0 < \rho \leq 1$. With this parameter selection, the LMI in (3.7) simplifies greatly. The following result is a special case of Theorem 3.1 for the selection (3.14).

THEOREM 3.2 (exponential convergence of time-invariant algorithms). *In Theorem* 3.1*, assume that the algorithm parameters as well as the matrices* M_k^1, M_k^2, M_k^3 *in* (3.6) *do not change with* k*. In other words,*

$$(A_k, B_k, C_k, E_k, M_k^1, M_k^2, M_k^3) = (A_0, B_0, C_0, E_0, M_0^1, M_0^2, M_0^3) \quad \text{for all } k.$$

Suppose there exists $a_0 > 0$*,* $P_0 \in \mathbb{S}_+^n$*, and* $\lambda_0 \geq 0$ *that satisfy*

$$\begin{bmatrix} A_0^\top P_0 A_0 - \rho^2 P_0 & A_0^\top P_0 B_0 \\ B_0^\top P_0 A_0 & B_0^\top P_0 B_0 \end{bmatrix} + a_0\rho^2 M_0^1 + a_0(1-\rho^2)M_0^2 + \lambda_0 M_0^3 \preceq 0 \tag{3.16}$$

for some $0 < \rho \leq 1$*. Then the sequence* $\{x_k\}$ *satisfies*

$$F(x_k) - F(x_\star) \leq \frac{a_0(F(x_0)-F(x_\star)) + (\xi_0 - \xi_\star)^\top P_0(\xi_0 - \xi_\star)}{a_0}\rho^{2k}. \tag{3.17}$$

Proof. By substituting the parameter selection (3.14) in (3.7) and factoring out the positive term ρ^{-2k-2} from the resulting LMI, we obtain (3.16), which no longer depends on k. Utilizing Theorem 3.1, the feasibility of (3.16) ensures (3.3), which in turn implies (3.17). The proof is complete. ∎

Remark 1. Regarding the parameter selection in (3.14), if we instead select $a_k \equiv 0$, $P_k = \rho^{-2k}P_0$ with $P_0 \succ 0$, and $0 < \rho \leq 1$, the Lyapunov function (3.2) simplifies to the quadratic function

$$V_k(\xi) = \rho^{-2k}(\xi - \xi_\star)^\top P_0(\xi - \xi_\star), \quad P_0 \succ 0. \tag{3.18}$$

Correspondingly, the LMI (3.16) in Theorem 3.2 reduces to

$$\begin{bmatrix} A_0^\top P_0 A_0 - \rho^2 P_0 & A_0^\top P_0 B_0 \\ B_0^\top P_0 A_0 & B_0^\top P_0 B_0 \end{bmatrix} + \lambda_0 M_0^3 \preceq 0. \tag{3.19}$$

By Theorem 3.1, if (3.19) is feasible for some $P_0 \succ 0, \lambda_0 \geq 0$, and $0 < \rho \leq 1$, then the Lyapunov function in (3.18) satisfies $V_{k+1}(\xi_{k+1}) \leq V_k(\xi_k)$, which translates to

$$(\xi_{k+1} - \xi_\star)^\top P_0(\xi_{k+1} - \xi_\star) \leq \rho^2(\xi_k - \xi_\star)^\top P_0(\xi_k - \xi_\star)$$

or, equivalently,

$$\|\xi_k - \xi_\star\|_2^2 \leq \rho^{2k}\mathrm{cond}(P_0)\|\xi_0 - \xi_\star\|_2^2. \tag{3.20}$$

The matrix inequality (3.19) is precisely the condition derived in [19, Theorem 4] for the case of strongly convex objective functions, time-invariant first-order algorithms, and *pointwise* IQCs.

Having established the main result, it now remains to determine the matrices M_k^i, $i \in \{0, 1, 2, 3\}$, that construct the LMI in (3.7). To this end, we first need to introduce IQCs in the context of optimization algorithms.

3.2. IQCs for optimization algorithms. In control theory, there are various approaches and criteria for stability of linear dynamical systems in feedback interconnection with a memoryless and possibly time-varying nonlinearity. In this context, IQCs, originally proposed by Megretski and Rantzer [20], are a powerful tool for describing various classes of nonlinearities, and are particularly useful for LMI-based stability analysis. Lessard, Recht, and Packard [19] have recently adapted the theory of IQCs for use in optimization algorithms. Specifically, they translate the first-order defining properties of convex functions into various forms of IQCs for their gradient mappings. In the following, we briefly describe the notion of pointwise IQCs (or quadratic constraints) that will be essential for subsequent developments.

3.2.1. Pointwise IQCs. Consider a mapping $\phi\colon \mathbb{R}^d \to \mathbb{R}^d$ and a chosen "reference" input-output pair[2] $(x_\star, \phi(x_\star))$, $x_\star \in \operatorname{dom}\phi$. We say that ϕ satisfies the *pointwise* IQC defined by $(Q_\phi, x_\star, \phi(x_\star))$ on $\mathcal{S} \subseteq \operatorname{dom}\phi$ if for all $x \in \mathcal{S}$, the following inequality holds [19]:

$$\begin{bmatrix} x - x_\star \\ \phi(x) - \phi(x_\star) \end{bmatrix}^\top Q_\phi \begin{bmatrix} x - x_\star \\ \phi(x) - \phi(x_\star) \end{bmatrix} \geq 0, \tag{3.21}$$

where $Q_\phi \in \mathbb{S}^{2d}$ is a symmetric, indefinite matrix.[3] Many inequalities in optimization can be represented as IQCs of the form (3.21). For instance, suppose $\phi(x)$ is L_ϕ-Lipschitz continuous on $\mathcal{S} \subseteq \operatorname{dom}\phi$ for some positive and finite L_ϕ, i.e., $\|\phi(x) - \phi(x_\star)\|_2 \leq L_\phi \|x - x_\star\|_2$ for all $(x, x_\star) \in \mathcal{S} \times \mathcal{S}$. By squaring both sides and rearranging terms, we obtain

$$\begin{bmatrix} x - x_\star \\ \phi(x) - \phi(x_\star) \end{bmatrix}^\top \begin{bmatrix} L_\phi^2 I_d & 0 \\ 0 & -I_d \end{bmatrix} \begin{bmatrix} x - x_\star \\ \phi(x) - \phi(x_\star) \end{bmatrix} \geq 0, \tag{3.22}$$

which equivalently describes Lipschitz continuity. As another example, assume ϕ is a *firmly nonexpansive mapping* on $\mathcal{S}$. That is, for all $(x, x_\star) \in \mathcal{S} \times \mathcal{S}$, we have that $\|\phi(x) - \phi(x_\star)\|_2^2 \leq (x - x_\star)^\top (\phi(x) - \phi(x_\star))$. This inequality can be rewritten as

$$\begin{bmatrix} x - x_\star \\ \phi(x) - \phi(x_\star) \end{bmatrix}^\top \begin{bmatrix} 0 & \frac{1}{2} I_d \\ \frac{1}{2} I_d & -I_d \end{bmatrix} \begin{bmatrix} x - x_\star \\ \phi(x) - \phi(x_\star) \end{bmatrix} \geq 0. \tag{3.23}$$

Note that by the Cauchy–Schwarz inequality, firm nonexpansiveness implies Lipschitz continuity with Lipschitz parameter equal to one; i.e., (3.23) implies (3.22) with $L_\phi = 1$. There are many other interesting properties such as monotonicity (also known as incremental passivity), one-sided Lipschitz continuity, cocoercivity, etc., that could be represented by quadratic constraints.

In the next subsection, we will focus on the gradient mapping of a convex function from an IQC perspective.

3.2.2. IQCs for (strongly) convex functions. Consider the gradient mapping $\phi = \nabla f$, where $f \in \mathcal{F}(m_f, L_f)$. It directly follows from the definition of (strong) convexity in (1.3) that ∇f satisfies the quadratic constraint

$$\begin{bmatrix} x - y \\ \nabla f(x) - \nabla f(y) \end{bmatrix}^\top \begin{bmatrix} -m_f I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} x - y \\ \nabla f(x) - \nabla f(y) \end{bmatrix} \geq 0. \tag{3.24}$$

Similarly, the Lipschitz inequality in (1.1) can be represented as

$$\begin{bmatrix} x - y \\ \nabla f(x) - \nabla f(y) \end{bmatrix}^\top \begin{bmatrix} L_f^2 I_d & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x - y \\ \nabla f(x) - \nabla f(y) \end{bmatrix} \geq 0. \tag{3.25}$$

To combine strong convexity and Lipschitz continuity in a single inequality, we note that ∇f also satisfies [23]

$$\frac{m_f L_f}{m_f + L_f} \|y - x\|_2^2 + \frac{1}{m_f + L_f} \|\nabla f(y) - \nabla f(x)\|_2^2 \leq (\nabla f(y) - \nabla f(x))^\top (y - x). \tag{3.26}$$

[2] As we will see later, the reference point is chosen as the fixed point of the interconnected system we wish to analyze.

[3] If Q_ϕ is positive (semi)definite, the quadratic constraint holds trivially and is not informative regarding ϕ.

The above inequality can be represented by the following quadratic constraint [19]:

$$\begin{bmatrix} x-y \\ \nabla f(x)-\nabla f(y) \end{bmatrix}^\top Q_f \begin{bmatrix} x-y \\ \nabla f(x)-\nabla f(y) \end{bmatrix} \geq 0, \quad Q_f = \begin{bmatrix} \frac{-m_f L_f}{m_f+L_f} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & \frac{-1}{m_f+L_f} I_d \end{bmatrix}. \tag{3.27}$$

In the language of IQCs, we can say that the map $\phi = \nabla f$ satisfies the pointwise IQC defined by $(Q_f, x_\star, \nabla f(x_\star))$, where the reference point $x_\star = y \in \mathcal{S}$ is arbitrary. Note that (3.27) encapsulates both (strong) convexity and Lipschitz continuity in a single IQC. It turns out that this quadratic constraint is both necessary and sufficient for the inclusion $f \in \mathcal{F}(m_f, L_f)$.

3.2.3. Nondifferentiable convex functions. The above analysis can be extended to nondifferentiable convex functions. Formally, the subdifferential ∂f of a convex function $f\colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ is defined as

$$\partial f(x) = \{\gamma\colon \gamma^\top (y-x) + f(x) \leq f(y) \text{ for all } y \in \operatorname{dom} f\}, \tag{3.28}$$

where γ is any subgradient of f, which we denote by $T_f(x)$. Adding the inequality in (3.28) to the same inequality but with x and y interchanged, we obtain

$$(T_f(x) - T_f(y))^\top (x-y) \geq 0,$$

which is equivalent to monotonicity of the subdifferential operator. Therefore, any subgradient of f satisfies (3.27) with $L_f = \infty$. Note that this property holds even when f is not convex.

4. Performance results for unconstrained smooth programming. In this section, we consider first-order algorithms designed to solve problems of the form

$$x_\star \in \operatorname{argmin}_{x\in\mathbb{R}^d} f(x) \quad \text{where} \quad f \in \mathcal{F}(m_f, L_f). \tag{4.1}$$

The well-known optimality condition in this case is

$$\mathcal{X}_\star = \{x_\star \in \operatorname{dom} f\colon \nabla f(x_\star) = 0\}.$$

We now consider an iterative first-order algorithm in the canonical form (2.1) for solving (4.1), where the feedback nonlinearity is given by $\phi = \nabla f$. Since the sequences $\{x_k\}$ and $\{y_k\}$ converge to the same fixed point in the optimal set by assumption, we must have that $\nabla f(y_\star) = \nabla f(x_\star) = 0$. In other words, the fixed points of (2.1) satisfy

$$\xi_\star = A_k \xi_\star \quad y_\star = C_k \xi_\star \quad u_\star = \nabla f(y_\star) = 0, \quad x_\star = E_k \xi_\star = y_\star \quad \text{for all } k. \tag{4.2}$$

In the following result, we characterize the quadratic bounds in (3.6) for the class $\mathcal{F}(m_f, L_f)$.

LEMMA 4.1. *Let $x_\star \in \operatorname{argmin}_{x\in\mathbb{R}^d} f(x)$ be a minimizer of $f \in \mathcal{F}(m_f, L_f)$ with a finite optimal value $f(x_\star)$. Consider an iterative first-order algorithm in the state-space form* (2.1) *with $\phi = \nabla f$, where the fixed points $(\xi_\star, u_\star, y_\star, x_\star)$ satisfy*

$$\xi_\star = A_k \xi_\star, \quad y_\star = C_k \xi_\star, \quad u_\star = \nabla f(y_\star) = 0, \quad x_\star = E_k \xi_\star = y_\star \quad \textit{for all } k. \tag{4.3}$$

Define $e_k = [(\xi_k - \xi_\star)^\top \ (u_k - u_\star)^\top]^\top$. Then the following inequalities hold for all k:

$$f(x_{k+1}) - f(x_k) \leq e_k^\top M_k^1 e_k, \tag{4.4a}$$

$$f(x_{k+1}) - f(x_\star) \leq e_k^\top M_k^2 e_k, \tag{4.4b}$$

$$0 \leq e_k^\top M_k^3 e_k, \tag{4.4c}$$

where M_k^1, M_k^2, M_k^3 *are given by*

$$M_k^1 = N_k^1 + N_k^2, \quad M_k^2 = N_k^1 + N_k^3, \quad M_k^3 = N_k^4, \tag{4.5}$$

with

$$N_k^1 = \begin{bmatrix} E_{k+1}A_k - C_k & E_{k+1}B_k \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} \frac{L_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} E_{k+1}A_k - C_k & E_{k+1}B_k \\ 0 & I_d \end{bmatrix},$$

$$N_k^2 = \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix},$$

$$N_k^3 = \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix},$$

$$N_k^4 = \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} \frac{-m_f L_f}{m_f + L_f} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & \frac{-1}{m_f + L_f} I_d \end{bmatrix} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}.$$

Proof. First, by Lipschitz continuity of ∇f, we can write

$$f(x_{k+1}) - f(y_k) \leq \begin{bmatrix} x_{k+1} - y_k \\ \nabla f(y_k) \end{bmatrix}^\top \begin{bmatrix} \frac{L_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} x_{k+1} - y_k \\ \nabla f(y_k) \end{bmatrix}. \tag{4.6}$$

From the recursion in (2.1), we have that

$$\begin{bmatrix} x_{k+1} - y_k \\ \nabla f(y_k) \end{bmatrix} = \begin{bmatrix} E_{k+1}A_k - C_k & E_{k+1}B_k \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix}. \tag{4.7}$$

Substituting (4.7) in (4.6) yields

$$f(x_{k+1}) - f(y_k) \leq e_k^\top N_k^1 e_k. \tag{4.8}$$

Next, we use (strong) convexity and the identity $y_k - x_k = (C_k - E_k)(\xi_k - \xi_\star)$ to write

$$\begin{aligned} f(y_k) - f(x_k) &\leq \begin{bmatrix} y_k - x_k \\ \nabla f(y_k) \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} y_k - x_k \\ \nabla f(y_k) \end{bmatrix} \\ &\leq e_k^\top \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix} e_k \\ &= e_k^\top N_k^2 e_k. \end{aligned} \tag{4.9}$$

Adding both sides of (4.8) and (4.9) yields

$$f(x_{k+1}) - f(x_k) \leq e_k^\top (N_k^1 + N_k^2) e_k = e_k^\top M_k^1 e_k.$$

By (strong) convexity and the identity $y_k - y_\star = C_k(\xi_k - \xi_\star)$, we can write

$$\begin{aligned} f(y_k) - f(y_\star) &\leq \begin{bmatrix} y_k - y_\star \\ \nabla f(y_k) \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} y_k - y_\star \\ \nabla f(y_k) \end{bmatrix} \\ &= e_k^\top \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & 0 \end{bmatrix} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix} e_k \\ &= e_k^\top N_k^3 e_k. \end{aligned} \tag{4.10}$$

By adding both sides of (4.8) and (4.10), we obtain

$$f(x_{k+1}) - f(x_\star) \leq e_k^\top (N_k^1 + N_k^3) e_k = e_k^\top M_k^2 e_k.$$

Finally, since $f \in \mathcal{F}(m_f, L_f)$, the gradient function ∇f satisfies the IQC in (3.27). Since $y_k - y_\star = C_k(\xi_k - \xi_\star)$, we can write

(4.11)
$$e_k^\top N_k^4 e_k = e_k^\top \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top, \; Q_f \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}, \; e_k = \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix}^\top, \; Q_f \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix} \geq 0.$$

The proof is now complete. ∎

In Lemma 4.1, we used Lipschitz continuity and strong convexity assumptions to find the matrices in (4.4). Explicitly, N_k^1 follows from Lipschitz continuity, while N_k^2 and N_k^3 are due to strong convexity. Finally, the matrix $M_k^3 = N_k^4$ describes the quadratic constraint between the input-output pairs (ξ_k, u_k) that are related via $u_k = \nabla f(C_k \xi_k)$. Note that $M_k^3 = N_k^4$ is an indefinite matrix as required.

Remark 2 (exploiting block diagonal structure). We can often exploit some special structure in the data matrices (A_k, B_k, C_k, E_k) to reduce the dimension of the LMI (3.7). For many algorithms, the matrices (A_k, B_k, C_k, E_k) are in the form $(A_k = \bar{A}_k \otimes I_d, B_k = \bar{B}_k \otimes I_d, C_k = \bar{C}_k \otimes I_d, E_k = \bar{E}_k \otimes I_d)$, where $(\bar{A}_k, \bar{B}_k, \bar{C}_k, \bar{E}_k)$ are lower dimensional matrices independent of d [19, section 4.2]. By selecting $P_k = \bar{P}_k \otimes I_d$, where $\bar{P}_k$ is a lower dimensional matrix, we can factor out all the Kronecker products $\otimes I_d$ from the matrices $M_k^0, M_k^1, M_k^2, M_k^3$ and make the dimension of the corresponding LMI (3.7) independent of d. In particular, a multistep method with $r \geq 1$ steps yields an $(r+1) \times (r+1)$ LMI. For instance, the gradient method ($r = 1$) and the Nesterov's accelerated method ($r = 2$) yield 2×2 and 3×3 LMIs, respectively. We will use this dimensionality reduction in the forthcoming case studies.

We can now use Lemma 4.1 in tandem with Theorem 3.1 to derive convergence rates for some existing algorithms in the literature.

4.1. Symbolic rate bounds. In order to certify a convergence rate for a given algorithm, we must first represent the algorithm in the canonical form (2.1) and obtain the matrices M_k^1, M_k^2, M_k^3 that characterize the bounds in (3.6). These matrices are provided in Lemma 4.1 for the case $f \in \mathcal{F}(m_f, L_f)$. Then we must formulate the LMI (3.7) and search for a feasible triple (a_k, P_k, σ_k). In view of (3.4), we seek to find the fastest convergence rate, i.e., the fastest growing $\{a_k\}$. In what follows, we illustrate this approach via analyzing the gradient method and Nesterov's accelerated algorithm.

4.1.1. The gradient method. Consider the gradient method applied to $f \in \mathcal{F}(m_f, L_f)$ with constant stepsize:

$$x_{k+1} = x_k - h \nabla f(x_k). \tag{4.12}$$

This recursion corresponds to the the state-space form (2.1) with $(A_k, B_k, C_k, E_k) = (I_d, -hI_d, I_d, I_d)$. By choosing $P_k = p_k I_d$ ($p_k \geq 0$), we can apply the dimensionality reduction outlined in Remark 2 and reduce the dimension of the LMI. After

dimensionality reduction, the matrices $M_k^i,\ i \in \{0,1,2,3\}$, in the LMI (3.7) read

$$
\begin{aligned}
M_k^0 &= \begin{bmatrix} p_{k+1}-p_k & -hp_{k+1} \\ -hp_{k+1} & h^2 p_{k+1} \end{bmatrix}, \\
M_k^1 &= \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{2}(L_f h^2 - 2h) \end{bmatrix}, \\
M_k^2 &= \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2}(L_f h^2 - 2h) \end{bmatrix}, \\
M_k^3 &= \begin{bmatrix} \frac{-m_f L_f}{m_f + L_f} & \frac{1}{2} \\ \frac{1}{2} & \frac{-1}{m_f + L_f} \end{bmatrix}.
\end{aligned} \tag{4.13}
$$

We first consider strongly convex functions ($m_f > 0$) for which we make two parameter selections as follows.

- By setting $p_k = \sigma_k = 0$, we obtain the LMI

$$
\begin{bmatrix} -\frac{m_f}{2}(a_{k+1} - a_k) & \frac{1}{2}(a_{k+1} - a_k) \\ \frac{1}{2}(a_{k+1} - a_k) & (\frac{L_f h^2}{2} - h)a_{k+1} \end{bmatrix} \preceq 0 \quad \text{for all } k.
$$

  It is easy to verify that this matrix inequality is equivalent to the conditions $a_{k+1} \le \rho^{-1} a_k$ and $0 \le h \le 2/L_f$, where $\rho = 1 + m_f(L_f h^2 - 2h)$. Solving for a_k and substituting all the parameters in (3.3), we conclude the following convergence rate for strongly convex functions:

$$
f(x_k) - f(x_\star) \le \big(1 + m_f(L_f h^2 - 2h)\big)^k \, (f(x_0) - f(x_\star)), \quad 0 \le h \le \frac{2}{L_f}.
$$

  Notice that the decay rate ρ obeys $0 \le \rho \le 1$ as h varies on $[0, 2/L_f]$. In particular, by optimizing ρ over h, we obtain the optimal stepsize $h = 1/L_f$, yielding the decay rate $\rho = 1 - m_f/L_f$.
- By the parameter selection $a_k \equiv 0$ and $p_k = \rho^{-2k} p_0$, $\sigma_k = \lambda_0 \rho^{-2k-2}$, the LMI simplifies to

$$
\begin{bmatrix} p_0 - \rho^2 p_0 & -hp_0 \\ -hp_0 & h^2 p_0 \end{bmatrix} + \lambda_0 \begin{bmatrix} \frac{-m_f L_f}{m_f + L_f} & \frac{1}{2} \\ \frac{1}{2} & \frac{-1}{m_f + L_f} \end{bmatrix} \preceq 0, \tag{4.14}
$$

  which is the same LMI as the one proposed in [19] and yields the decay rate $\rho = \max(|1 - hm_f|, |1 - hL_f|)$.

We now consider convex functions ($m_f = 0$). By the particular selections $p_k = p$ and $\sigma_k = \sigma$, the LMI (3.7) reduces to

$$
\begin{bmatrix} 0 & \frac{1}{2}(a_{k+1} - a_k - 2ph + \sigma) \\ \frac{1}{2}(a_{k+1} - a_k - 2ph + \sigma) & (\frac{L_f h^2}{2} - h)a_{k+1} + ph^2 - \frac{\sigma}{L_f} \end{bmatrix} \preceq 0 \quad \text{for all } k, \tag{4.15}
$$

which is homogeneous in $(a_k, a_{k+1}, p, \sigma)$. We can therefore assume $p = 1$ without loss of generality. With these selections, the above LMI becomes equivalent to the following inequalities:

$$
a_{k+1} = a_k + 2h - \sigma, \quad \left(\frac{L_f h^2}{2} - h\right) a_{k+1} + h^2 - \frac{\sigma}{L_f} \le 0 \quad \text{for all } k.
$$

Assuming $a_0 = 0$ and solving for the fastest growing a_k that satisfies the above constraints, we obtain the rate bound

$$f(x_k) - f(x_\star) \leq \frac{L_f \|x_0 - x_\star\|_2^2}{Ck}, \tag{4.16a}$$

where C is given by

$$C = \begin{cases} 2L_f h & \text{for } 0 \leq L_f h \leq 1, \\ \dfrac{2(L_f h)^2(2 - L_f h)}{(L_f h)^2 - 2L_f h + 2} & \text{for } 1 \leq L_f h \leq 2. \end{cases} \tag{4.16b}$$

We have provided the detailed derivations in Appendix A. We observe that we can characterize both the exponential and subexponential rates using the same LMI.

4.1.2. Nesterov's accelerated method. We now analyze Nesterov's accelerated method [22] applied to $f \in \mathcal{F}(m_f, L_f)$, which consists of the following recursions:

$$\begin{aligned} x_{k+1} &= y_k - h\nabla f(y_k), \\ y_k &= x_k + \beta_k(x_k - x_{k-1}), \end{aligned} \tag{4.17}$$

where $\beta_k \geq 0$ is the momentum coefficient and $h > 0$ is the step size. With an appropriate tuning, this method exhibits an $\mathcal{O}(1/k^2)$ convergence rate when $m_f = 0$.

One such tuning is [22, 3]

$$\beta_k = t_k^{-1}(t_{k-1} - 1), \quad t_k = \frac{1}{2}\left(1 + \sqrt{1 + 4t_{k-1}^2}\right), \quad t_{-1} = 1, \quad 0 < h \leq L_f^{-1}. \tag{4.18}$$

Notice that by this selection, we can verify that $t_k^2 - t_{k-1}^2 = t_k$ and $t_{k-1} \geq (k+2)/2$. By defining the state vector $\xi_k = [x_{k-1}^\top \; x_k^\top]^\top$, we can write (4.17) in the canonical form

$$\begin{aligned} \xi_{k+1} &= \begin{bmatrix} 0 & I_d \\ -\beta_k I_d & (1+\beta_k) I_d \end{bmatrix} \xi_k + \begin{bmatrix} 0 \\ -hI_d \end{bmatrix} \nabla f(y_k), \\ y_k &= \begin{bmatrix} -\beta_k & (1+\beta_k) I_d \end{bmatrix} \xi_k, \\ x_k &= \begin{bmatrix} 0 & 1 \end{bmatrix} \xi_k. \end{aligned} \tag{4.19}$$

The fixed points of (4.19) are $(\xi_\star, u_\star, y_\star, x_\star) = ([x_\star^\top \; x_\star^\top]^\top, 0, x_\star, x_\star)$, where $x_\star \in \mathcal{X}_\star$ is any optimal solution to (4.1). Making use of Lemma 4.1, the matrices M_k^i $i \in \{0, 1, 2, 3\}$ for Nesterov's accelerated method read

$$\begin{aligned} M_k^0 &= \begin{bmatrix} A_k^\top P_{k+1} A_k - P_k & A_k^\top P_{k+1} B_k \\ B_k^\top P_{k+1} A_k & B_k^\top P_{k+1} B_k \end{bmatrix}, \\ M_k^1 &= \begin{bmatrix} -\frac{1}{2} m_f \beta_k^2 & \frac{1}{2} m_f \beta_k^2 & -\frac{1}{2}\beta_k \\ \frac{1}{2} m_f \beta_k^2 & -\frac{1}{2} m_f \beta_k^2 & \frac{1}{2}\beta_k \\ -\frac{1}{2}\beta_k & \frac{1}{2}\beta_k & \frac{1}{2} L_f h^2 - h \end{bmatrix}, \\ M_k^2 &= \begin{bmatrix} -\frac{1}{2} m_f \beta_k^2 & \frac{1}{2} m_f \beta_k(\beta_k + 1) & -\frac{1}{2}\beta_k \\ \frac{1}{2} m_f \beta_k(\beta_k + 1) & -\frac{1}{2} m_f (\beta_k + 1)^2 & \frac{1}{2}(\beta_k + 1) \\ -\frac{1}{2}\beta_k & \frac{1}{2}(\beta_k + 1) & \frac{1}{2} L_f h^2 - h \end{bmatrix}, \\ M_k^3 &= \begin{bmatrix} -\beta_k I_d & 0 \\ (1+\beta_k) I_d & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \frac{-m_f L_f}{m_f + L_f} & \frac{1}{2} \\ \frac{1}{2} & \frac{-1}{m_f + L_f} \end{bmatrix} \begin{bmatrix} -\beta_k I_d & (1+\beta_k) I_d & 0 \\ 0 & 0 & I_d \end{bmatrix}. \end{aligned} \tag{4.20}$$

We now consider convex settings ($m_f = 0$). It is straightforward to verify that for the parameter selection $\sigma_k = 0$, $a_k = t_{k-1}^2$ (with $a_0 = 1$), and

$$P_k = \frac{1}{2h}\begin{bmatrix} 1 - t_{k-1} \\ t_{k-1} \end{bmatrix}\begin{bmatrix} 1 - t_{k-1} & t_{k-1} \end{bmatrix},$$

the LMI (3.7) holds with equality, i.e., all the entries of the matrix is zero. Therefore, Theorem 3.1 implies

$$f(x_k) - f(x_\star) \leq \frac{f(x_0) - f(x_\star) + \frac{1}{2h}\|x_0 - x_\star\|_2^2}{t_{k-1}^2} = \mathcal{O}\left(\frac{1}{k^2}\right), \tag{4.21}$$

where the equality follows from the fact that $t_{k-1} \geq (k+2)/2$.

The analysis of Nesterov's method shows that finding a symbolic feasible pair (a_k, P_k) to the LMI (3.7) can be subtle. Nevertheless, we can also search for these parameters via a numerical scheme, as we describe next.

4.2. Numerical bounds for exponential rates. We can also use the results of Theorem 3.1 to search for the parameters (a_k, P_k) numerically. This approach is particularly efficient for time-invariant algorithms with exponential convergence. Under these assumptions, the sequence of LMIs in (3.7) collapses into the single LMI in (3.16), which no longer depends on the iteration index k. We can then use this LMI to find the exponential decay rate numerically. Explicitly, the matrix inequality (3.16) is an LMI in (a_0, P_0, λ_0) for a fixed ρ^2. We can therefore use a bisection search to find the smallest value of the convergence rate ρ that satisfies (3.16) for some (a_0, P_0, λ_0). Notice that the LMI in (3.16) is homogeneous in its decision variables. We can therefore assume $\lambda_0 = 1$ without loss of generality.

4.2.1. Nesterov's accelerated method (strong convexity). In Nesterov's accelerated method applied to strongly convex problems ($m_f > 0$), the momentum parameter does not change with k but may depend on the condition number κ_f. Nesterov proposed the following parameter selection for the algorithm in (4.17) and the corresponding analytical rate bound [23]:

$$h = \frac{1}{L_f}, \quad \beta = \frac{\sqrt{\kappa_f} - 1}{\sqrt{\kappa_f} + 1}, \quad \rho = \sqrt{1 - \frac{1}{\sqrt{\kappa_f}}}. \tag{4.22}$$

In Figure 2, we plot the analytical rate bound ρ given in (4.22) for various values of the condition number. We also plot the numerical rate bounds obtained by solving the SDP in (3.16) with M_0^1, M_0^2, and M_0^3 given in (4.20), and h and β selected according to (4.22). Finally, we plot the theoretical lower bound on the convergence rate for the class $F(m_f, L_f)$ [23]. We observe that the semidefinite program yields slightly better bounds than the analytical rate bound, showing the nonconservatism of the proposed semidefinite program. We remark that in [19] the authors make use of quadratic Lyapunov functions and "off-by-one" IQCs to obtain numerical rate bounds for strongly convex problems. They showed that pointwise IQCs alone exhibit crude bounds, and the use of off-by-one IQCs improves the numerical solutions greatly. In contrast, we have utilized *nonquadratic* Lyapunov functions and *pointwise* IQCs, which yield nonconservative rate bounds. This nonconservatism is due to the inclusion of the term $a_k(F(x_k) - F(x_\star))$ in the Lyapunov function. We conjecture that, by incorporating off-by-one IQCs or other IQCs developed in [19] in our Lyapunov framework, we can further improve the numerical bounds.

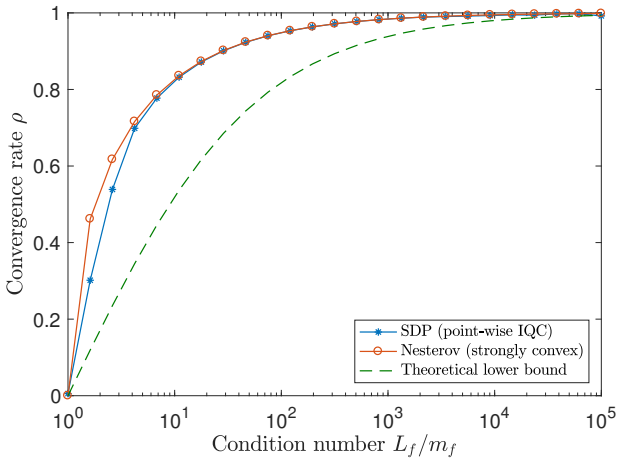

FIG. 2. *Comparison of rate bounds in Nesterov's method for different ratios $\kappa_f = L_f/m_f$ using the parameter selection $h = 1/L_f$ and $\beta = \frac{\sqrt{\kappa_f}-1}{\sqrt{\kappa_f}+1}$. For this parameter selection, the analytical rate bound is $\rho = \sqrt{1-\frac{1}{\sqrt{\kappa_f}}}$ [23]. The theoretical rate bound is $\rho_{lb} = 1-\frac{1}{\sqrt{\kappa_f}}$.*

4.3. Numerical bounds for subexponential rates. For time-varying algorithms and nonstrongly convex functions, the convergence rate is subexponential and the LMI (3.7) becomes dependent on the iteration number. In this case, a numerical approach amounts to solving an infinite sequence of LMIs to find a rate-generating sequence $\{a_k\}$. Nevertheless, we can truncate the sequence of LMIs in order to obtain rate bounds for a *finite* number of iterations. Specifically, for a given $N > 0$, we consider the following semidefinite program:

$$\begin{aligned} &\text{maximize} && a_N \\ &\text{subject to} && \text{for } k = 0, 1, \dots, N-1: \\ &&& M_k^0 + a_k M_k^1 + (a_{k+1} - a_k) M_k^2 + \sigma_k M_k^3 \preceq 0, \\ &&& a_{k+1} \geq a_k, \quad \sigma_k \geq 0, \quad P_k \succeq 0, \end{aligned} \tag{4.23}$$

with decision variables $\{(a_k, P_k, \sigma_k)\}_{k=1}^N$. Denoting the optimal solution of (4.23) by $a_N^\star$, Theorem 3.1 immediately implies

$$f(x_N) - f(x_\star) \leq \frac{V_0(x_0, \xi_0)}{a_N^\star}. \tag{4.24}$$

In other words, (4.23) searches for the smallest upper bound on the Nth (last) iterate suboptimality, subject to the stability constraint imposed by the LMI in (3.7). Notice that (4.23) is homogeneous in the decision variables. To get a sensible problem, we must normalize the variables by, for example, requiring all of them to add up to a positive constant. Furthermore, the kth LMI in (4.23) is a function of a_k, a_{k+1}, P_k, P_{k+1}, and σ_k. This implies that the SDP is banded with a fixed bandwidth independent of N, the number of iterations. We can exploit this sparsity structure in solving the semidefinite program efficiently. For instance, for Nesterov's accelerated method and $N = 10^3$ iterations, solving the corresponding semidefinite program takes less than 10 seconds with an off-the-shelf solver.

In Figure 3, we plot numerical rate bounds obtained by solving (4.23) for Nesterov's accelerated method (4.20) with the parameter selection given in (4.18). We also plot the analytical rate bound given in (4.21). We observe that the numerical rate bound coincides with the analytical rate.

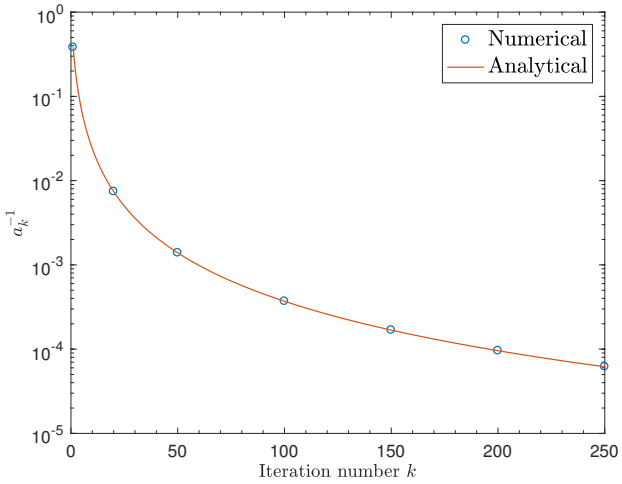

FIG. 3. *Comparison of rate bounds obtained by numerically solving the semidefinite program in* (4.23) *and analytical rate bounds for Nesterov's accelerated method with the parameter selection given in* (4.18).

5. Composite optimization problems. In this section, we consider composite optimization problems of the form

$$\mathcal{X}_\star = \operatorname{argmin}_{x\in\mathbb{R}^d}\{F(x) = f(x) + g(x)\}, \tag{5.1}$$

where $f\colon \mathbb{R}^d \to \mathbb{R}$ is differentiable CCP, while $g\colon \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ is nondifferentiable and CCP. We assume that the optimal solution set $\mathcal{X}_\star$ is nonempty and closed and that the optimal value $F(x_\star)$ is finite. Under these assumptions, the optimality condition for (5.1) is given by

$$\mathcal{X}_\star = \{x_\star \in \operatorname{dom} f \cap \operatorname{dom} g\colon 0 \in \nabla f(x_\star) + \partial g(x_\star)\}. \tag{5.2}$$

Formally, the objective function in (5.1) is nonsmooth, and subgradient methods are very slow. Splitting methods such as proximal algorithms circumvent this issue by exploiting the special structure of the objective function to achieve convergence rates comparable to their counterparts in smooth programming. In this section, we analyze proximal algorithms using Theorem 3.1. To this end, we first show that we can represent these algorithms in the canonical form (2.1), where the feedback nonlinearity ϕ is the generalized gradient mapping of F, which we will define next. By deriving the proximal counterpart of Lemma 4.1, we can then immediately apply Theorem 3.1 to proximal algorithms.

5.1. Generalized gradient mapping. Let $g\colon \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ be a CCP function. The proximal operator $\Pi_{g,h}\colon \mathbb{R}^d \to \mathbb{R}^d$ of g with parameter $h > 0$ is defined

as

$$\Pi_{g,h}(x) = \operatorname{argmin}_{y\in\mathbb{R}^d} \left\{ g(y) + \frac{1}{2h}\|y-x\|_2^2 \right\}. \tag{5.3}$$

For the composite function in (5.1), we define the generalized gradient mapping $\phi_h\colon \mathbb{R}^d \to \mathbb{R}^d$ as

$$\phi_h(x) = \frac{1}{h}(x - \Pi_{g,h}(x - h\nabla f(x))),\ h > 0, \tag{5.4}$$

with $\operatorname{dom}\phi_h = \operatorname{dom} f$. Notice that when $g(x) \equiv 0$ (so that $\Pi_{g,h}(x) = x$), the generalized gradient mapping simplifies to the gradient function ∇f. Furthermore, we have that $\phi_h(x_\star) = 0$ for $x_\star \in \mathcal{X}_\star$, i.e., ϕ_h vanishes at optimality. In the following proposition, we characterize several properties of ϕ_h which will prove useful.

PROPOSITION 5.1. *Consider the composite function $F = f+g$ with $f\in\mathcal{F}(m_f, L_f)$ and $g \in \mathcal{F}(0,\infty)$. Correspondingly, define the generalized gradient mapping ϕ_h of F as in* (5.4).

1. *ϕ_h satisfies the pointwise IQC defined by $(Q_{\phi_h}, x_\star, \phi_h(x_\star))$, where Q_{ϕ_h} is given by*

$$Q_{\phi_h} = \begin{bmatrix} \frac{1}{2h}(\gamma_f^2 - 1) & \frac{1}{2} \\ \frac{1}{2} & -\frac{h}{2} \end{bmatrix} \otimes I_d, \tag{5.5}$$

   with $\gamma_f = \max\{|1-hL_f|, |1-hm_f|\}$.

2. *The inequality*

$$F(y-h\phi_h(y)) - F(x) \le \phi_h(y)^\top(y-x) - \frac{m_f}{2}\|y-x\|_2^2 + \left(\frac{1}{2}L_f h^2 - h\right)\|\phi_h(y)\|_2^2 \tag{5.6}$$

   holds for all $h \ge 0$ and $x, y \in \operatorname{dom} F$.

3. *$\phi_h(x_\star) = 0$ if and only if $x_\star \in \operatorname{argmin} F(x)$.*

Proof. See Appendix B. ∎

5.2. Proximal algorithms. Using the definition of generalized gradient mapping in (5.4), we can represent proximal algorithms with the same state-space structure as in (2.1), where the feedback nonlinearity is ϕ_h. For example, Nesterov's accelerated proximal gradient method is defined by

$$\begin{aligned} x_{k+1} &= \Pi_{g,h}(y_k - h\nabla f(y_k)), \\ y_k &= x_k + \beta_k(x_k - x_{k-1}), \end{aligned} \tag{5.7}$$

which, by using (5.4), can be rewritten as

$$\begin{aligned} x_{k+1} &= x_k + \beta_k(x_k - x_{k-1}) - h\phi_h(y_k), \\ y_k &= x_k + \beta_k(x_k - x_{k-1}). \end{aligned} \tag{5.8}$$

By defining the state vector $\xi_k = [x_{k-1}^\top \; x_k^\top]^\top \in \mathbb{R}^{2d}$, the corresponding state-space matrices (A_k, B_k, C_k) are given by

$$\left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & 0 \end{array}\right] = \left[\begin{array}{cc|c} 0 & I_d & 0 \\ -\beta_k I_d & (\beta_k+1) I_d & -h I_d \\ \hline -\beta_k I_d & (\beta_k+1) I_d & 0 \end{array}\right]. \tag{5.9}$$

We observe that (5.9) has the same structure as Nesterov's accelerated method without proximal operation, with the difference that ∇f is replaced by ϕ_h in the nonlinear block. Recall the assumption that the sequences $\{x_k\}$ and $\{y_k\}$ converge to the same fixed point in the optimal set. Since ϕ_h is zero at optimality, we must therefore have that $\phi_h(y_\star) = \phi_h(x_\star) = 0$. In other words, the fixed points satisfy

$$\xi_\star = A_k \xi_\star, \quad y_\star = C_k \xi_\star, \quad u_\star = \phi_h(y_\star) = 0, \quad x_\star = E_k \xi_\star = y_\star \quad \text{for all } k. \tag{5.10}$$

Having characterized the generalized gradient mapping with quadratic constraints (Proposition 5.1), we are now ready to develop the proximal counterpart of Lemma 4.1.

LEMMA 5.2. *Let $x_\star \in \operatorname{argmin} F(x)$ be a minimizer of $F = f + g$ with a finite optimal value $F(x_\star)$, where $f \in \mathcal{F}(m_f, L_f)$ and $g \in \mathcal{F}(0, \infty)$. Consider a proximal first-order algorithm in the state-space form* (2.1) *with $\phi = \phi_h$ defined as in* (5.4). *Suppose the fixed points $(\xi_\star, u_\star, y_\star, x_\star)$ satisfy*

$$\xi_\star = A_k \xi_\star, \quad y_\star = C_k \xi_\star, \quad u_\star = \phi_h(y_\star) = 0, \quad x_\star = E_k \xi_\star = y_\star \quad \textit{for all } k. \tag{5.11}$$

Then the following inequalities hold for all k:

$$F(x_{k+1}) - F(x_k) \le e_k^\top M_k^1 e_k, \tag{5.12a}$$

$$F(x_{k+1}) - F(x_\star) \le e_k^\top M_k^2 e_k, \tag{5.12b}$$

$$0 \le e_k^\top M_k^3 e_k, \tag{5.12c}$$

where $e_k = [(\xi_k - \xi_\star)^\top \; (u_k - u_\star)^\top]^\top$ and M_k^1, M_k^2, M_k^3 are given by

$$\begin{aligned} M_k^1 &= \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & (\frac{1}{2} L_f h^2 - h) \end{bmatrix} \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix}, \\ M_k^2 &= \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & (\frac{1}{2} L_f h^2 - h) \end{bmatrix} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}, \\ M_k^3 &= \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top Q_{\phi_h} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}. \end{aligned} \tag{5.13}$$

Proof. See Appendix C. ∎

Remark 3. In [19], the authors use a different block diagonal representation of proximal algorithms, in which the linear component is in parallel feedback connection with the gradient function ∇f, and with the subdifferential operator ∂g. Then, each nonlinear block is described by its corresponding IQC, i.e., the IQC of gradient mappings and subdifferential operators. In contrast, we collectively represent all the nonlinearities in a single feedback component (the generalized gradient mapping), whose IQC is given in Lemma 5.1.

In the following, we use Lemma 5.2 in conjunction with Theorem 3.1 to analyze the proximal gradient method and the proximal variant of Nesterov's accelerated method.

5.2.1. Proximal gradient method. The classical proximal gradient method is defined by the recursion

$$x_{k+1} = \Pi_{hg}(x_k - h\nabla f(x_k)), \tag{5.14}$$

which, by using the definition of the generalized gradient mapping in (5.4), can be written as

$$x_{k+1} = x_k - h\phi_h(x_k). \tag{5.15}$$

The state-space matrices are therefore given by $(A_k, B_k, C_k, E_k) = (I_d, -hI_d, I_d, I_d)$. By selecting $P_k = p_k I_d,\ p_k \geq 0$, the matrices $M_k^i, i = 0, 1, 2, 3$, are given by

$$M_k^0 = \begin{bmatrix} p_{k+1}-p_k & -hp_{k+1} \\ -hp_{k+1} & h^2 p_{k+1} \end{bmatrix} \otimes I_d, \tag{5.16a}$$

$$M_k^1 = \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{2}(L_f h^2 - 2h) \end{bmatrix} \otimes I_d, \tag{5.16b}$$

$$M_k^2 = \begin{bmatrix} -\frac{1}{2}m_f & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2}(L_f h^2 - 2h) \end{bmatrix} \otimes I_d, \tag{5.16c}$$

$$M_k^3 = \begin{bmatrix} \frac{1}{2h}(\gamma_f^2 - 1) & \frac{1}{2} \\ \frac{1}{2} & -\frac{h}{2} \end{bmatrix} \otimes I_d, \tag{5.16d}$$

where $\gamma_f = \max\{|1 - hL_f|, |1 - hm_f|$.

Strongly convex case. We first consider the selection $a_k \equiv 0$ for strongly convex settings. Then the LMI (5.16) simplifies to

$$\begin{bmatrix} p_{k+1}-p_k & -hp_{k+1} \\ -hp_{k+1} & h^2 p_{k+1} \end{bmatrix} + \sigma_k \begin{bmatrix} \frac{\gamma_f^2 - 1}{2h} & \frac{1}{2} \\ \frac{1}{2} & -\frac{h}{2} \end{bmatrix} \leq 0.$$

It can be verified that the above LMI is equivalent to the conditions

$$\sigma_k/(2h) \leq p_k/\gamma_f^2, \quad p_{k+1} - p_k \leq \sigma_k(1 - \gamma_f^2)/(2h).$$

These two conditions together imply $p_{k+1} \leq p_k/\gamma_f^2$. Therefore, we can write $p_k = \gamma_f^{-2k} p_0,\ p_0 > 0$. Using the bound (3.20), we can establish the bound

$$\|x_k - x_\star\|_2^2 \leq \left(\max\{|1-hL_f|, |1-hm_f|\}\right)^{2k} \|x_0 - x_\star\|_2^2.$$

On the other hand, setting $p_k \equiv 0$ in (5.16) yields the LMI

$$\begin{bmatrix} -\dfrac{m_f}{2}(a_{k+1} - a_k) & \dfrac{a_{k+1} - a_k}{2} \\ \dfrac{a_{k+1} - a_k}{2} & \left(\dfrac{L_f h^2}{2} - h\right) a_{k+1} \end{bmatrix} \preceq 0.$$

Omitting the details, we obtain from the above LMI that $a_{k+1} \leq \rho^{-2} a_k$ and $0 \leq h \leq 2/L_f$, where $\rho^2 = 1 + m_f(L_f h^2 - 2h)$. Substituting a_k in (3.17) yields the bound

$$F(x_k) - F(x_\star) \leq (1 + m_f(L_f h^2 - 2h))^k (F(x_0) - F(x_\star)).$$

In particular, the optimal decay rate is attained at $h = 1/L_f$ and is equal to $\rho = 1 - m_f/L_f$.

Convex case. When the differentiable component of the objective is convex ($m_f = 0$), we select $p_k = p > 0, \sigma_k = \sigma$ in (5.16) to arrive at the LMI

$$\begin{bmatrix} \frac{\sigma}{2h}(\gamma_f^2 - 1) & \frac{1}{2}(a_{k+1} - a_k - 2ph + \sigma) \\ \frac{1}{2}(a_{k+1} - a_k - 2ph + \sigma) & \left(\frac{L_f h^2}{2} - h\right) a_{k+1} + ph^2 - \frac{\sigma h}{2} \end{bmatrix} \preceq 0.$$

To further simplify the LMI, we take $\sigma = 0$. Then the LMI enforces that

$$a_{k+1} = a_k + 2ph, \quad a_0 \geq 0, \quad (L_f h^2/2 - h)(a_{k+1}) + ph^2 \leq 0.$$

Solving for a_k leads to

$$F(x_k) - F(x_\star) \leq \frac{a_0(F(x_0) - F(x_\star)) + p\|x_0 - x_\star\|_2^2}{a_0 + 2phk}.$$

In particular, if $a_0 = 0$, then it must hold that $0 \leq h \leq 1/L_f$, and we recover the convergence result in [3, Theorem 3.1].

5.2.2. Accelerated proximal gradient method. Consider the proximal variant of Nesterov's accelerated method outlined in (5.7), for which the state-space matrices are given in (5.9). Making use of Lemma 5.2, the matrices $M_k^i, i \in \{0, 1, 2, 3\}$, read

$$M_k^0 = \begin{bmatrix} A_k^\top P_{k+1} A_k - P_k & A_k^\top P_{k+1} B_k \\ B_k^\top P_{k+1} A_k & B_k^\top P_{k+1} B_k \end{bmatrix},$$

$$M_k^1 = \begin{bmatrix} -\frac{1}{2} m_f \beta_k^2 & \frac{1}{2} m_f \beta_k^2 & -\frac{1}{2}\beta_k \\ \frac{1}{2} m_f \beta_k^2 & -\frac{1}{2} m_f \beta_k^2 & \frac{1}{2}\beta_k \\ -\frac{1}{2}\beta_k & \frac{1}{2}\beta_k & \frac{1}{2} L_f h^2 - h \end{bmatrix},$$

$$M_k^2 = \begin{bmatrix} -\frac{1}{2} m_f \beta_k^2 & \frac{1}{2} m_f \beta_k(\beta_k + 1) & -\frac{1}{2}\beta_k \\ \frac{1}{2} m_f \beta_k(\beta_k + 1) & -\frac{1}{2} m_f (\beta_k + 1)^2 & \frac{1}{2}(\beta_k + 1) \\ -\frac{1}{2}\beta_k & \frac{1}{2}(\beta_k + 1) & \frac{1}{2} L_f h^2 - h \end{bmatrix},$$

$$M_k^3 = \begin{bmatrix} -\beta_k I_d & 0 \\ (1 + \beta_k) I_d & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \frac{1}{2h}(\gamma_f^2 - 1) I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & -\frac{h}{2} I_d \end{bmatrix} \begin{bmatrix} -\beta_k I_d & (1 + \beta_k) I_d & 0 \\ 0 & 0 & I_d \end{bmatrix}. \tag{5.17}$$

Observe that the matrices M_k^0, M_k^1, and M_k^2 are precisely the same as those of Nesterov's method without proximal operation. The only difference is in M_k^3. As a result, by setting $\sigma_k = 0$ (the coefficient of M_k^3) in the LMI (3.7), the analysis of Nesterov's accelerated method in section 4.1.2 immediately applies to the proximal variant [11].

Remark 4 (gradient methods with projection). For the case when $g(x) = \mathbb{I}_{\mathcal{X}}(x)$ is the indicator function of a nonempty, closed convex set $\mathcal{X} \subset \mathbb{R}^d$, the proximal operator $\Pi_{g,h}$ reduces to projection onto $\mathcal{X}$. Due to projection, we must have $x_k \in \mathcal{X}$ for all k, implying $g(x_k) = 0$. Therefore, the convergence result of Theorem 3.1 holds for the suboptimality $f(x_k) - f(x_\star)$.

6. Further topics. In this section, we consider further applications of the developed framework, namely, calculus of IQCs for various operators in optimization, continuous-time models, and more importantly, algorithm design.

6.1. Calculus of IQCs. We now describe some operations on mappings from an IQC perspective, namely, inversion, affine operations, and function composition. These operations form a calculus that is useful for determining IQCs for commonly used nonlinear operators in optimization algorithms, such as proximal operators, projection operators, reflection operators, etc., and their compositions.

It directly follows from the definition of pointwise IQCs in (3.21) that if ϕ satisfies multiple pointwise IQCs defined by $(Q_{\phi,i},\, x_\star,\, \phi(x_\star))$, $i = 1, 2, \ldots, \ell$, it also satisfies the pointwise IQC defined by $(\sum_{i=1}^{\ell} \sigma_i Q_{\phi,i}\ , x_\star,\, \phi(x_\star))$, where $\sigma_i \geq 0,\ i = 1, 2, \ldots, \ell$. Further, ϕ also satisfies the IQC defined by $(Q, x_\star, \phi(x_\star))$ for any $Q \succeq Q_\phi$. In the next two lemmas, we study the effect of inversion and affine transformation on IQCs.

LEMMA 6.1 (IQC for inversion). *Consider an invertible map $\phi\colon \mathbb{R}^d \to \mathbb{R}^d$ with $\phi^{-1}(\operatorname{dom}\phi) \subseteq \operatorname{dom}\phi$ satisfying the pointwise IQC defined by $(Q_\phi, x_\star, \phi(x_\star))$. Then, the inverse map $\phi^{-1}\colon \mathbb{R}^d \to \mathbb{R}^d$ satisfies the pointwise IQC defined by $(Q_{\phi^{-1}}, \phi(x_\star), x_\star)$, where*

$$Q_{\phi^{-1}} = \begin{bmatrix} 0 & I_d \\ I_d & 0 \end{bmatrix} Q_\phi \begin{bmatrix} 0 & I_d \\ I_d & 0 \end{bmatrix}. \tag{6.1}$$

Proof. By the substitution $x \leftarrow \phi^{-1}(x)$ in (3.21), we obtain

$$\begin{bmatrix} \phi^{-1}(x) - \phi^{-1}(x_\star) \\ x - x_\star \end{bmatrix}^\top Q_\phi \begin{bmatrix} \phi^{-1}(x) - \phi^{-1}(x_\star) \\ x - x_\star \end{bmatrix} \geq 0. \tag{6.2}$$

Further, we have

$$\begin{bmatrix} \phi^{-1}(x) - \phi^{-1}(x_\star) \\ x - x_\star \end{bmatrix} = \begin{bmatrix} 0 & I_d \\ I_d & 0 \end{bmatrix} \begin{bmatrix} x - x_\star \\ \phi^{-1}(x) - \phi^{-1}(x_\star) \end{bmatrix}. \tag{6.3}$$

Substituting (6.3) in (6.2) yields (6.1). ∎

LEMMA 6.2 (IQC for affine operations). *Consider a map $\phi\colon \mathbb{R}^d \to \mathbb{R}^d$ satisfying the pointwise IQC defined by $(Q_\phi, x_\star, \phi(x_\star))$. Correspondingly, define the map $\psi(x) = S_2 x + S_1 \phi(S_0 x)$ with $S_0(\operatorname{dom}\phi) \subseteq \operatorname{dom}\phi$, where $S_0, S_1, S_2 \in \mathbb{R}^{d\times d}$, and S_1 is invertible. Then, ψ satisfies the pointwise IQC defined by $(Q_\psi, x_\star, \psi(x_\star))$, where*

$$Q_\psi = \begin{bmatrix} S_0^\top & -(S_1^{-1}S_2)^\top \\ 0 & S_1^{-1} \end{bmatrix} Q_\phi \begin{bmatrix} S_0 & 0 \\ -S_1^{-1}S_2 & (S_1^{-1})^\top \end{bmatrix}. \tag{6.4}$$

Proof. By the substitution $x \leftarrow S_0 x$ in (3.21), we obtain

$$\begin{bmatrix} S_0x - S_0x_\star \\ \phi(S_0x) - \phi(S_0x_\star) \end{bmatrix}^\top Q_\phi \begin{bmatrix} S_0x - S_0x_\star \\ \phi(S_0x) - \phi(S_0x_\star) \end{bmatrix} \geq 0. \tag{6.5}$$

Further, since $\psi(x) = S_2x + S_1\phi(S_0x)$, we have

$$\begin{bmatrix} S_0x - S_0x_\star \\ \phi(S_0x) - \phi(S_0x_\star) \end{bmatrix} = \begin{bmatrix} S_0 & 0 \\ -S_1^{-1}S_2 & S_1^{-1} \end{bmatrix} \begin{bmatrix} x - x_\star \\ \psi(x) - \psi(x_\star) \end{bmatrix}. \tag{6.6}$$

Substituting (6.6) in (6.5) yields (6.4). ∎

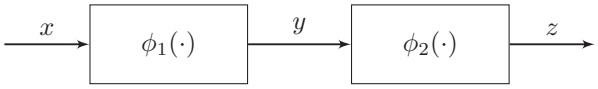

FIG. 4. *Cascade connection of two nonlinear mappings.*

Finally, we study the composition of mappings. Specifically, consider the cascade connection of two mappings $\phi_1, \phi_2 \colon \mathbb{R}^d \to \mathbb{R}^d$, $i = 1, 2$, as in Figure 4, where $y = \phi_1(x)$ and $z = \phi_2(y)$. Further assume ϕ_1 and ϕ_2 satisfy pointwise IQCs defined by $(Q_{\phi_1}, x_\star, y_\star)$ and $(Q_{\phi_2}, y_\star, z_\star)$, respectively. By definition, these mappings impose the following quadratic constraints on the pairs (x, y) and (y, z):

$$\begin{bmatrix} x - x_\star \\ y - y_\star \end{bmatrix}^\top Q_{\phi_1} \begin{bmatrix} x - x_\star \\ y - y_\star \end{bmatrix} \geq 0, \quad \begin{bmatrix} y - y_\star \\ z - z_\star \end{bmatrix}^\top Q_{\phi_2} \begin{bmatrix} y - y_\star \\ z - z_\star \end{bmatrix} \geq 0.$$

These two constraints separately define a quadratic constraint on the triple (x, y, z), which can be encapsulated in a single constraint, as follows:

$$\begin{bmatrix} x - x_\star \\ y - y_\star \\ z - z_\star \end{bmatrix}^\top Q_\psi \begin{bmatrix} x - x_\star \\ y - y_\star \\ z - z_\star \end{bmatrix} \geq 0, \tag{6.7a}$$

where $Q_\psi \in \mathbb{S}^{3d}$ is given by

$$Q_\psi = \begin{bmatrix} I_d & 0 \\ 0 & I_d \\ 0 & 0 \end{bmatrix} \sigma_1 Q_{\phi_1} \begin{bmatrix} I_d & 0 & 0 \\ 0 & I_d & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ I_d & 0 \\ 0 & I_d \end{bmatrix} \sigma_2 Q_{\phi_2} \begin{bmatrix} 0 & I_d & 0 \\ 0 & 0 & I_d \end{bmatrix}, \tag{6.7b}$$

with $\sigma_1, \sigma_2 \geq 0$. The quadratic constraint in (6.7a) follows by substituting (6.7b) into (6.7a). In the language of IQCs, we say that the map $\psi = [\phi_1^\top \ (\phi_2 \circ \phi_1)^\top]^\top \colon \mathbb{R}^d \to \mathbb{R}^{2d}$ satisfies the pointwise IQC defined by $(Q_\psi, x_\star, \psi(x_\star))$, where Q_ψ is given by (6.7b).

We remark that the above treatment can be extended to multiple compositions. Specifically, for ℓ mappings in a cascade connection, the corresponding ℓ individual IQCs can be grouped into a single quadratic constraint on the concatenated vector of the input-output signals.

6.1.1. Proximal operators. Recall the definition of proximal operator for $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$:

$$\Pi_{f,h}(x) = \operatorname{argmin}_{y \in \mathbb{R}^d} \left\{ f(y) + \frac{1}{2h} \|y - x\|_2^2 \right\}. \tag{6.8}$$

To characterize $\Pi_{f,h}$ from an IQC perspective, we note that for any given $x \in \operatorname{dom} f$, a necessary condition for optimality in (6.8) is that

$$0 \in \partial f(\Pi_{g,h}(x)) + \frac{1}{h}(\Pi_{f,h}(x) - x) \quad \text{for all } x \in \operatorname{dom} f, \tag{6.9}$$

which is an implicit equation on $\Pi_{f,h}(x)$. In the next proposition, we show how to obtain a quadratic constraint for the proximal operator $\Pi_{f,h}$ from that of the subgradient T_f by using the necessary optimality condition (6.9) that couples these two operators.

PROPOSITION 6.3 (IQCs for proximal operators). *Let $f\colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a closed proper function whose subgradient T_f satisfies the pointwise IQC defined by $(Q_f, x_\star, T_f(x_\star))$, where $T_f(x_\star) \in \partial f(x_\star)$. Then, the proximal operator Π_{hf} satisfies the pointwise IQC defined by $(Q_{\Pi_{hf}}, x_\star, \Pi_{hf}(x_\star))$, where*

$$Q_{\Pi_{hf}} = \begin{bmatrix} 0 & h^{-1}I_d \\ I_d & -h^{-1}I_d \end{bmatrix} Q_f \begin{bmatrix} 0 & I_d \\ h^{-1}I_d & -h^{-1}I_d \end{bmatrix}. \tag{6.10}$$

Proof. Suppose $T_f(x) \in \partial f(x)$ ($T_f(x) = \nabla f(x)$ when f is differentiable) satisfies the pointwise IQC defined by $(Q_f, x_\star, T_f(x_\star))$. By the substitutions $x \leftarrow \Pi_{hf}(x)$ and $x_\star \leftarrow \Pi_{hf}(x_\star)$ in (3.21), we obtain

$$\begin{bmatrix} \Pi_{hf}(x) - \Pi_{hf}(x_\star) \\ T_f(\Pi_{hf}(x)) - T_f(\Pi_{hf}(x_\star)) \end{bmatrix}^\top Q_f \begin{bmatrix} \Pi_{hf}(x) - \Pi_{hf}(x_\star) \\ T_f(\Pi_{hf}(x)) - T_f(\Pi_{hf}(x_\star)) \end{bmatrix} \geq 0. \tag{6.11}$$

On the other hand, by the optimality condition (6.9), we have $T_f(\Pi_{hf}(x)) = \frac{1}{h}(x - \Pi_{hf}(x))$. Substituting this into (6.11), we obtain

$$\begin{bmatrix} \Pi_{hf}(x) - \Pi_{hf}(x_\star) \\ \frac{1}{h}(x - x_\star) - \frac{1}{h}(\Pi_{hf}(x) - \Pi_{hf}(x_\star)) \end{bmatrix}^\top Q_f \begin{bmatrix} \Pi_{hf}(x) - \Pi_{hf}(x_\star) \\ \frac{1}{h}(x - x_\star) - \frac{1}{h}(\Pi_{hf}(x) - \Pi_{hf}(x_\star)) \end{bmatrix} \geq 0. \tag{6.12}$$

Further, we can write

$$\begin{bmatrix} \Pi_{hf}(x) - \Pi_{hf}(x_\star) \\ \frac{1}{h}(x - x_\star) - \frac{1}{h}(\Pi_{hf}(x) - \Pi_{hf}(x_\star)) \end{bmatrix} = \begin{bmatrix} 0 & I_d \\ \frac{1}{h}I_d & -\frac{1}{h}I_d \end{bmatrix} \begin{bmatrix} x - x_\star \\ \Pi_{hf}(x) - \Pi_{hf}(x_\star) \end{bmatrix}. \tag{6.13}$$

By substituting (6.13) in (6.12), we arrive at the desired inequality in (6.10). ∎

Notice that by (6.9), we have that $\Pi_{hf} = (I + h\partial f)^{-1}$. In other words, the proximal operator is obtained by the operations $\partial f \to I + h\partial f \to (I + h\partial f)^{-1}$, i.e., an affine operation on ∂f followed by an inversion. Therefore, to obtain the IQC of Π_{hf} from that of ∂f, we can directly use Lemmas 6.1 and 6.2 to arrive at an alternative derivation of (6.10).

6.1.2. IQCs for projection operators. The projection operator is the proximal operator Π_{hf} for the particular selection $f(x) = \mathbb{I}_{\mathcal{X}}(x)$, where $\mathbb{I}_{\mathcal{X}}$ is the extended-value indicator function of the nonempty closed convex set $\mathcal{X} \subset \mathbb{R}^d$ onto which we project. Since f is nondifferentiable and convex in this case, its subgradient operator T_f satisfies the pointwise IQC defined by $(Q_f, x_\star, T_f(x_\star))$, where Q_f is given by (3.27) with $L_f = \infty$. It then follows from Proposition 6.3 that the projection operator $\Pi_{\mathcal{X}}$ satisfies the IQC defined by $(Q_{\Pi_{\mathcal{X}}}, x_\star, \Pi_{\mathcal{X}}(x_\star))$, where

$$Q_{\Pi_{\mathcal{X}}} = \begin{bmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & -1 \end{bmatrix} \otimes I_d. \tag{6.14}$$

This IQC corresponds to the firm nonexpansiveness property of the projection operator [7], which implies the Lipschitz continuity of $\Pi_{\mathcal{X}}$ with Lipschitz parameter equal to one.

6.2. Beyond convexity. The convergence analysis of several algorithms does not make full use of convexity. In other words, convexity is sufficient for convergence of these algorithms. This has motivated the introduction of function classes that are relaxations of convexity. In this subsection, we briefly discuss some of these classes and how they can be related to the framework developed in this paper. Formally, consider a continuously differentiable function $f\colon \mathbb{R}^d \to \mathbb{R}$ that satisfies the bounds

$$\begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix}^\top R'_f \begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix} \leq f(x)-f(x_\star) \leq \begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix}^\top R_f \begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix}, \tag{6.15}$$

where $R_f, R'_f \in \mathbb{S}^{2d}$ are symmetric matrices and $x_\star$ is such that $\nabla f(x_\star)=0$. It follows from (6.15) that

$$\begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix}^\top (R_f - R'_f) \begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix} \geq 0. \tag{6.16}$$

Note that since $\nabla f(x_\star)=0$, the above inequality implies that ∇f satisfies the pointwise IQC defined by $(R_f - R'_f, x_\star, \nabla f(x_\star))$. Several function classes can be written in the form (6.15), where R_f and R'_f differ for each class. We give three examples below.

6.2.1. (Strongly) convex functions. In section 3.2.2, we considered IQCs for convex functions. Specifically, the quadratic inequality (3.26) is necessary and sufficient for the inclusion $f \in \mathcal{F}(m_f, L_f)$. An equivalent inequality involving function values is [29][4]

$$\begin{aligned} f(y)-f(x)-\nabla f(x)^\top(y-x) \geq{}& \frac{1}{2(L_f-m_f)}\|\nabla f(y)-\nabla f(x)\|_2^2 \\ &+ \frac{m_f L_f}{2(L_f-m_f)}\|y-x\|_2^2 - \frac{m_f}{L_f-m_f}(\nabla f(y)-\nabla f(x))^\top(y-x). \end{aligned} \tag{6.17}$$

If we restrict (6.17) to hold only for the particular selections $(x,y)=(x_\star,x)$ and $(x,y)=(x,x_\star)$, we obtain a new class of functions that can be put in the form (6.15) with R'_f, R_f given by

$$R'_f = \begin{bmatrix} \frac{m_f L_f}{2(L_f-m_f)} & \frac{-m_f}{2(L_f-m_f)} \\ \frac{-m_f}{2(L_f-m_f)} & \frac{1}{2(L_f-m_f)} \end{bmatrix} \otimes I_d, \quad R_f = \begin{bmatrix} \frac{-m_f L_f}{2(L_f-m_f)} & \frac{L_f}{2(L_f-m_f)} \\ \frac{L_f}{2(L_f-m_f)} & \frac{-1}{2(L_f-m_f)} \end{bmatrix} \otimes I_d. \tag{6.18}$$

Using (6.16), we can conclude that

$$\begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f L_f}{m_f+L_f} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & -\frac{1}{m_f+L_f} I_d \end{bmatrix} \begin{bmatrix} x-x_\star \\ \nabla f(x) \end{bmatrix} \geq 0. \tag{6.19}$$

Note that this quadratic inequality is the same as that of convex functions but only holds when reference point $x_\star$ in the definition of pointwise IQC satisfies $\nabla f(x_\star)=0$.

[4]Note that by adding both sides of (6.17) to the inequality obtained by interchanging x and y in (6.17), we obtain (3.26).

6.2.2. Weakly smooth weakly quasi-convex functions. Suppose f is continuously differentiable and satisfies [13]

$$\frac{1}{\Gamma_f}\|\nabla f(x)\|_2^2 \le f(x) - f(x_\star) \le \frac{1}{\tau_f}\nabla f(x)^\top (x - x_\star) \quad \text{for all } x \in \mathcal{S}, \tag{6.20}$$

where $x_\star$ is a global minimum of f, and $0 < \tau_f, \Gamma_f < \infty$. These inequalities ensure that any point with vanishing gradient is optimal [13], i.e., $\nabla f(x_\star) = 0$. The inequality (6.20) can be put in the form (6.15), where R'_f, R_f, and Q_f are given by

$$R'_f = \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{\Gamma_f} \end{bmatrix} \otimes I_d, \quad R_f = \begin{bmatrix} 0 & \frac{1}{2\tau_f} \\ \frac{1}{2\tau_f} & 0 \end{bmatrix} \otimes I_d, \quad Q_f = \begin{bmatrix} 0 & \frac{1}{2\tau_f} \\ \frac{1}{2\tau_f} & -\frac{1}{\Gamma_f} \end{bmatrix} \otimes I_d. \tag{6.21}$$

6.2.3. Polyak–Łojasiewicz (PL) condition. Suppose f is continuously differentiable and satisfies

$$0 \le f(x) - f(x_\star) \le \frac{1}{2m_f}\|\nabla f(x)\|_2^2 \quad \text{for all } x \in \mathcal{S} \tag{6.22}$$

for some $m_f > 0$. Again, this class can be put in the form (6.15).

6.3. Continuous-time models. There is a close connection between iterative algorithms and discretization of ordinary differential equations (ODEs). In fact, many iterative first-order optimization algorithms reduce to their "generative" ODEs by time-scaling and infinitesimal stepsizes. In this subsection, we consider convergence analysis of continuous-time models for solving the unconstrained problem in (4.1). Specifically, consider the following continuous-time dynamical system in state-space form:

$$\dot{\xi}(t) = A(t)\xi(t) + B(t)u(t), \quad y(t) = C(t)\xi(t), \quad u(t) = \nabla f(y(t)) \quad \text{for all } t \ge t_0, \tag{6.23}$$

where at each continuous time $t \ge t_0$, $\xi(t) \in \mathbb{R}^n$ is the state, $y(t) \in \mathbb{R}^d$ is the output ($d \le n$), and $u(t) = \nabla f(y(t))$ is the feedback input. We assume (6.23) solves (4.1) asymptotically from all admissible initial conditions; i.e., $y(t)$ satisfies $\lim_{t\to\infty} f(y(t)) = f(y_\star)$, where the optimal point $y_\star$ obeys $\nabla f(y_\star) = 0$. Therefore, any fixed point of (6.23) satisfies

$$0 = A(t)\xi_\star, \quad y_\star = C(t)\xi_\star, \quad u_\star = \nabla f(y_\star) = 0 \quad \text{for all } t \ge t_0. \tag{6.24}$$

We replicate the convergence analysis of discrete-time models using the Lyapunov function

$$V(\xi(t), t) = a(t)(f(y(t)) - f(y_\star)) + (\xi(t) - \xi_\star)^\top P(t)(\xi(t) - \xi_\star), \tag{6.25}$$

where $(\xi(t), y(t))$ satisfies (6.23) and $(\xi_\star, y_\star)$ satisfies (6.24). The Lyapunov function is parameterized by $P(t) \in \mathbb{S}_+^n$, and $a(t) \ge 0$. If $a(t)$ and $P(t)$ are such that $\dot{V}(\xi(t), t) \le 0$, then we could guarantee that $V(\xi(t), t) \le V(\xi(t_0), t_0)$, which in turn implies

$$0 \le f(y(t)) - f(y_\star) \le V(\xi(t_0), t_0)/a(t) = \mathcal{O}(1/a(t)) \quad \text{for all } t \ge t_0. \tag{6.26}$$

In other words, $a(t)$ provides a lower bound on the convergence rate. Ideally, we are interested in finding the best bound, which translates into the fastest growing $a(t)$. In the following theorem, we develop an LMI to find such an $a(t)$.

THEOREM 6.4. *Let $f \in \mathcal{F}(m_f, L_f)$, and consider the continuous-time dynamics in (6.23), whose fixed points satisfy (6.24). Suppose there exist a differentiable nondecreasing $a(t)\colon [t_0,\infty) \to \mathbb{R}_+$, a differentiable $P(t)\colon [t_0,\infty) \to \mathbb{S}^n_+$, and a continuous $\sigma(t)\colon [t_0,\infty) \to \mathbb{R}_+$ that satisfy*

$$M_0(t) + a(t)M_1(t) + \dot{a}(t)M_2(t) + \sigma(t)M_3(t) \preceq 0 \quad \text{for all } t \geq t_0, \tag{6.27}$$

where

$$M_0(t) = \begin{bmatrix} P(t)A(t)+A(t)^\top P(t)+\dot{P}(t) & P(t)B(t) \\ B(t)^\top P(t) & 0 \end{bmatrix},$$

$$M_1(t) = \frac{1}{2}\begin{bmatrix} 0 & (C(t)A(t)+\dot{C}(t))^\top \\ C(t)A(t)+\dot{C}(t) & C(t)B(t)+B(t)^\top C(t)^\top \end{bmatrix},$$

$$M_2(t) = \begin{bmatrix} C(t)^\top & 0 \\ 0 & I_d \end{bmatrix}\begin{bmatrix} -\frac{m_f}{2}I_d & \frac{1}{2}I_d \\ \frac{1}{2}I_d & 0 \end{bmatrix}\begin{bmatrix} C(t) & 0 \\ 0 & I_d \end{bmatrix},$$

$$M_3(t) = \begin{bmatrix} C(t)^\top & 0 \\ 0 & I_d \end{bmatrix}\begin{bmatrix} -\frac{m_f L_f}{m_f+L_f}I_d & \frac{1}{2}I_d \\ \frac{1}{2}I_d & -\frac{1}{m_f+L_f}I_d \end{bmatrix}\begin{bmatrix} C(t) & 0 \\ 0 & I_d \end{bmatrix}.$$

Then, for any $y(t_0) \in \operatorname{dom} f$, the following inequality holds for all $t \geq t_0$:

$$f(y(t)) - f(y_\star) \leq \frac{a(t_0)(f(y(t_0))-f(y_\star)) + (\xi(t_0)-\xi_\star)^\top P(t_0)(\xi(t_0)-\xi_\star)}{a(t)}. \tag{6.28}$$

Proof. It suffices to show that the LMI condition in (6.27) implies $\dot{V}(\xi(t),t) \leq 0$. The time derivative of the Lyapunov function (6.25) is

$$\dot{V} = \dot{a}(f(y) - f(y_\star)) + a\nabla f(y)^\top \dot{y} + 2(\xi-\xi_\star)^\top P\dot{\xi} + (\xi-\xi_\star)^\top \dot{P}(\xi-\xi_\star). \tag{6.29}$$

We have dropped the arguments for notational simplicity. We proceed to bound all the terms in the right-hand side of (6.29), using the assumption $f \in \mathcal{F}(m_f, L_f)$. By invoking (strong) convexity, we can write

$$\begin{aligned} f(y)-f(y_\star) &\leq \begin{bmatrix} y-y_\star \\ \nabla f(y) - \nabla f(y_\star) \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2}I_d & \frac{1}{2}I_d \\ \frac{1}{2}I_d & 0 \end{bmatrix} \begin{bmatrix} y-y_\star \\ \nabla f(y) - \nabla f(y_\star) \end{bmatrix} \\ &= \begin{bmatrix} \xi-\xi_\star \\ u-u_\star \end{bmatrix}^\top \begin{bmatrix} C & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2}I_d & \frac{1}{2}I_d \\ \frac{1}{2}I_d & 0 \end{bmatrix} \begin{bmatrix} C & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \xi-\xi_\star \\ u-u_\star \end{bmatrix} \\ &= e^\top M_2 e, \end{aligned} \tag{6.30}$$

where we have defined $e = \begin{bmatrix}(\xi-\xi_\star)^\top & (u-u_\star)^\top\end{bmatrix}$. Further, we can write

$$\begin{aligned} \nabla f(y)^\top \dot{y} &= (u-u_\star)^\top (CA(\xi-\xi_\star) + CB(u-u_\star) + \dot{C}(\xi-\xi_\star)) \\ &= \begin{bmatrix} \xi-\xi_\star \\ u-u_\star \end{bmatrix}^\top \begin{bmatrix} 0 & \frac{1}{2}(CA+\dot{C})^\top \\ \frac{1}{2}(CA+\dot{C}) & \frac{1}{2}(CB+B^\top C^\top) \end{bmatrix} \begin{bmatrix} \xi-\xi_\star \\ u-u_\star \end{bmatrix} \\ &= e^\top M_1 e, \end{aligned} \tag{6.31}$$

where we have used (6.23) and (6.24). Similarly, we can write

$$2(\xi-\xi_\star)^\top P\dot{\xi} = \begin{bmatrix} \xi-\xi_\star \\ u-u_\star \end{bmatrix}^\top \begin{bmatrix} PA+A^\top P & PB \\ B^\top P^\top & 0 \end{bmatrix} \begin{bmatrix} \xi-\xi_\star \\ u-u_\star \end{bmatrix} = e^\top M_0 e. \tag{6.32}$$

Finally, since $f \in \mathcal{F}(m_f, L_f)$, ∇f satisfies the quadratic constraint in (3.27). Therefore, we can write

$$e^\top M_3 e = \begin{bmatrix} \xi - \xi_\star \\ u - u_\star \end{bmatrix}^\top \begin{bmatrix} C & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f L_f}{m_f + L_f} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & -\frac{1}{m_f + L_f} I_d \end{bmatrix} \begin{bmatrix} C & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \xi - \xi_\star \\ u - u_\star \end{bmatrix} \tag{6.33}$$
$$= \begin{bmatrix} y - y_\star \\ u - u_\star \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f L_f}{m_f + L_f} I_d & \frac{1}{2} I_d \\ \frac{1}{2} I_d & -\frac{1}{m_f + L_f} I_d \end{bmatrix} \begin{bmatrix} y - y_\star \\ u - u_\star \end{bmatrix} \geq 0.$$

By substituting (6.30)–(6.32) in (6.29) and rearranging terms, we can write

$$\dot{V} \leq e^\top \left(M_0 + a M_1 + \dot{a} M_2 \right) e. \tag{6.34}$$

The LMI in (6.27) implies

$$M_0 + a M_1 + \dot{a} M_2 \preceq -\sigma M_3. \tag{6.35}$$

Multiplying (6.35) on the left and right by $e^\top$ and e, respectively, and substituting the result back into (6.34) yields

$$\dot{V} \leq -\sigma e^\top M_3 e \leq 0,$$

where the second inequality follows from (6.33). The proof is now complete. ∎

According to Theorem 6.4, we can find the rate generating function $a(t)$ by solving the LMI in (6.27). More precisely, this LMI defines a first-order differential inequality on $a(t)$ whose solutions certify an $\mathcal{O}(1/a(t))$ convergence rate. The best lower bound on the convergence rate (i.e., the fastest growing $a(t)$) can be found by solving the following symbolic optimization problem:

$$\underset{\dot{a}(t) \geq 0, \sigma(t) \geq 0}{\text{maximize}} \ \dot{a}(t) \text{ subject to } \dot{a}(t) M_0(t) + a(t) M_1(t) + M_2(t) + \sigma(t) M_3(t) \preceq 0. \tag{6.36}$$

The optimality condition for (6.36) translates into a first-order ODE on $a(t)$. The solution to this ODE yields the best rate bound that can be certified using the Lyapunov function (6.25). In the following, we specialize the model in (6.23) to the particular case of the gradient flow (section 6.3.1) and its accelerated variant (section 6.3.2), where we will use Theorem 6.4 to derive the corresponding convergence rates.

6.3.1. Continuous-time gradient flow. Consider the following ODE for solving (4.1):

$$\dot{x}(t) = -\alpha \nabla f(x(t)), \quad x(0) \in \operatorname{dom} f, \tag{6.37}$$

where $\alpha > 0$. This ODE can be represented in the form (6.23) with $n = d$, and $(A, B, C) = (0_d, -\alpha I_d, I_d)$. By selecting $P(t) = p I_d$, $p \geq 0$, and applying the dimensionality reduction outlined in Remark 2, we obtain the following LMI:

$$\begin{bmatrix} -\frac{m_f}{2} \dot{a}(t) & \frac{1}{2} \dot{a}(t) - p\alpha \\ \frac{1}{2} \dot{a}(t) - p\alpha & -\alpha a(t) \end{bmatrix} + \sigma(t) \begin{bmatrix} \frac{-m_f L_f}{m_f + L_f} & \frac{1}{2} \\ \frac{1}{2} & \frac{-1}{m_f + L_f} \end{bmatrix} \preceq 0. \tag{6.38}$$

By elementary calculations, it can be verified that the solution to the corresponding optimization problem in (6.36) is $\sigma(t) = 0$, and $\dot{a}(t) = 2p + m_f \alpha a(t) + ((m_f \alpha a(t))^2 +$

$2pm_f\alpha a(t))^{1/2}$. Setting $p = 0$ and solving the latter ODE with initial condition $a(0) > 0$ yields $a(t) = a(0)\exp(2m_f\alpha t)$. Therefore, the gradient flow (6.37) exhibits the following convergence rate for strongly convex f:

$$f(x(t)) - f(x_\star) \leq e^{-2m_f\alpha t}(f(x(0)) - f(x_\star)).$$

Now we consider convex functions ($m_f = 0$) for which the LMI reduces to

$$\begin{bmatrix} 0 & \frac{1}{2}\dot{a}(t) - p\alpha + \frac{\sigma(t)}{2} \\ \frac{1}{2}\dot{a}(t) - p\alpha + \frac{\sigma(t)}{2} & -\alpha a(t) - \frac{\sigma(t)}{L_f} \end{bmatrix} \leq 0.$$

This LMI condition is equivalent to the condition $\dot{a}(t) \leq 2p\alpha - \sigma(t)$. Therefore, by setting $\sigma(t) = 0$, we obtain the optimal (fastest growing) $a(t)$, which satisfies the ODE $\dot{a}(t) = 2p\alpha$. Solving this ODE with the initial condition $a(0) \geq 0$, we obtain the rate bound

$$f(x(t)) - f(x_\star) \leq \frac{a(0)(f(x(0)) - f(x_\star)) + p\|x(0) - x_\star\|_2^2}{a(0) + 2p\alpha t}.$$

6.3.2. Continuous-time accelerated gradient flow. As a second case study, we consider the following second-order ODE for solving (4.1):

$$\ddot{x}(t) + \frac{r}{t}\dot{x}(t) + \nabla f(x(t)) = 0,\ r > 0. \tag{6.39}$$

This ODE is the continuous-time limit of Nesterov's accelerated scheme combined with an appropriate time-scaling [27]. The ODE (6.39) and its variants have been investigated extensively in the literature [1, 5, 2]. A state-space representation of (6.39) is given by

$$\begin{aligned} \dot{\xi}(t) &= \begin{bmatrix} -\frac{r-1}{t}I_d & \frac{r-1}{t}I_d \\ 0 & 0 \end{bmatrix}\xi(t) + \begin{bmatrix} 0 \\ -\frac{t}{r-1}I_d \end{bmatrix}\nabla f(y(t)), \\ y(t) &= \begin{bmatrix} I_d & 0 \end{bmatrix}\xi(t), \end{aligned} \tag{6.40}$$

where $\xi_1 = x$, $\xi_2 = x + t/(r-1)\dot{x}$ are the states, $\xi = [\xi_1^\top \ \ \xi_2^\top]^\top \in \mathbb{R}^{2d}$ is the state vector, and $y = \xi_1$ is the output. The fixed points of (6.40) are $(\xi_\star, y_\star, u_\star) = ([x_\star^\top \ x_\star^\top]^\top, x_\star, 0)$, where $x_\star \in \mathcal{X}_\star$ is any optimal solution satisfying $\nabla f(x_\star) = 0$.

We now analyze the convergence rate of (6.40) for convex functions ($m_f = 0$). By selecting $P(t) = \hat{P}I_d$, where $\hat{P} \in \mathbb{S}_{++}^2$ is time-invariant, and applying the dimensionality reduction of Remark 2, we arrive at the 3×3 LMI

$$\begin{bmatrix} -\frac{2(r-1)p_{11}}{t} & \frac{(r-1)(p_{11}-p_{21})}{t} & \frac{\dot{a}(t)+\sigma}{2} - \frac{(r-1)a(t)}{2t} - \frac{tp_{12}}{r-1} \\ \frac{(r-1)(p_{11}-p_{21})}{t} & \frac{2(r-1)p_{21}}{t} & \frac{(r-1)a(t)}{2t} - \frac{tp_{22}}{r-1} \\ \frac{\dot{a}(t)+\sigma}{2} - \frac{(r-1)a(t)}{2t} - \frac{tp_{12}}{r-1} & \frac{(r-1)a(t)}{2t} - \frac{tp_{22}}{r-1} & -\frac{\dot{a}(t)}{2L_f} - \frac{\sigma}{L_f} \end{bmatrix} \preceq 0,$$

where $\hat{P} = [p_{ij}]$. A simple analytic solution to the above LMI can be obtained by choosing $p_{11} = p_{12} = p_{21} = 0$. With this particular choice, the LMI simplifies to the conditions

$$\frac{\dot{a}(t) + \sigma(t)}{2} - \frac{(r-1)a(t)}{2t} = 0, \quad p_{22} = \left(\frac{r-1}{t}\right)^2 \frac{a(t)}{2}. \tag{6.41}$$

Using the assumption that p_{22} is constant, together with the condition $\sigma(t) \geq 0$, the above conditions are equivalent to $a(t) = ct$ and $p_{22} = c(r-1)^2/2$, where $c > 0$ and $r \geq 0$. Using Theorem 6.4, we obtain the convergence rate

$$f(x(t)) - f(x_\star) \leq \frac{(r-1)^2 \|x(0) - x_\star\|_2^2}{2t^2}, \quad r \geq 3.$$

This convergence result agrees with [27, Theorem 5]. More generally, by allowing the matrix $P(t)$ to be time-dependent, the LMI (6.27) can be used to directly answer the following question: How does the convergence rate of the accelerated gradient flow change with the parameter r?

6.4. Algorithm design. In this subsection, we briefly explore algorithm tuning and design using the developed LMI framework. In particular, we consider robustness as a design criterion. It has been shown in [9, 19, 8] that there is a trade-off between an algorithm's rate of convergence and its robustness against inexact information about the oracle. In particular, fast methods such as Nesterov's accelerated method require first-order information with higher accuracy than standard gradient methods to obtain a solution with a given accuracy [9]. To explain this trade-off in our framework, we recall the proof of Theorem 3.1, in which we showed that the LMI

$$M_k^0 + a_k M_k^1 + (a_{k+1} - a_k) M_k^2 + \sigma_k M_k^3 \preceq 0 \quad \text{for all } k \tag{6.42}$$

ensures that the Lyapunov function satisfies

$$V_k(\xi_{k+1}) \leq V(\xi_k) - \sigma_k e_k^\top M_k^3 e_k \quad \text{for all } k. \tag{6.43}$$

In view of (6.43), the nonnegative term $\sigma_k e_k^\top M_k^3 e_k$ provides an additional stability margin and hence makes the algorithm robust against uncertainties in the algorithm or underlying assumptions (for instance, the value of m_f or L_f). Based on this observation, we propose the LMI

$$M_k^0 + a_k M_k^1 + (a_{k+1} - a_k) M_k^2 + \sigma_k M_k^3 + S_k \preceq 0 \quad \text{for all } k, \tag{6.44}$$

where S_k is any symmetric matrix that satisfies $e_k^\top S_k e_k \geq 0$ for all k. In particular, any $S_k \succeq 0$ is a valid choice. By revisiting the proof of Theorem 3.1, we see that the feasibility of the above LMI imposes the stricter condition

$$V_{k+1}(\xi_{k+1}) \leq V_k(\xi_k) - e_k^\top (\sigma_k M_k^3 + S_k) e_k, \quad e_k^\top S_k e_k \geq 0, \tag{6.45}$$

on the decrement of the Lyapunov function. The LMI in (6.44) is the robust counterpart of (3.7). Now we can use (6.44) to search for the parameters of the algorithm, considering S_k as a tuning parameter that makes the trade-off between robustness and rate of convergence.

6.4.1. Robust gradient method. As an illustrative example, consider the gradient method applied to $f \in \mathcal{F}(m_f, L_f)$. Consider the robust counterpart of the LMI in (4.14):

$$\begin{bmatrix} p - \rho^2 p & -hp \\ -hp & h^2 p \end{bmatrix} + \lambda \begin{bmatrix} \frac{-m_f L_f}{m_f + L_f} & \frac{1}{2} \\ \frac{1}{2} & \frac{-1}{m_f + L_f} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & s \end{bmatrix} \preceq 0, \quad s \geq 0. \tag{6.46}$$

This LMI is homogeneous in (p, λ, s). We can hence assume $p = 1$. Using the Schur complement, the above LMI is equivalent to

$$\begin{bmatrix} -\rho^2 - \lambda\frac{m_f L_f}{m_f + L_f} & \frac{\lambda}{2} & 1 \\ \frac{\lambda}{2} & -\frac{\lambda}{m_f + L_f} + s & -h \\ 1 & -h & -1 \end{bmatrix} \preceq 0, \tag{6.47}$$

which is now an LMI in (ρ^2, λ, h, s). By treating s as a tuning parameter for robustness and minimizing the convergence factor ρ^2 over (λ, h), we can design stepsizes that yield the best convergence rate for a given level of robustness. Conversely, by treating ρ^2 as a tuning parameter and maximizing s over (λ, h), we can design stepsizes which yield the largest robustness margin for a desired convergence rate.

6.4.2. Robust Nesterov's accelerated method. As our design experiment, we consider Nesterov's accelerated method applied to a strongly convex f:

$$\begin{aligned} x_{k+1} &= y_k - h\nabla f(y_k), \\ y_k &= x_k + \beta(x_k - x_{k-1}). \end{aligned} \tag{6.48}$$

Specifically, we consider the robust version of the LMI in (3.16), where the matrices $M_k^i,\ i \in \{0, 1, 2, 3\}$, are given as in (4.20), and the robustness matrix is chosen as $sI_3,\ s \geq 0$. For a given condition number $\kappa_f = \frac{L_f}{m_f}$ and robustness margin s, we use the LMI to compute the convergence factor ρ on the grid $(h, \beta) \in [0\ \frac{2}{L_f}] \times [0\ 1]$. See subsection 4.2.

In Figure 5, we plot the contour plots of ρ for $s = 0$ and $s = 0.01$, respectively. The condition number is fixed at $L_f/m_f = 10$. We observe that when s is nonzero, the parameters of the robust algorithm shift towards smaller stepsizes and higher momentum coefficients, leading to higher robustness and lower convergence rates.

7. Concluding remarks. In this paper, we have developed a linear matrix inequality (LMI) framework, built on the notion of integral quadratic constraints (IQCs) from robust control theory and Lyapunov stability, to certify both exponential and subexponential convergence rates of first-order optimization algorithms. To this end, we proposed a class of time-varying Lyapunov functions that are suitable for generating nonconservative convergence rates in addition to proving stability. We showed that the developed LMI can often be solved in closed form. In particular, we applied the technique to the gradient method, the proximal gradient method, and their accelerated extensions to recover the known analytical upper bounds on their performance. Furthermore, we showed that numerical schemes can also be used to solve the LMI.

In this paper, we have only used pointwise IQCs to model nonlinearities. More complicated IQCs, such as "off-by-one" IQCs, have shown to be fruitful in improving numerical rate bounds in strongly convex settings [19]. One direction for future work would be to use these IQCs in tandem with the Lyapunov function proposed in this paper to further improve the numerical bounds in nonstrongly convex problems. Obtaining better worst-case bounds is useful in a variety of applications, such as model predictive control (MPC). MPC is a sequential optimization-based control scheme, which is particularly useful for constrained and nonlinear control tasks. Implementation of MPC requires the solution of a constrained optimization problem in real time within the sampling period to a specific accuracy determined from stability considerations [26]. It is thus important to bound a priori, in a nonconservative manner, the

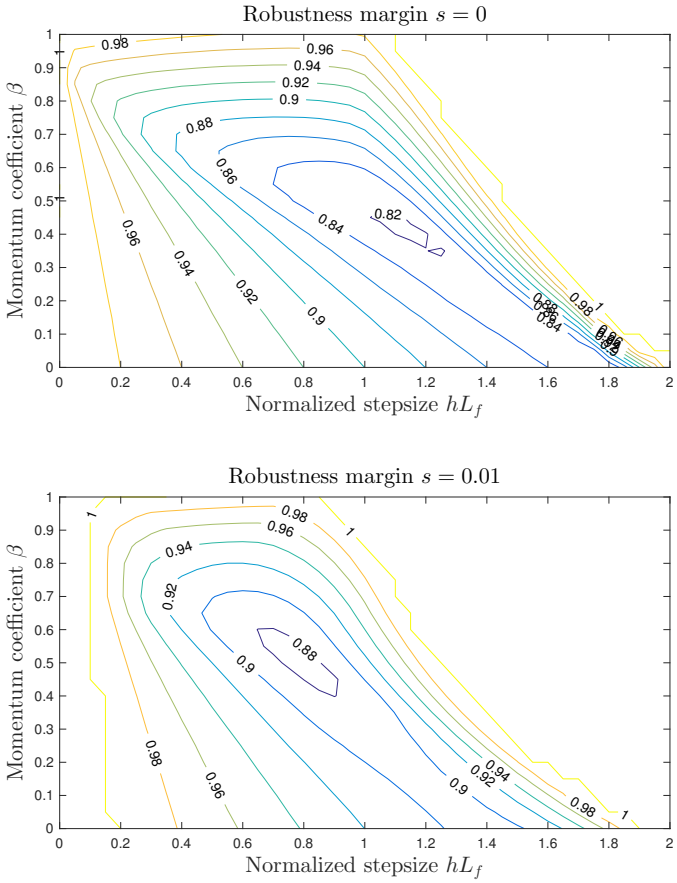

FIG. 5. *Plot of convergence rate ρ of Nesterov's accelerated method as a function of stepsize h and momentum parameter β, and for two values of the robustness parameter s. Higher values of s increase the robustness of the algorithm at the expense of reduced convergence rate.*

number of iterations needed for a specified accuracy. Improving the numerical rate bounds will allow us to optimize this bound for every problem instance. More generally, having a nonconservative estimation of convergence rate allows us to compare different algorithms, which must be done by extensive simulations otherwise. We will pursue these applications in future work.

Appendix A. Symbolic convergence rates for the gradient method. The LMI in (4.15) with $p = 1$ along with the condition $a_{k+1} \geq a_k$ is equivalent to the inequalities

$$a_{k+1} \geq a_k, \tag{A.1}$$

$$\left(\frac{L_f h^2}{2} - h\right) a_{k+1} + h^2 - \frac{\sigma}{L_f} \leq 0, \tag{A.2}$$

$$-\left(\frac{a_{k+1} - a_k - 2h + \sigma}{2}\right)^2 \geq 0. \tag{A.3}$$

The last inequality implies $a_{k+1} = a_k + 2h - \sigma$. Assuming $a_0 = 0$ and solving for a_k, we obtain $a_k = (2h - \sigma)k$. Therefore, the fastest convergence rate corresponds to the smallest σ. By substituting a_k in (A.1) and (A.2), we obtain

$$2h - \sigma \geq 0, \quad \left(\frac{L_f h^2}{2} - h\right)(2h - \sigma)(k+1) + h^2 - \frac{\sigma}{L_f} \leq 0. \tag{A.4}$$

Since the second inequality must hold for all $k \geq 0$, we must have that $L_f h^2/2 - h \leq 0$ or, equivalently, $0 \leq h \leq 2/L_f$. Under this condition, it suffices to ensure that the second inequality in (A.4) holds for $k = 0$. This leads to

$$\max\left(0, \frac{(L_f h)(L_f h - 1)(2h)}{(L_f h)^2 - 2(L_f h) + 2}\right) \leq \sigma \leq 2h. \tag{A.5}$$

Therefore, the optimal (minimum) σ is

$$\sigma_{opt} = \begin{cases} 0 & \text{if } 0 \leq hL_f \leq 1, \\ \dfrac{(L_f h)(L_f h - 1)(2h)}{(L_f h)^2 - 2(L_f h) + 2} & \text{if } 1 < hL_f \leq 2. \end{cases} \tag{A.6}$$

By substituting all the parameters in (3.4), we obtain

$$f(x_k) - f(x_\star) \leq \frac{\|x_0 - x_\star\|_2^2}{(2h - \sigma_{opt})k}, \tag{A.7}$$

which is the same as (4.16).

Appendix B. Proof of Proposition 5.1.

Proof of part 1. Since g is nondifferentiable and convex, it follows from the discussion in sections 6.1.1 and 6.1.2 that $\Pi_{g,h}$ is firmly nonexpansive and hence Lipschitz continuous with Lipschitz parameter equal to one. Further, it is well known that the map $x \mapsto x - h\nabla f(x)$ is Lipschitz continuous with Lipschitz constant $\gamma_f = \max\{|1 - hL_f|, |1 - hm_f|\}$; see, for example, [4] for a proof. Therefore, the composition $\Pi_{g,h}(x - h\nabla f(x))$ is Lipschitz continuous with parameter γ_f. In other words, we can write

$$\|\Pi_{g,h}(x - h\nabla f(x)) - \Pi_{g,h}(x_\star - h\nabla f(x_\star))\|_2^2 \leq \gamma_f^2 \|x - x_\star\|_2^2.$$

Making the substitution $\Pi_{g,h}(x - h\nabla f(x)) = x - h\phi_h(x)$, completing the squares, and rearranging terms yield

$$\begin{bmatrix} x - x_\star \\ \phi_h(x) - \phi_h(x_\star) \end{bmatrix}^\top \begin{bmatrix} \frac{1}{2h}(\gamma_f^2 - 1)I_d & \frac{1}{2}I_d \\ \frac{1}{2}I_d & -\frac{h}{2}I_d \end{bmatrix} \begin{bmatrix} x - x_\star \\ \phi_h(x) - \phi_h(x_\star) \end{bmatrix} \geq 0.$$

Proof of part 2. First, note that the optimality condition of the proximal operator, defined in (5.3), is

$$0 \in \partial g(\Pi_{g,h}(w)) + \frac{1}{h}(\Pi_{g,h}(w) - w)$$

or, equivalently,

$$0 = T_g(\Pi_{g,h}(w)) + \frac{1}{h}(\Pi_{g,h}(w) - w), \ T_g \in \partial g, \tag{B.1}$$

where $T_g(w)$ denotes a subgradient of g at w. On the other hand, by the definition of the generalized gradient mapping in (5.4), we have that

$$\Pi_{g,h}(y - h\nabla f(y)) = y - h\phi_h(y). \tag{B.2}$$

Substituting (B.2) and $w = y - h\nabla f(y)$ in (B.1), we can equivalently write $\phi_h(y)$ as

$$\phi_h(y) = \nabla f(y) + T_g(y - h\phi_h(y)). \tag{B.3}$$

Consider the points $x, y, z \in \operatorname{dom} f$. We can write

$$\begin{aligned} f(z) - f(y) &\le \nabla f(y)^\top (z-y) + \frac{L_f}{2}\|z-y\|_2^2, \\ f(y) - f(x) &\le \nabla f(y)^\top (y-x) - \frac{m_f}{2}\|y-x\|_2^2. \end{aligned}$$

In the first and second inequality, we have used Lipschitz continuity and strong convexity, respectively. Adding both sides yields

$$f(z) - f(x) \le \nabla f(y)^\top (z-x) + \frac{L_f}{2}\|z-y\|_2^2 - \frac{m_f}{2}\|y-x\|_2^2. \tag{B.4}$$

Further, since g is convex, we can write

$$g(z) - g(x) \le T_g(z)^\top (z-x), \quad T_g(z) \in \partial g(z), \quad x, z \in \operatorname{dom} g. \tag{B.5}$$

Adding both sides of (B.4) and (B.5) for all $x, z \in \operatorname{dom} f \cap \operatorname{dom} g$, $y \in \operatorname{dom} f$ and making the substitutions $z = y - h\phi_h(y)$ and (B.3) yield (5.6).

Proof of part 3. Suppose $\phi_h(y) = 0$ for some $y \in \operatorname{dom} \phi_h$. It then follows from (B.3) that $0 = \nabla f(y) + T_g(y)$ or, equivalently, $0 \in \nabla f(y) + \partial g(y)$. This implies that $y \in \mathcal{X}_\star$, according to (5.2). Conversely, suppose $y \in \mathcal{X}_\star$. We therefore have $\nabla f(y) = -T_g(y)$. Substituting this in (B.3) yields $\phi_h(y) = T_g(y - h\phi_h(y)) - T_g(y)$. Since T_g is monotone, we can write

$$0 \le (T_g(y - h\phi_h(y)) - T_g(y))^\top (y - h\phi_h(y) - y) = -h\|\phi_h(y)\|_2^2 \quad \text{for all } h.$$

Therefore, we must have that $\phi_h(y) = 0$. The proof is now complete. ∎

Appendix C. Proof of Lemma 5.2. In order to bound $F(x_{k+1}) - F(x_k)$ and $F(x_{k+1}) - F(x_\star)$, we use the inequality

$$F(y - h\phi_h(y)) - F(x) \le \phi_h(y)^\top (y-x) - \frac{m_f}{2}\|y-x\|_2^2 + \left(\frac{1}{2}L_f h^2 - h\right)\|\phi_h(y)\|_2^2, \tag{C.1}$$

which we proved in Proposition 5.1. Specifically, we substitute $(x, y) = (x_\star, y_k)$ in (C.1) to get

$$\begin{aligned} F(x_{k+1}) - F(x_\star) &\le (u_k - u_\star)^\top (y_k - y_\star) + \left(\frac{L_f h^2}{2} - h\right)\|u_k - u_\star\|_2^2 - \frac{m_f}{2}\|y_k - y_\star\|_2^2 \\ &= \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & (\frac{1}{2}L_f h^2 - h) \end{bmatrix} \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix} \\ &= \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix}^\top \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & (\frac{1}{2}L_f h^2 - h) \end{bmatrix} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix} \\ &= e_k^\top M_k^2 e_k, \end{aligned}$$

where we have used the identities $u_\star = \phi_h(y_\star) = 0$ and $y_k - y_\star = C_k(\xi_k - \xi_\star)$. Similarly, in (C.1) we substitute $(x, y) = (x_k, y_k)$ to obtain

(C.2)

$$\begin{aligned}
&F(x_{k+1}) - F(x_k) \le (u_k - u_\star)^\top (y_k - x_k) + \left(\frac{1}{2}L_f h^2 - h\right)\|u_k - u_\star\|_2^2 - \frac{m_f}{2}\|y_k - x_k\|_2^2 \\
&= \begin{bmatrix} y_k - x_k \\ u_k - u_\star \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & (\frac{1}{2}L_f h^2 - h) \end{bmatrix} \begin{bmatrix} y_k - x_k \\ u_k - u_\star \end{bmatrix} \\
&= \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix}^\top \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix}^\top \begin{bmatrix} -\frac{m_f}{2} & \frac{1}{2} \\ \frac{1}{2} & (\frac{1}{2}L_f h^2 - h) \end{bmatrix} \begin{bmatrix} C_k - E_k & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix} \\
&= e_k^\top M_k^1 e_k,
\end{aligned}$$

where we have used $x_\star = y_\star$ and $y_k - x_k = (C_k - E_k)(\xi_k - \xi_\star)$ to obtain the second equality. Finally, by Proposition 5.1 $u_k = \phi_h(y_k)$ satisfies the pointwise IQC defined by $(Q_{\phi_h}, x_\star, \phi_h(x_\star))$. Therefore, we can write

(C.3)
$$\begin{aligned}
e_k^\top M_k^3 e_k &= \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix}^\top \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix}^\top Q_{\phi_h} \begin{bmatrix} C_k & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \xi_k - \xi_\star \\ u_k - u_\star \end{bmatrix} \\
&= \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix}^\top Q_{\phi_h} \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix} \\
&\ge 0,
\end{aligned}$$

where we have used the identity $y_k - y_\star = C_k(\xi_k - \xi_\star)$ to obtain the second inequality. The proof is complete. ∎

REFERENCES

[1] F. Alvarez, *On the minimizing property of a second order dissipative system in Hilbert spaces*, SIAM J. Control Optim., 38 (2000), pp. 1102–1119, https://doi.org/10.1137/S0363012998335802.

[2] H. Attouch, J. Peypouquet, and P. Redont, *Fast convex optimization via inertial dynamics with Hessian driven damping*, J. Differential Equations, 261 (2016), pp. 5734–5783, https://doi.org/10.1016/j.jde.2016.08.020.

[3] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202, https://doi.org/10.1137/080716542.

[4] D. P. Bertsekas, *Convex Optimization Algorithms*, Athena Scientific, 2015.

[5] A. Cabot, H. Engler, and S. Gadat, *On the long time behavior of second order differential equations with asymptotically small dissipation*, Trans. Amer. Math. Soc., 361 (2009), pp. 5983–6017, https://doi.org/10.1090/S0002-9947-09-04785-0.

[6] A. Cherukuri, E. Mallada, S. Low, and J. Cortés, *The role of convexity in saddle-point dynamics: Lyapunov function and robustness*, IEEE Trans. Automat. Control, 63 (2018), pp. 2449–2464, https://doi.org/10.1109/TAC.2017.2778689.

[7] P. L. Combettes and J.-C. Pesquet, *Proximal splitting methods in signal processing*, in Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer, 2011, pp. 185–212, https://doi.org/10.1007/978-1-4419-9569-8_10.

[8] S. Cyrus, B. Hu, B. Van Scoy, and L. Lessard, *A robust accelerated optimization algorithm for strongly convex functions*, in Proceedings of the 2018 Annual American Control Conference (ACC), IEEE, 2018, pp. 1376–1381, https://doi.org/10.23919/ACC.2018.8430824.

[9] O. Devolder, F. Glineur, and Y. Nesterov, *First-order methods of smooth convex optimization with inexact oracle*, Math. Programming, 146 (2014), pp. 37–75, https://doi.org/10.1007/s10107-013-0677-5.

[10] Y. Drori and M. Teboulle, *Performance of first-order methods for smooth convex minimization: A novel approach*, Math. Programming, 145 (2014), pp. 451–482, https://doi.org/10.1007/s10107-013-0653-0.

[11] M. Fazlyab, A. Ribeiro, M. Morari, and V. M. Preciado, *A dynamical systems perspective to convergence rate analysis of proximal algorithms*, in Proceedings of the 55th Annual

Allerton Conference on Communication, Control, and Computing, IEEE, 2017, pp. 354–360, https://doi.org/10.1109/ALLERTON.2017.8262759.

[12] D. Feijer and F. Paganini, *Stability of primal–dual gradient dynamics and applications to network optimization*, Automatica, 46 (2010), pp. 1974–1981, https://doi.org/10.1016/j.automatica.2010.08.011.

[13] M. Hardt, T. Ma, and B. Recht, *Gradient Descent Learns Linear Dynamical Systems*, preprint, https://arxiv.org/abs/1609.05191, 2016.

[14] E. Hazan, K. Levy, and S. Shalev-Shwartz, *Beyond convexity: Stochastic quasi-convex optimization*, in Advances in Neural Information Processing Systems, NIPS Proc. 28, C. Cortes et al., eds., Neural Information Processing Systems Foundation, Inc., 2015, pp. 1594–1602.

[15] B. Hu and L. Lessard, *Control interpretations for first-order optimization methods*, in Proceedings of the 2017 American Control Conference, IEEE, 2017, pp. 3114–3119, https://doi.org/10.23919/ACC.2017.7963426.

[16] B. Hu and L. Lessard, *Dissipativity theory for Nesterov's accelerated method*, in Proceedings of the Thirty-Fourth International Conference on Machine Learning, Proc. Mach. Learn. Res. 70, International Machine Learning Society (IMLS), 2017, pp. 1549–1557.

[17] H. Karimi, J. Nutini, and M. Schmidt, *Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition*, in Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2016, pp. 795–811, https://doi.org/10.1007/978-3-319-46128-1_50.

[18] D. Kim and J. A. Fessler, *Optimized first-order methods for smooth convex minimization*, Math. Programming, 159 (2016), pp. 81–107, https://doi.org/10.1007/s10107-015-0949-3.

[19] L. Lessard, B. Recht, and A. Packard, *Analysis and design of optimization algorithms via integral quadratic constraints*, SIAM J. Optim., 26 (2016), pp. 57–95, https://doi.org/10.1137/15M1009597.

[20] A. Megretski and A. Rantzer, *System analysis via integral quadratic constraints*, IEEE Trans. Automat. Control, 42 (1997), pp. 819–830, http://doi.org/10.1109/9.587335.

[21] I. Necoara, Yu. Nesterov, and F. Glineur, *Linear convergence of first order methods for non-strongly convex optimization*, Math. Program., 2018, pp. 1–39, https://doi.org/10.1007/s10107-018-1232-1.

[22] Y. Nesterov, *A method of solving a convex programming problem with convergence rate o (1/k2)*, Soviet Math. Dokl., 27 (1983), pp. 372–376.

[23] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, Appl. Optim. 87, Kluwer Academic Publishers, 2013.

[24] R. Nishihara, L. Lessard, B. Recht, A. Packard, and M. I. Jordan, *A General Analysis of the Convergence of ADMM*, preprint, https://arxiv.org/abs/1502.02009, 2015.

[25] B. Polyak, *Some methods of speeding up the convergence of iteration methods*, USSR Comput. Math. Math. Phys., 4 (1964), pp. 1–17, https://doi.org/10.1016/0041-5553(64)90137-5.

[26] 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 Trans. Automat. Control, 57 (2012), pp. 1391–1403, https://doi.org/10.1109/TAC.2011.2176389.

[27] W. Su, S. Boyd, and E. J. Candès, *A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights*, J. Mach. Learn. Res., 17 (2016), pp. 1–43.

[28] A. B. Taylor, J. M. Hendrickx, and F. Glineur, *Exact Worst-Case Convergence Rates of the Proximal Gradient Method for Composite Convex Minimization*, J. Optim. Theory Appl., 178 (2018), pp. 455-476, https://doi.org/10.1007/s10957-018-1298-1.

[29] A. B. Taylor, J. M. Hendrickx, and F. Glineur, *Smooth strongly convex interpolation and exact worst-case performance of first-order methods*, Math. Programming, 161 (2017), pp. 307–345, https://doi.org/10.1007/s1010.

[30] J. Wang and N. Elia, *Control approach to distributed optimization*, in Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing, IEEE, 2010, pp. 557–561, https://doi.org/10.1109/ALLERTON.2010.5706956.

[31] J. Wang and N. Elia, *A control perspective for centralized and distributed convex optimization*, in Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference, IEEE, 2011, pp. 3800–3805, https://doi.org/10.1109/CDC.2011.6161503.

[32] A. Wibisono, A. C. Wilson, and M. I. Jordan, *A variational perspective on accelerated methods in optimization*, Proc. Natl. Acad. Sci. USA, 113 (2016), pp. E7351–E7358, https://doi.org/10.1073/pnas.1614734113.

[33] A. C. Wilson, B. Recht, and M. I. Jordan, *A Lyapunov Analysis of Momentum Methods in Optimization*, preprint, https://arxiv.org/abs/1611.02635, 2016.

[34] V. Yakubovich, *Frequency conditions for the absolute stability of control systems with several nonlinear or linear nonstationary blocks*, Avtomat. i Telemekh., 6 (1967), pp. 5–30 (in Russian).