Analysis and Design of Optimization Algorithms via Integral Quadratic Constraints

Laurent Lessard   Benjamin Recht   Andrew Packard

SIAM Journal on Optimization, vol. 26, no. 1, pp. 57–95, Jan. 2016

(Oct. 16, 2017) this version fixes typos present in the published version

Abstract

This manuscript develops a new framework to analyze and design iterative optimization algorithms built on the notion of Integral Quadratic Constraints (IQC) from robust control theory. IQCs provide sufficient conditions for the stability of complicated interconnected systems, and these conditions can be checked by semidefinite programming. We discuss how to adapt IQC theory to study optimization algorithms, proving new inequalities about convex functions and providing a version of IQC theory adapted for use by optimization researchers. Using these inequalities, we derive numerical upper bounds on convergence rates for the Gradient method, the Heavy-ball method, Nesterov's accelerated method, and related variants by solving small, simple semidefinite programming problems. We also briefly show how these techniques can be used to search for optimization algorithms with desired performance characteristics, establishing a new methodology for algorithm design.

1 Introduction

Convex optimization algorithms provide a powerful toolkit for robust, efficient, large-scale optimization algorithms. They provide not only effective tools for solving optimization problems, but are guaranteed to converge to accurate solutions in provided time budgets [23, 26], are robust to errors and time delays [22, 41], and are amendable to declarative modeling that decouples the algorithm design from the problem formulation [2, 8, 16]. However, as we push up against the boundaries of the convex analysis framework, try to build more complicated models, and aim to deploy optimization systems in highly complex environments, the mathematical guarantees of convexity start to break. The standard proof techniques for analyzing convex optimization rely on deep insights by experts and are devised on an algorithm-by-algorithm basis. It is thus not clear how to extend the toolkit to more diverse scenarios where multiple objectives—such as robustness, accuracy, and speed—need to be delicately balanced.

This paper marks an attempt at providing a systematized approach to the design and analysis optimization algorithms using techniques from control theory. Our strategy is to adapt the notion of an *integral quadratic constraint* from robust control theory [19]. These constraints link sequences of inputs and outputs of operators, and are ideally suited to proving algorithmic convergence. We will see that for convex functions, we can derive these constraints using only the standard first-order characterization of convex functions, and that these inequalities will be sufficient to reduce the analysis of first-order methods to the solution of a very small semidefinite program. Our IQC framework puts the analysis of algorithms in a unified proof framework, and enables new analyses of algorithms by minor perturbations of existing proofs. This new system aims to simplify and automate the analysis of optimization programs, and perhaps to open new directions for algorithm design.

Our methods are inspired by the recent work of Drori and Teboulle [7]. In their manuscript, the authors propose writing down the first-order convexity inequality for all steps of an algorithmic procedure. They then derive a semidefinite program that analytically verifies very tight bounds for the convergence rate for the Gradient method, and numerically precise bounds for convergence of Nesterov's method and other first-order methods. The main drawback of the Drori and Teboulle approach is that the size of the semidefinite program scales with the number of time steps desired. Thus, it becomes computationally laborious to analyze algorithms that require more than a few hundred iterations.

Integral quadratic constraints will allow us to circumvent this issue. A typical example of one of our semidefinite programs might have a 3×3 positive semidefinite decision variable, 3 scalar variables, a 5×5 semidefinite cone constraint, and 4 scalar constraints. Such a problem can be solved in less than 10 milliseconds on a laptop with standard solvers.

We are able to analyze a variety of methods in our framework. We show that our framework recovers the standard rates of convergence for the Gradient method applied to strongly convex functions. We show that we can numerically estimate the performance of Nesterov's method. Indeed, our analysis provides slightly sharper bounds than Nesterov's proof. We show how our system fails to certify the stability of the popular Heavy-ball method of Polyak for strongly convex functions whose condition ratio is larger than 18. Based on this analysis, we are able to construct a one-dimensional strongly convex function whose condition ratio is 25 and prove analytically that the Heavy-ball method fails to find the global minimum of this function. This suggests that our tools can also be used as a way to guide the construction of counterexamples.

We show that our methods extend immediately to the projected and proximal variants of all the first order methods we analyze. We also show how to extend our analysis to functions that are convex but not strongly convex, and provide bounds on convergence that are within a logarithmic factor of the best upper bounds. We also demonstrate that our methods can bound convergence rates when the gradient is perturbed by relative deterministic noise. We show how different parameter settings lead to very different degradations in performance bounds as the noise increases.

Finally, we turn to algorithm *design*. Since our semidefinite program takes as input the parameters of our iterative scheme, we can search over these parameters. For simple two-step methods, our algorithms are parameterized by 3 parameters, and we show how we can derive first-order methods that achieve nearly the same rate of convergence as Nesterov's accelerated method but are more robust to noise.

The manuscript is organized as follows. We begin with a discussion of discrete-time dynamical system and how common optimization algorithms can be viewed as feedback interconnections between a known *linear* system with an uncertain *nonlinear* component. We then turn to show how quadratic Lyapunov functions can be used to certify rates of convergence for optimization problems and can be found by semidefinite programming. This immediately leads to the notion of an integral quadratic constraint. Another contribution of this work is a new form of IQC analysis geared specifically toward rate-of-convergence conclusions, and accessible to optimization researchers. We also discuss their history in robust control theory and how they can be derived. With these basic IQCs in hand, we then turn to analyzing the Gradient method and Nesterov method, their projected and proximal variants, and their robustness to noise. We discuss one possible brute-force technique for designing new algorithms, and how we can outperform existing methods. Finally, we conclude with many directions for future work.

1.1 Notation and conventions

Common matrices. The $d \times d$ identity matrix and zero matrix are denoted I_d and 0_d, respectively. Subscripts are omitted when they are to be inferred by context.

Norms and sequences. We define ℓ_{2e}^n to be the set of all one-sided sequences $x : \mathbb{N} \to \mathbb{R}^n$. We sometimes omit n and simply write ℓ_{2e} when the superscript is clear from context. The notation $\|\cdot\| : \mathbb{R}^n \to \mathbb{R}$ denotes the standard 2-norm. The subset $\ell_2 \subset \ell_{2e}$ consists of all square-summable sequences. In other words, $x \in \ell_2$ if and only if $\sum_{k=0}^{\infty} \|x_k\|^2$ is convergent.

Convex functions. For a given $0 < m < L$, we define $S(m, L)$ to be the set of functions $f : \mathbb{R}^d \to \mathbb{R}$ that are continuously differentiable, strongly convex with parameter m, and have Lipschitz gradients with parameter L. In other words, f satisfies

$$m\|x-y\|^2 \le (\nabla f(x) - \nabla f(y))^\mathsf{T}(x-y) \le L\|x-y\|^2 \quad \text{for all } x, y \in \mathbb{R}^d$$

We call $\kappa := L/m$ the *condition ratio* of $f \in S(m, L)$. We adopt this terminology to distinguish the condition ratio of a function from the related concept of *condition number* of a matrix. The connection is that if f is twice differentiable, we have the bound: $\operatorname{cond}(\nabla^2 f(x)) \le \kappa$ for all $x \in \mathbb{R}^d$, where $\operatorname{cond}(\cdot)$ is the condition number.

Kronecker product The Kronecker product of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ is denoted $A \otimes B \in \mathbb{R}^{mp\times nq}$ and given by:

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

Two useful properties of the Kronecker product are that $(A \otimes B)^\mathsf{T} = A^\mathsf{T} \otimes B^\mathsf{T}$ and that $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$ whenever the matrix dimensions are such that the products AC and BD make sense.

2 Optimization algorithms as dynamical systems

A linear dynamical system is a set of recursive linear equations of the form

$$\xi_{k+1} = A\xi_k + Bu_k \tag{2.1a}$$
$$y_k = C\xi_k + Du_k\,. \tag{2.1b}$$

At each timestep $k = 0, 1, \dots$, $u_k \in \mathbb{R}^d$ is the *input*, $y_k \in \mathbb{R}^d$ is the *output*, and $\xi_k \in \mathbb{R}^m$ is the *state*. We can write the dynamical system (2.1) compactly by stacking the matrices into a block using the notation

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right].$$

We can connect this linear system in *feedback* with a nonlinearity ϕ by defining the rule

$$\xi_{k+1} = A\xi_k + Bu_k \tag{2.2a}$$
$$y_k = C\xi_k + Du_k \tag{2.2b}$$
$$u_k = \phi(y_k)\,. \tag{2.2c}$$

In this case, the output is transformed by the map $\phi : \mathbb{R}^d \to \mathbb{R}^d$ and is then used as the input to the linear system.

In this paper, we will be interested in the case when the interconnected nonlinearity has the form $\phi(y) = \nabla f(y)$ where $f \in S(m, L)$. In particular, we will consider algorithms designed to solve the optimization problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}}\; f(x) \tag{2.3}$$

as dynamical systems and see how this new viewpoint can give us insights into convergence analysis. Section 5.3 considers variants of (2.3) where the decision variable x is constrained or f is non-smooth.

Standard first order methods such as the Gradient method, Heavy-ball method, and Nesterov's accelerated method, can all be cast in the form (2.2). In all cases, the nonlinearity is the mapping $\phi(y) = \nabla f(y)$. The state transition matrices A, B, C, D differ for each algorithm. The Gradient method can be expressed as

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{c|c} I_d & -\alpha I_d \\ \hline I_d & 0_d \end{array}\right]. \tag{2.4}$$

To verify this, substitute (2.4) into (2.2) and obtain

$$\begin{aligned} \xi_{k+1} &= \xi_k - \alpha u_k \\ y_k &= \xi_k \\ u_k &= \nabla f(y_k) \end{aligned}$$

Eliminating y_k and u_k and renaming ξ to x yields

$$x_{k+1} = x_k - \alpha \nabla f(x_k)$$

which is the familiar Gradient method with constant stepsize. Nesterov's accelerated method for strongly convex functions is given by the dynamical system

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} (1+\beta)I_d & -\beta I_d & -\alpha I_d \\ I_d & 0_d & 0_d \\ \hline (1+\beta)I_d & -\beta I_d & 0_d \end{array}\right] \tag{2.5}$$

Verifying that (2.5) is equivalent to Nesterov's method takes only slightly more effort than it did for the Gradient method. Substituting (2.5) into (2.2) now yields

$$\xi_{k+1}^{(1)} = (1+\beta)\xi_k^{(1)} - \beta\xi_k^{(2)} - \alpha u_k \tag{2.6a}$$

$$\xi_{k+1}^{(2)} = \xi_k^{(1)} \tag{2.6b}$$

$$y_k = (1+\beta)\xi_k^{(1)} - \beta\xi_k^{(2)} \tag{2.6c}$$

$$u_k = \nabla f(y_k) \tag{2.6d}$$

Note that (2.6b) asserts that the partial state $\xi^{(2)}$ is a delayed version of the state $\xi^{(1)}$. Substituting (2.6b) into (2.6a) gives the simplified system

$$\begin{aligned} \xi_{k+1}^{(1)} &= (1+\beta)\xi_k^{(1)} - \beta\xi_{k-1}^{(1)} - \alpha u_k \\ y_k &= (1+\beta)\xi_k^{(1)} - \beta\xi_{k-1}^{(1)} \\ u_k &= \nabla f(y_k) \end{aligned}$$

Eliminating u_k and renaming $\xi^{(1)}$ to x yields the common form of Nesterov's method

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

Note that other variants of this algorithm exist for which the α and β parameters are updated at each iteration. In this paper, we restrict our analysis to the constant-parameter version above. The Heavy-ball method is given by

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} (1+\beta)I_d & -\beta I_d & -\alpha I_d \\ I_d & 0_d & 0_d \\ \hline I_d & 0_d & 0_d \end{array}\right] \tag{2.7}$$

One can check by similar analysis that (2.7) is equivalent to the update rule

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \,.$$

2.1 Proving algorithm convergence

Convergence analysis of convex optimization algorithms typically follows a two step procedure. First one must show that the algorithm has a fixed point that solves the optimization problem in question. Then, one must verify that from a reasonable starting point, the algorithm converges to this optimal solution at a specified rate.

In dynamical systems, such proofs are called stability analysis. By writing common first order methods as dynamical systems, we can unify their stability analysis. For a general problem with minimum occurring at $y_\star$, a necessary condition for optimality is that $u_\star = \nabla f(y_\star) = 0$. Substituting into (2.1), the fixed point satisfies

$$y_\star = C\xi_\star \quad \text{and} \quad \xi_\star = A\xi_\star$$

In particular, A must have an eigenvalue of 1. If the blocks of A are diagonal as in the Gradient, Heavy-ball, or Nesterov methods shown above, then the eigenvalue of 1 will have a geometric multiplicity of at least d.

Proving that all paths lead to the optimal solution requires more effort and constitutes the bulk of what is studied herein. Before we proceed for general convex f, it is instructive to study what happens for quadratic f.

2.2 Quadratic problems

Suppose f is a convex, quadratic function $f(y) = \frac{1}{2}y^\mathsf{T} Q y - p^\mathsf{T} y + r$, where $mI_d \preceq Q \preceq LI_d$ in the positive definite ordering. The gradient of f is simply $\nabla f(y) = Qy - p$ and the optimal solution is $y_\star = Q^{-1}p$.

What happens when we run a first order method on a quadratic problem? Assume throughout this section that $D = 0$. Substituting the equation for $y_\star$ and $\nabla f(y)$ back into (2.2), we obtain the system of equations:

$$\begin{aligned} \xi_{k+1} &= A\xi_k + Bu_k \\ y_k &= C\xi_k \\ u_k &= \nabla f(y_k) = Qy_k - p = Q(y_k - y_\star) \end{aligned}$$

Now make use of the fixed-point equations $y_\star = C\xi_\star$ and $\xi_\star = A\xi_\star$ and we obtain $u_k = QC(\xi_k - \xi_\star)$. Eliminating y_k and u_k from the above equations, we obtain

$$\xi_{k+1} - \xi_\star = (A + BQC)(\xi_k - \xi_\star) \tag{2.8}$$

Let $T := A + BQC$ denote the closed-loop state transition matrix. A necessary and sufficient condition for ξ_k to converge to $\xi_\star$ is that the *spectral radius* of T is strictly less than 1. Recall that the spectral radius of a matrix M is defined as the largest magnitude of the eigenvalues of M. We denote the spectral radius by $\rho(M)$. It is a fact that

$$\rho(M) \le \|M^k\|^{1/k} \qquad \text{for all } k \text{ and} \qquad \rho(M) = \lim_{k\to\infty} \|M^k\|^{1/k}$$

where $\|\cdot\|$ is the induced 2-norm. Therefore, for any $\varepsilon > 0$, we have for all k sufficiently large that $\rho(T)^k \le \|T^k\| \le (\rho(T) + \varepsilon)^k$. Hence, we can bound the convergence rate:

$$\|\xi_k - \xi_\star\| = \|T^k(\xi_0 - \xi_\star)\| \le \|T^k\|\|\xi_0 - \xi_\star\| \le (\rho(T) + \varepsilon)^k \|\xi_0 - \xi_\star\|\,.$$

So the spectral radius also determines the rate of convergence of the algorithm. With only bounds on the eigenvalues of Q, we can provide conditions under which the algorithms above converge for quadratic f.

Proposition 1 *The following table gives worst-case rates for different algorithms and parameter choices when applied to a class of* ***convex quadratic functions****. We assume here that $f : \mathbb{R}^d \to \mathbb{R}$ where $f(x) = \frac{1}{2}x^\mathsf{T} Qx - p^\mathsf{T} x + r$ and Q is any matrix that satisfies $mI_d \preceq Q \preceq LI_d$. We also define $\kappa := L/m$.*

Method	*Parameter choice*	*Rate bound*	*Comment*
Gradient	$\alpha = \frac{1}{L}$	$\rho = 1 - \frac{1}{\kappa}$	*popular choice*
Nesterov	$\alpha = \frac{1}{L}$, $\beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$	$\rho = \sqrt{1 - \frac{1}{\sqrt{\kappa}}}$	*standard choice*
Gradient	$\alpha = \frac{2}{L+m}$	$\rho = \frac{\kappa-1}{\kappa+1}$	*optimal tuning*
Nesterov	$\alpha = \frac{4}{3L+m}$, $\beta = \frac{\sqrt{3\kappa+1}-2}{\sqrt{3\kappa+1}+2}$	$\rho = 1 - \frac{2}{\sqrt{3\kappa+1}}$	*optimal tuning*
Heavy-ball	$\alpha = \frac{4}{(\sqrt{L}+\sqrt{m})^2}$, $\beta = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2$	$\rho = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$	*optimal tuning*

All of these results are proven by elementary linear algebra and the bounds are tight. In other words, there exists a quadratic function that achieves the worst-case ρ. See Appendix A for more detail.

Unfortunately, the proof technique used in Proposition 1 does not extend to the case where f is a more general strongly convex function. However, a different characterization of stability does generalize and will be described in Section 3. It turns out that for linear systems, stability is equivalent to the feasibility of a particular semidefinite program. We will see in the sequel that similar semidefinite programs can be used to certify stability of nonlinear systems.

Proposition 2 *Suppose $T \in \mathbb{R}^{d\times d}$. Then $\rho(T) < \rho$ if and only if there exists a $P \succ 0$ satisfying $T^\mathsf{T} PT - \rho^2 P \prec 0$.*

The proof of Proposition 2 is elementary so we omit it. The use of Linear Matrix Inequalities (LMI) to characterize stability of a linear time-invariant system dates back to Lyapunov [18], and we give a more detailed account of this history in Section 3.4. Now suppose we are studying a dynamical system of the form $\xi_{k+1} - \xi_\star = T(\xi_k - \xi_\star)$ as in (2.8). Then, if there exists a $P \succ 0$ satisfying $T^\mathsf{T} PT - \rho^2 P \prec 0$,

$$(\xi_{k+1} - \xi_\star)^\mathsf{T} P(\xi_{k+1} - \xi_\star) < \rho^2(\xi_k - \xi_\star)^\mathsf{T} P(\xi_k - \xi_\star) \tag{2.9}$$

along all trajectories. If $\rho < 1$, then the sequence $\{\xi_k\}_{k\ge 0}$ converges linearly to $\xi_\star$. Iterating (2.9) down to $k = 0$, we see that

$$(\xi_k - \xi_\star)^\mathsf{T} P(\xi_k - \xi_\star) < \rho^{2k}(\xi_0 - \xi_\star)^\mathsf{T} P(\xi_0 - \xi_\star) \tag{2.10}$$

which implies that

$$\|\xi_k - \xi_\star\| < \sqrt{\mathrm{cond}(P)}\,\rho^k \|\xi_0 - \xi_\star\| \tag{2.11}$$

where $\mathrm{cond}(P)$ is the condition number of P. In what follows, we will generalize this semidefinite programming approach to yield feasibility problems that are sufficient to characterize when the closed loop system (2.2) converges and which provide bounds on the distance to optimality as well. The function

$$V(\xi) = (\xi - \xi_\star)^\mathsf{T} P(\xi - \xi_\star) \tag{2.12}$$

is called a *Lyapunov function* for the dynamical system. This function strictly decreases over all trajectories and hence certifies that the algorithm is *stable*, i.e., converges to nominal values. The conventional method for proving stability of an electromechanical system is to show that some notion of *total energy* always decreases over time. Lyapunov functions provide a convenient mathematical formulation of this notion of total energy.

The question for the remainder of the paper is how can we search for Lyapunov-like functions that guarantee algorithmic convergence when f is not quadratic.

3 Proving convergence using integral quadratic constraints

When the function being minimized is quadratic as explored in Section 2.2, its gradient is affine and the interconnected dynamical system is a simple linear difference equation whose stability and convergence rate is analyzed solely in terms of eigenvalues of the closed-loop system. When the cost function is not quadratic, the gradient update is not an affine function and hence a different analysis technique is required.

A popular technique in the control theory literature is to use *integral quadratic constraints* (IQCs) to capture features of the behavior of partially-known components. The term IQC was introduced in the seminal paper by Megretski and Rantzer [19]. In that work, the authors analyzed continuous time dynamical systems and the constraints involved integrals of quadratic functions, hence the name IQC.

In the development that follows, we repurpose the classical IQC theory for use in algorithm analysis. This requires using discrete time dynamical systems so our constraints will involve sums of quadratics rather than integrals. We also adapt the theory in a way that allows us to certify a specific convergence rate in addition to stability.

3.1 An introduction to IQCs

IQCs provide a convenient framework for analyzing interconnected dynamical systems that contain components that are noisy, uncertain, or otherwise difficult to model. The idea is to replace this troublesome component by a quadratic constraint on its inputs and outputs that is known to be satisfied by all possible instances of the component. If we can certify that the newly constrained system performs as desired, then the original system must do so as well.

Suppose $\phi : \ell_{2\mathrm{e}} \to \ell_{2\mathrm{e}}$ is the troublesome function we wish to analyze. The equation $u = \phi(y)$ can be represented using a block diagram, as in Figure 1.

$$y \longrightarrow \boxed{\phi} \longrightarrow u$$

Figure 1: Block-diagram representation of the map ϕ.

Although we do not know ϕ exactly, we assume that we have some knowledge of the constraints it imposes on the pair (y, u). For example, suppose it is known that ϕ satisfies the following properties:

(i) ϕ is static and memoryless: $\phi(y_0, y_1, \dots) = (g(y_0), g(y_1), \dots)$ for some $g : \mathbb{R}^d \to \mathbb{R}^d$.

(ii) g is L-Lipschitz: $\|g(y_1) - g(y_2)\| \le L\|y_1 - y_2\|$ for all $y_1, y_2 \in \mathbb{R}^d$.

Now suppose that $y = (y_0, y_1, \dots)$ is an arbitrary sequence of vectors in $\mathbb{R}^d$, and $u = \phi(y)$ is the output of the unknown function applied to y. Property (ii) implies that $\|u_k - u_\star\| \le L\|y_k - y_\star\|$ for all k, where $(y_\star, u_\star)$ is any pair of vectors satisfying $u_\star = g(y_\star)$ that will serve as a reference point. In matrix form, this is

$$\begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix}^\mathsf{T} \begin{bmatrix} L^2 I_d & 0_d \\ 0_d & -I_d \end{bmatrix} \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix} \ge 0 \qquad \text{for } k = 0, 1, \dots \tag{3.1}$$

Core idea behind IQC. Instead of analyzing a system that contains ϕ, we analyze the system where ϕ is removed, but we enforce the constraints (3.1) on the signals (y, u). Since (3.1) is true for all admissible choices of ϕ, then any properties we can prove for the constrained system must hold for the original system as well.

Note that (3.1) is rather special in that the quadratic coupling of (y, u) is pointwise; it only manifests itself as separate quadratic constraints on each (y_k, u_k). It is possible to specify more general quadratic constraints that couple different k values, and the key insight above still holds. To do this, introduce auxiliary sequences $\zeta, z \in \ell_{2\mathrm{e}}$ together with a map Ψ characterized by the matrices $(A_\Psi, B_\Psi^y, B_\Psi^u, C_\Psi, D_\Psi^y, D_\Psi^u)$ and the recursion

$$\zeta_0 = \zeta_\star \tag{3.2a}$$
$$\zeta_{k+1} = A_\Psi \zeta_k + B_\Psi^y y_k + B_\Psi^u u_k \tag{3.2b}$$
$$z_k = C_\Psi \zeta_k + D_\Psi^y y_k + D_\Psi^u u_k \tag{3.2c}$$

where we will define the initial condition $\zeta_\star$ shortly. The equations (3.2) define an affine map $z = \Psi(y, u)$. Assuming a reference point $(y_\star, u_\star)$ as before, we can define the associated reference $(\zeta_\star, z_\star)$ that is a fixed point of (3.2). In other words,

$$\zeta_\star = A_\Psi \zeta_\star + B_\Psi^y y_\star + B_\Psi^u u_\star \tag{3.3a}$$
$$z_\star = C_\Psi \zeta_\star + D_\Psi^y y_\star + D_\Psi^u u_\star \tag{3.3b}$$

We will require that $\rho(A_\Psi) < 1$, which ensures that (3.3) has a unique solution $(\zeta_\star, z_\star)$ for any choice of $(y_\star, u_\star)$. Note that the reference points are defined in such a way that if we use $y = (y_\star, y_\star, \dots)$ and $u = (u_\star, u_\star, \dots)$ in (3.2), we will obtain $\zeta = (\zeta_\star, \zeta_\star, \dots)$ and $z = (z_\star, z_\star, \dots)$.

We then consider the quadratic forms $(z_k - z_\star)^\mathsf{T} M (z_k - z_\star)$ for a given symmetric matrix M (typically indefinite). Note that each such quadratic form is a function of $(y_0, \dots, y_k, u_0, \dots, u_k)$ that is determined by our choice of $(\Psi, M, y_\star, u_\star)$. In our previous example (3.1), Ψ has no dynamics and the corresponding Ψ and M are

$$\Psi = \left[\begin{array}{c|cc} A_\Psi & B_\Psi^y & B_\Psi^u \\ \hline C_\Psi & D_\Psi^y & D_\Psi^u \end{array}\right] = \left[\begin{array}{c|cc} 0_d & 0_d & 0_d \\ \hline 0_d & I_d & 0_d \\ 0_d & 0_d & I_d \end{array}\right] \qquad M = \begin{bmatrix} L^2 I_d & 0_d \\ 0_d & -I_d \end{bmatrix} \tag{3.4}$$

In other words, if we use the definitions (3.4), then $(z_k - z_\star)^\mathsf{T} M (z_k - z_\star) \ge 0$ is the same as (3.1). In general, these sorts of quadratic constraints are called IQCs. We consider four different types of IQCs, which we now define.

Definition 3 *Suppose $\phi : \ell_{2\mathrm{e}}^d \to \ell_{2\mathrm{e}}^d$ is an unknown map and $\Psi : \ell_{2\mathrm{e}}^d \times \ell_{2\mathrm{e}}^d \to \ell_{2\mathrm{e}}^m$ is a given linear map of the form (3.2) with $\rho(A_\Psi) < 1$. Suppose $(y_\star, u_\star) \in \mathbb{R}^{2d}$ is a given reference point and let $(\zeta_\star, z_\star)$ be the unique solution of (3.3). Suppose $y \in \ell_{2\mathrm{e}}^d$ is an arbitrary sequence. Let $u = \phi(y)$ and let $z = \Psi(y, u)$ as in (3.2). We say ϕ satisfies the*

1. ***Pointwise IQC*** *defined by* $(\Psi, M, y_\star, u_\star)$ *if for all* $y \in \ell_{2e}^d$ *and* $k \geq 0$*,*

$$(z_k - z_\star)^\mathsf{T} M (z_k - z_\star) \geq 0$$

2. ***Hard IQC*** *defined by* $(\Psi, M, y_\star, u_\star)$ *if for all* $y \in \ell_{2e}^d$ *and* $k \geq 0$*,*

$$\sum_{t=0}^{k} (z_t - z_\star)^\mathsf{T} M (z_t - z_\star) \geq 0$$

3. *ρ-**Hard IQC*** *defined by* $(\Psi, M, \rho, y_\star, u_\star)$ *if for all* $y \in \ell_{2e}^d$ *and* $k \geq 0$*,*

$$\sum_{t=0}^{k} \rho^{-2t} (z_t - z_\star)^\mathsf{T} M (z_t - z_\star) \geq 0$$

4. ***Soft IQC*** *defined by* $(\Psi, M, y_\star, u_\star)$ *if for all* $y - y_\star \in \ell_2^d$*,*

$$\sum_{t=0}^{\infty} (z_t - z_\star)^\mathsf{T} M (z_t - z_\star) \geq 0 \qquad \textit{(and the sum is convergent)}$$

Note that the example (3.1) is a pointwise IQC. Examples of the other types of IQCs will be described in Section 3.3. Note that the sets of maps satisfying the various IQCs defined above are nested as follows:

$$\{\text{all pointwise IQCs}\} \subset \{\text{all } \rho\text{-hard IQCs, } \rho < 1\} \subset \{\text{all hard IQCs}\} \subset \{\text{all soft IQCs}\}$$

For example, if ϕ satisfies a pointwise IQC defined by $(\Psi, M, y_\star, u_\star)$ then it must also satisfy the hard IQC defined by the same $(\Psi, M, y_\star, u_\star)$. The notions of *hard IQC* and the more general *soft IQC* (sometimes simply called *IQC*) were introduced in [19] and their relationship is discussed in [38]. These concepts are useful in proving that a dynamic system is stable, but do not directly allow for the derivation of useful bounds on convergence rates. The definitions of *pointwise* and *ρ-hard* IQCs are new, and were created for the purpose of better characterizing convergence rates, as we will see in Section 3.2.

Finally, note that $y_\star$ and $u_\star$ are nominal inputs and outputs for the unknown ϕ, and they can be tuned to certify different fixed points of the interconnected system. We will see in Section 3.2 that certifying a particular convergence rate to some fixed point does not require prior knowledge of fixed point; only knowledge that the fixed point exists.

3.2 Stability and performance results

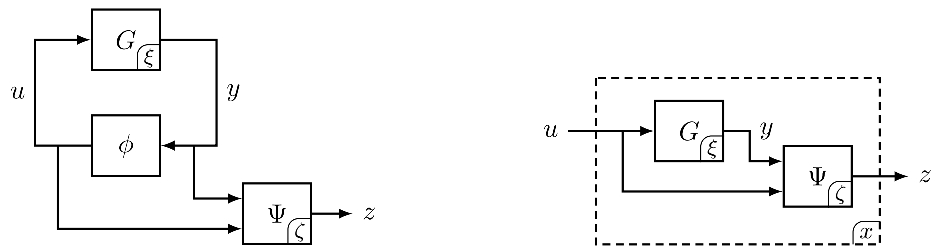
In this section, we show how IQCs can be used to prove that iterative algorithms converge and to bound the rate of convergence. In both cases, the certification requires solving a tractable convex program. We note that the original work on IQCs [19] only proved stability (boundedness). Some other works have addressed exponential stability [12, 34, 35], but the emphasis of these works is on proving the *existence* of an exponential decay rate, and so the rates constructed are very conservative. We require rates that are less conservative, and this is reflected in the inclusion of ρ in the LMI of our main result, Theorem 4.

We will now combine the dynamical system framework of Section 2 and the IQC theory of Section 3.1. Suppose $G : \ell_{2e}^d \to \ell_{2e}^d$ is an affine map $u \mapsto y$ described by the recursion

$$\xi_{k+1} = A\xi_k + Bu_k \tag{3.5a}$$

$$y_k = C\xi_k \tag{3.5b}$$

where (A, B, C) are matrices of appropriate dimensions. The map is affine rather than linear because of the initial condition ξ_0. As in Section 2, G is the iterative algorithm we wish to analyze, and using the general formalism of Section 3.1, ϕ is the nonlinear map $(y_0, y_1, \dots) \mapsto (u_0, u_1, \dots)$ that characterizes the feedback. Of course, this framework subsumes the special case of interest in which $u_k = \nabla f(y_k)$ for each k. We assume that ϕ satisfies an IQC, and this IQC is characterized by a map Ψ and matrix M. We can interpret $z = \Psi(y, u)$ as a filtered version of the signals u and y. These equations can be represented using a block-diagram as in Figure 2a.



(a) The auxiliary system Ψ produces z, which is a filtered version of the signals y and u.

(b) The nonlinearity ϕ is replaced by a constraint on z, so we may remove ϕ entirely.

Figure 2: Feedback interconnection between a system G and a nonlinearity ϕ. An IQC is a constraint on (y, u) satisfied by ϕ. We only analyze the constrained system and so we may remove the ϕ block entirely.

Consider the dynamics of G and Ψ from (3.5) and (3.2), respectively. Upon eliminating y, the recursions may be combined to obtain

$$\begin{bmatrix} \xi_{k+1} \\ \zeta_{k+1} \end{bmatrix} = \begin{bmatrix} A & 0 \\ B_\Psi^y C & A_\Psi \end{bmatrix} \begin{bmatrix} \xi_k \\ \zeta_k \end{bmatrix} + \begin{bmatrix} B \\ B_\Psi^u \end{bmatrix} u_k \tag{3.6a}$$

$$z_k = \begin{bmatrix} D_\Psi^y C & C_\Psi \end{bmatrix} \begin{bmatrix} \xi_k \\ \zeta_k \end{bmatrix} + D_\Psi^u u_k \tag{3.6b}$$

More succinctly, (3.6) can be written as

$$\begin{aligned} x_{k+1} &= \hat{A} x_k + \hat{B} u_k \\ z_k &= \hat{C} x_k + \hat{D} u_k \end{aligned} \qquad \text{where we defined } x_k := \begin{bmatrix} \xi_k \\ \zeta_k \end{bmatrix} \tag{3.7}$$

The dynamical system (3.7) is represented in Figure 2b by the dashed box. Our main result is as follows.

Theorem 4 (Main result) *Consider the block interconnection of Figure 2a. Suppose G is given by* (3.5) *and Ψ is given by* (3.2). *Define $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ as in* (3.6)–(3.7). *Suppose $(\xi_\star, \zeta_\star, y_\star, u_\star, z_\star)$ is a fixed point of* (3.5) *and* (3.2). *In other words,*

$$\xi_\star = A \xi_\star + B u_\star \tag{3.8a}$$

$$y_\star = C \xi_\star \tag{3.8b}$$

$$\zeta_\star = A_\Psi \zeta_\star + B_\Psi^y y_\star + B_\Psi^u u_\star \tag{3.8c}$$

$$z_\star = C_\Psi \zeta_\star + D_\Psi^y y_\star + D_\Psi^u u_\star \tag{3.8d}$$

*Suppose ϕ satisfies the ρ-**hard IQC** defined by $(\Psi, M, \rho, y_\star, u_\star)$ where $0 \le \rho \le 1$. Consider the following LMI.*

$$\begin{bmatrix} \hat{A}^\mathsf{T} P \hat{A} - \rho^2 P & \hat{A}^\mathsf{T} P \hat{B} \\ \hat{B}^\mathsf{T} P \hat{A} & \hat{B}^\mathsf{T} P \hat{B} \end{bmatrix} + \lambda \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix}^\mathsf{T} M \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} \preceq 0 \tag{3.9}$$

If (3.9) *is feasible for some* $P \succ 0$ *and* $\lambda \geq 0$*, then for any* ξ_0*, we have*

$$\|\xi_k - \xi_\star\| \leq \sqrt{\operatorname{cond}(P)}\, \rho^k \, \|\xi_0 - \xi_\star\| \quad \textit{for all } k$$

where $\operatorname{cond}(P)$ *is the condition number of* P*.*

Proof. Let $x, u, z \in \ell_{2\mathrm{e}}$ be a set of sequences that satisfies (3.7). Suppose (P, λ) is a solution of (3.9). Multiply (3.9) on the left and right by $\begin{bmatrix}(x_k - x_\star)^\mathsf{T} & (u_k - u_\star)^\mathsf{T}\end{bmatrix}$ and its transpose, respectively. Making use of (3.7)–(3.8), we obtain

$$(x_{k+1} - x_\star)^\mathsf{T} P (x_{k+1} - x_\star) - \rho^2\, (x_k - x_\star)^\mathsf{T} P (x_k - x_\star) + \lambda\, (z_k - z_\star)^\mathsf{T} M (z_k - z_\star) \leq 0 \quad (3.10)$$

Multiply (3.10) by ρ^{-2k} for each k and sum over k. The first two terms yield a telescoping sum and we obtain

$$\rho^{-2k+2}(x_k - x_\star)^\mathsf{T} P (x_k - x_\star) - \rho^2 (x_0 - x_\star)^\mathsf{T} P (x_0 - x_\star) \\ + \lambda \sum_{t=0}^{k-1} \rho^{-2t} (z_t - z_\star)^\mathsf{T} M (z_t - z_\star) \leq 0$$

Because ϕ satisfies the ρ-hard IQC defined by $(\Psi, M, \rho, y_\star, u_\star)$, the summation part of the inequality is nonnegative for all k. Therefore,

$$(x_k - x_\star)^\mathsf{T} P (x_k - x_\star) \leq \rho^{2k} (x_0 - x_\star)^\mathsf{T} P (x_0 - x_\star)$$

for all k and consequently $\|x_k - x_\star\| \leq \sqrt{\operatorname{cond}(P)}\, \rho^k \, \|x_0 - x_\star\|$. Recall from (3.7) that $x_k = (\xi_k, \zeta_k)$ and from (3.2a) that $\zeta_0 = \zeta_\star$. Therefore,

$$\begin{aligned}
\|\xi_k - \xi_\star\|^2 &\leq \|x_k - x_\star\|^2 \\
&\leq \operatorname{cond}(P) \rho^{2k} \|x_0 - x_\star\|^2 \\
&= \operatorname{cond}(P) \rho^{2k} \left(\|\xi_0 - \xi_\star\|^2 + \|\zeta_0 - \zeta_\star\|^2 \right) \\
&= \operatorname{cond}(P) \rho^{2k} \|\xi_0 - \xi_\star\|^2
\end{aligned}$$

and this completes the proof. ■

We now make several comments regarding Theorem 4.

Pointwise and hard IQCs Theorem 4 can easily be adapted to other types of IQCs.

1. If the pointwise IQC defined by some $(\Psi, M, y_\star, u_\star)$ is satisfied, then so is the ρ-hard IQC defined by $(\Psi, M, \rho, y_\star, u_\star)$ for any ρ. Therefore, we may apply Theorem 4 directly and ignore the ρ-hardness constraint. The smallest ρ that makes (3.9) feasible will correspond to the best exponential rate we can guarantee.

2. Hard IQCs are a special case of ρ-hard IQCs with $\rho = 1$. Therefore, if the LMI (3.9) is feasible, Theorem 4 guarantees that $\|\xi_k - \xi_\star\| \leq \sqrt{\operatorname{cond}(P)} \|\xi_0 - \xi_\star\|$. In other words, the iterates are bounded (but not necessarily convergent).

3. If a ρ_1-hard IQC is satisfied, then so is the ρ-hard IQC for any $\rho \geq \rho_1$. Also, if (3.9) is feasible for some ρ_2, it will also be feasible for any $\rho \geq \rho_2$. Therefore, if we use a ρ_1-hard IQC and (3.9) is feasible for ρ_2, then the smallest exponential rate we can guarantee is $\rho = \max(\rho_1, \rho_2)$.

Multiple IQCs Theorem 4 can also be generalized to the case where ϕ satisfies multiple IQCs. Suppose ϕ satisfies the ρ-hard IQCs defined by $(\Psi_i, M_i, \rho, y_\star^{(i)}, u_\star^{(i)})$ for $i = 1, \ldots, r$. Simply redefine the matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ in a manner analogous to (3.7), but where the output is now $(z_k^{(1)}, \ldots, z_k^{(r)})$. Instead of (3.9), use

$$\begin{bmatrix} \hat{A}^\mathsf{T} P \hat{A} - \rho^2 P & \hat{A}^\mathsf{T} P \hat{B} \\ \hat{B}^\mathsf{T} P \hat{A} & \hat{B}^\mathsf{T} P \hat{B} \end{bmatrix} + \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix}^\mathsf{T} \begin{bmatrix} \lambda_1 M_1 & & \\ & \ddots & \\ & & \lambda_r M_r \end{bmatrix} \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} \preceq 0 \tag{3.11}$$

where $\lambda_1, \ldots, \lambda_r \geq 0$. Thus, when (3.11) is multiplied out as in (3.10), we now obtain

$$(x_{k+1} - x_\star)^\mathsf{T} P (x_{k+1} - x_\star) - \rho^2 \, (x_k - x_\star)^\mathsf{T} P (x_k - x_\star) \\ + \sum_{i=1}^r \lambda_i \, (z_k^{(i)} - z_\star^{(i)})^\mathsf{T} M_i (z_k^{(i)} - z_\star^{(i)}) \leq 0$$

and the rest of the proof proceeds as in Theorem 4.

Remark on Lyapunov functions In the quadratic case treated in Section 2.2, a quadratic Lyapunov function is constructed from the solution P in (2.12). In the case of IQCs, such a quadratic function cannot serve as a Lyapunov function because it does not strictly decrease over all trajectories. Nevertheless, Theorem 4 shows how ρ-hard IQCs can be used to certify a convergence rate and no Lyapunov function is explicitly constructed. We can explain this difference more explicitly. If $V(x)$ is a Lyapunov function, then by definition it satisfies the properties;

(i) $\lambda_1 \|x - x_\star\|^2 \leq V(x) \leq \lambda_2 \|x - x_\star\|^2$ for all x and k.

(ii) $V(x_{k+1}) \leq \rho^2 V(x_k)$ for all system trajectories $\{x_k\}_{k \geq 0}$.

Property (ii) implies that

$$V(x_k) \leq \rho^{2k} V(x_0) \tag{3.12}$$

which, combined with Property (i) implies that $\|x_k - x_\star\| \leq \sqrt{\lambda_2/\lambda_1}\, \rho^k \|x_0 - x_\star\|$. In Theorem 4, we use $V(x) = (x - x_\star)^\mathsf{T} P (x - x_\star)$, which satisfies (i) but not (ii). So $V(x)$ is *not* a Lyapunov function in the technical sense. Nevertheless, we prove directly that (3.12) holds, and so the desired result still holds. That is, $V(x)$ serves the same purpose as a Lyapunov function.

3.3 IQCs for convex functions

We will derive three IQCs that are useful for describing gradients of strongly convex functions: the *sector* (pointwise) IQC, the *off-by-one* (hard) IQC, and *weighted off-by-one* (ρ-hard) IQC. In general, gradients of strongly convex functions satisfy an infinite family of IQCs, originally characterized by Zames and Falb for the single-input-single-output case [47]. A generalization of the Zames-Falb IQCs to multidimensional functions is derived in [11]. Both the sector and off-by-one IQCs are special cases of Zames-Falb, while the weighted off-by-one IQC is a convex combination of the sector and off-by-one IQCs. While the Zames-Falb family is infinite, the three simple IQCs mentioned above are the only ones used in this paper. IQCs can be used to describe many other types of functions as well, and further examples are available in [19]. We begin with some fundamental inequalities that describe strongly convex function.

Proposition 5 (basic properties) *Suppose $f \in S(m,L)$. Then the following properties hold for all $x, y \in \mathbb{R}^d$.*

$$f(y) \le f(x) + \nabla f(x)^\mathsf{T}(y-x) + \frac{L}{2}\|y-x\|^2 \tag{3.13a}$$

$$(\nabla f(y) - \nabla f(x))^\mathsf{T}(y-x) \ge \frac{1}{L}\|\nabla f(y) - \nabla f(x)\|^2 \tag{3.13b}$$

$$f(y) \ge f(x) + \nabla f(x)^\mathsf{T}(y-x) + \frac{1}{2L}\|\nabla f(y) - \nabla f(x)\|^2 \tag{3.13c}$$

$$\begin{bmatrix} y-x \\ \nabla f(y) - \nabla f(x) \end{bmatrix}^\mathsf{T} \begin{bmatrix} -2mLI_d & (L+m)I_d \\ (L+m)I_d & -2I_d \end{bmatrix} \begin{bmatrix} y-x \\ \nabla f(y) - \nabla f(x) \end{bmatrix} \ge 0 \tag{3.13d}$$

Proof. Property (3.13a) follows from the definition of Lipschitz gradients. Properties (3.13b) and (3.13c) are commonly known as *co-coercivity.* To prove (3.13d), define $g(x) := f(x) - \frac{m}{2}\|x\|^2$ and note that $g \in S(0, L-m)$. Applying (3.13b) to g and rearranging, we obtain

$$(L+m)(\nabla f(y) - \nabla f(x))^\mathsf{T}(y-x) \ge mL\|y-x\|^2 + \|\nabla f(y) - \nabla f(x)\|^2$$

which is precisely (3.13d). Detailed derivations of these properties can be found for example in [23]. ■

Lemma 6 (sector IQC) *Suppose $f_k \in S(m,L)$ for each k, and $(y_\star, u_\star)$ is a common reference point for the gradients of f_k. In other words, $u_\star = \nabla f_k(y_\star)$ for all $k \ge 0$. Let $\phi := (\nabla f_0, \nabla f_1, \dots)$. If $u = \phi(y)$, then ϕ satisfies the **pointwise IQC** defined by*

$$\Psi = \begin{bmatrix} LI_d & -I_d \\ -mI_d & I_d \end{bmatrix} \qquad \text{and} \qquad M = \begin{bmatrix} 0_d & I_d \\ I_d & 0_d \end{bmatrix}$$

The corresponding quadratic inequality is that for all $y \in \ell_2^d$ and $k \ge 0$, we have

$$\begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix}^\mathsf{T} \begin{bmatrix} -2mLI_d & (L+m)I_d \\ (L+m)I_d & -2I_d \end{bmatrix} \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix} \ge 0 \tag{3.14}$$

Proof. Equation (3.14) follows immediately from (3.13d) by using $(f, x, y) \to (f_k, y_\star, y_k)$. It can be verified that

$$\Psi^\mathsf{T} M \Psi = \begin{bmatrix} -2mLI_d & (L+m)I_d \\ (L+m)I_d & -2I_d \end{bmatrix} \qquad \text{and} \qquad z_k - z_\star = \Psi \begin{bmatrix} y_k - y_\star \\ u_k - u_\star \end{bmatrix}$$

and therefore (3.14) is equivalent to $(z_k - z_\star)^\mathsf{T} M (z_k - z_\star) \ge 0$ as required. ■

Remark 7 *In Lemma 6, we use a slight abuse of notation in representing the map $\Psi : \ell_{2e}^d \times \ell_{2e}^d \to \ell_{2e}^m$. In writing Ψ as a matrix in $\mathbb{R}^{2d \times 2d}$, we mean that Ψ is a static map that operates pointwise on (y, u). In other words,*

$$z_k = \Psi \begin{bmatrix} y_k \\ u_k \end{bmatrix} \qquad \textit{for all } k.$$

Lemma 8 (off-by-one IQC) *Suppose $f \in S(m,L)$ and $(y_\star, u_\star)$ is a point satisfying $u_\star = \nabla f(y_\star) = 0$. Let $\phi := (\nabla f, \nabla f, \dots)$. Then ϕ satisfies the **hard IQC** defined by*

$$\Psi = \left[\begin{array}{c|cc} 0_d & -LI_d & I_d \\ \hline I_d & LI_d & -I_d \\ 0_d & -mI_d & I_d \end{array}\right] \qquad \text{and} \qquad M = \begin{bmatrix} 0_d & I_d \\ I_d & 0_d \end{bmatrix}$$

The corresponding quadratic inequality is that for all $y \in \ell_2^d$ and $k \geq 0$, we have

$$(\tilde{u}_0 - m\tilde{y}_0)^\mathsf{T}(L\tilde{y}_0 - \tilde{u}_0) + \sum_{t=1}^{k}(\tilde{u}_t - m\tilde{y}_t)^\mathsf{T}\big(L(\tilde{y}_t - \tilde{y}_{t-1}) - (\tilde{u}_t - \tilde{u}_{t-1})\big) \geq 0 \tag{3.15}$$

where we have defined $\tilde{y}_k := y_k - y_\star$ and $\tilde{u}_k := u_k - u_\star$.

Proof. Define the function

$$g(x) := f(x) - f(y_\star) - \frac{m}{2}\|x - y_\star\|^2$$

It is straightforward to check that $g \in S(0, L-m)$, and $g(x) \geq g(y_\star) = 0$ for all $x \in \mathbb{R}^d$. Applying (3.13c) using $(f, x, y) \to (g, y_\star, y_k)$, we observe that

$$q_k := (L-m)g(y_k) - \frac{1}{2}\|\nabla g(y_k)\|^2 \geq 0 \qquad \text{for all } k \geq 0 \tag{3.16}$$

Moreover, $\nabla g(y_k) = \nabla f(y_k) - m(y_k - y_\star) = \tilde{u}_k - m\tilde{y}_k$. Therefore, we may manipulate the first term in (3.15) to eliminate $\tilde{u}_0$ and obtain

$$\begin{aligned}
(\tilde{u}_0 - m\tilde{y}_0)^\mathsf{T}(L\tilde{y}_0 - \tilde{u}_0) &= \nabla g(y_0)^\mathsf{T}((L-m)\tilde{y}_0 - \nabla g(y_0)) \\
&= (L-m)\nabla g(y_0)^\mathsf{T}\tilde{y}_0 - \|\nabla g(y_0)\|^2 \\
&\geq (L-m)g(y_0) - \tfrac{1}{2}\|\nabla g(y_0)\|^2 \\
&= q_0
\end{aligned} \tag{3.17}$$

where the inequality follows from applying (3.13c) using $(f, x, y) \to (g, y_0, y_\star)$. Similarly, the t^{th} term in the sum in (3.15) can be bounded by eliminating $\tilde{u}_t$ and $\tilde{u}_{t-1}$.

$$\begin{aligned}
&(\tilde{u}_t - m\tilde{y}_t)^\mathsf{T}(L(\tilde{y}_t - \tilde{y}_{t-1}) - (\tilde{u}_t - \tilde{u}_{t-1})) \\
&\qquad = (L-m)\nabla g(y_t)^\mathsf{T}(\tilde{y}_t - \tilde{y}_{t-1}) - \nabla g(y_t)^\mathsf{T}(\nabla g(y_t) - \nabla g(y_{t-1})) \\
&\qquad \geq (L-m)(g(y_t) - g(y_{t-1})) - \tfrac{1}{2}\|\nabla g(y_t)\|^2 + \tfrac{1}{2}\|\nabla g(y_{t-1})\|^2 \\
&\qquad = q_t - q_{t-1}
\end{aligned} \tag{3.18}$$

where the inequality follows this time from applying (3.13c) using $(f, x, y) \to (g, y_t, y_{t-1})$. Substituting (3.17) and (3.18) into the left-hand side of (3.15), the sum telescopes and we obtain the lower bound q_k, which is nonnegative from (3.16).

To verify the IQC factorization, note that the state equations for Ψ given in the statement of Lemma 8 are

$$\left\{\begin{aligned} \zeta_0 &= \zeta_\star \\ \zeta_{k+1} &= -Ly_k + u_k \\ z_k &= \begin{bmatrix} \zeta_k + Ly_k - u_k \\ -my_k + u_k \end{bmatrix} \end{aligned}\right\} \implies \left\{\begin{aligned} z_0 &= \begin{bmatrix} \zeta_\star + Ly_0 - u_0 \\ -my_0 + u_0 \end{bmatrix} \\ z_k &= \begin{bmatrix} L(y_k - y_{k-1}) - (u_k - u_{k-1}) \\ -my_k + u_k \end{bmatrix}, \quad k \geq 1 \end{aligned}\right\}$$

Moreover, the solution to the fixed-point equations (3.3) are

$$\zeta_\star = -Ly_\star + u_\star \qquad \text{and} \qquad z_\star = \begin{bmatrix} 0 \\ -my_\star + u_\star \end{bmatrix}$$

Therefore, we conclude that

$$z_0 - z_\star = \begin{bmatrix} L\tilde{y}_0 - \tilde{u}_0 \\ -m\tilde{y}_0 + \tilde{u}_0 \end{bmatrix} \quad \text{and} \quad z_k - z_\star = \begin{bmatrix} L(\tilde{y}_k - \tilde{y}_{k-1}) - (\tilde{u}_k - \tilde{u}_{k-1}) \\ -m\tilde{y}_k + \tilde{u}_k \end{bmatrix}, \ k \geq 1$$

and it follows that $\sum_{t=0}^{k}(z_t - z_\star)^\mathsf{T} M (z_t - z_\star) \geq 0$ is equivalent to (3.15), as required. ■

Note that the sector IQC (3.14) is a special case of the off-by-one IQC when $k = 0$. The off-by-one IQC is itself a special case of the Zames-Falb IQC, which we now describe.

Lemma 9 (Zames-Falb IQC) *Suppose $f \in S(m, L)$ and $(y_\star, u_\star)$ is a point satisfying $u_\star = \nabla f(y_\star) = 0$. Let $\phi := (\nabla f, \nabla f, \dots)$ and let $h_1, h_2, \dots$ be any sequence of real numbers that satisfies*

(i) $\{h_\tau\}_{\tau \geq 1}$ *is finitely nonzero, and h_s is the last nonzero component.*

(ii) $0 \leq h_\tau \leq 1$ *for all $\tau \geq 1$.*

(iii) $\sum_{\tau=1}^{\infty} h_\tau \leq 1$.

Then ϕ satisfies the ***hard IQC*** *defined by*

$$\Psi = \left[\begin{array}{cccc|cc} 0_d & 0_d & \dots & 0_d & -LI_d & I_d \\ I_d & 0_d & \dots & 0_d & 0_d & 0_d \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0_d & \dots & I_d & 0_d & 0_d & 0_d \\ \hline h_1 I_d & h_2 I_d & \dots & h_s I_d & LI_d & -I_d \\ 0_d & 0_d & \dots & 0_d & -mI_d & I_d \end{array}\right] \qquad \text{and} \qquad M = \begin{bmatrix} 0_d & I_d \\ I_d & 0_d \end{bmatrix}$$

The corresponding quadratic inequality is that for all $y \in \ell_2^d$ and $k \geq 0$, we have

$$\sum_{t=0}^{k} (\tilde{u}_t - m\tilde{y}_t)^\mathsf{T} \left(L\left(\tilde{y}_t - \sum_{\tau=1}^{t} h_\tau \tilde{y}_{t-\tau}\right) - \left(\tilde{u}_t - \sum_{\tau=1}^{t} h_\tau \tilde{u}_{t-\tau}\right)\right) \geq 0 \tag{3.19}$$

where we have defined $\tilde{y}_k := y_k - y_\star$ and $\tilde{u}_k := u_k - u_\star$.

Proof. We will construct a proof for a general sequence $h_1, h_2, \dots$ by first considering a specific set of sequences. Fix some $j \geq 1$ and consider the case where

$$h_\tau = \begin{cases} 1 & \tau = j \\ 0 & \tau \neq j \end{cases}$$

For $t < j$, the terms in the sum (3.19) have the form

$$(\tilde{u}_t - m\tilde{y}_t)^\mathsf{T}(L\tilde{y}_t - \tilde{u}_t)$$

which are bounded below by $q_t \geq 0$, as proven in Lemma 8, (3.16)–(3.17). For $t \geq j$, the terms in the sum (3.19) have the form

$$(\tilde{u}_t - m\tilde{y}_t)^\mathsf{T}(L(\tilde{y}_t - \tilde{y}_{t-j}) - (\tilde{u}_t - \tilde{u}_{t-j}))$$

which are bounded below by $q_t - q_{t-j}$, as proven in (3.18). Summing up (3.19) for all t yields a telescoping sum, thereby proving that (3.19) holds. This can be thought of an "off-by-j" IQC. Indeed, when $j = 1$, we recover the off-by-one IQC of Lemma 8.

Now note that if we take a convex combination of the inequalities (3.19) corresponding to each off-by-j IQC and let the associated coefficient be h_j, we have proven (3.19) for the case of a general sequence $h_1, h_2, \dots$. ∎

Though we will not make use of the more general Zames-Falb family of inequalities, we include them as they are interesting in their own right and may find applications in future work. We conclude this section with a ρ-hard version of the off-by-one IQC. This final IQC will be critical for deriving convergence rates.

Lemma 10 (weighted off-by-one IQC) *Suppose $f \in S(m, L)$ and $(y_\star, u_\star)$ is a point satisfying $u_\star = \nabla f(y_\star) = 0$. Let $\phi := (\nabla f, \nabla f, \dots)$. Then for any $(\bar\rho, \rho)$ satisfying $0 \le \bar\rho \le \rho \le 1$, ϕ satisfies the ρ-**hard IQC** defined by*

$$\Psi = \left[\begin{array}{c|cc} 0_d & -LI_d & I_d \\ \hline \bar\rho^2 I_d & LI_d & -I_d \\ 0_d & -mI_d & I_d \end{array}\right] \qquad \text{and} \qquad M = \begin{bmatrix} 0_d & I_d \\ I_d & 0_d \end{bmatrix}$$

The corresponding quadratic inequality is that for all $y \in \ell_2^d$ and $k \ge 0$, we have

$$(\tilde u_0 - m\tilde y_0)^\mathsf{T}(L\tilde y_0 - \tilde u_0) + \sum_{t=1}^{k} \rho^{-2t}(\tilde u_t - m\tilde y_t)^\mathsf{T}\big(L(\tilde y_t - \bar\rho^2 \tilde y_{t-1}) - (\tilde u_t - \bar\rho^2 \tilde u_{t-1})\big) \ge 0 \quad (3.20)$$

where we have defined $\tilde y_k := y_k - y_\star$ and $\tilde u_k := u_k - u_\star$.

Proof. Note that the weighted off-by-one IQC is a Zames-Falb IQC with $h = (\bar\rho^2, 0, \dots)$. Thus the hardness and the factorization (Ψ, M) follows from Lemma 9. In order to prove ρ-hardness (3.20), a bit more work is required. First, observe (see remarks on pointwise and hard IQCs after Theorem 4) that it suffices to show $\bar\rho$-hardness, and this will imply ρ-hardness. The t^{th} term in the sum in (3.20) can be bounded as follows. First, define the general terms in the sector (Lemma 6) and off-by-one (Lemma 8) inequalities:

$$\begin{aligned} s_t &:= (\tilde u_t - m\tilde y_t)^\mathsf{T}(L\tilde y_t - \tilde u_t) \\ p_t &:= (\tilde u_t - m\tilde y_t)^\mathsf{T}\big(L(\tilde y_t - \tilde y_{t-1}) - (\tilde u_t - \tilde u_{t-1})\big) \end{aligned}$$

Algebraic manipulations reveal that the general term in the sum (3.20) satisfies

$$\begin{aligned} (\tilde u_t - m\tilde y_t)^\mathsf{T}\big(L(\tilde y_t - \bar\rho^2\tilde y_{t-1}) - (\tilde u_t - \bar\rho^2 \tilde u_{t-1})\big) &= (1-\bar\rho^2)s_t + \bar\rho^2 p_t \\ &\ge (1-\bar\rho^2)q_t + \bar\rho^2(q_t - q_{t-1}) \\ &= q_t - \bar\rho^2 q_{t-1} \end{aligned}$$

where the inequalities follow from (3.17) and (3.18). Substituting the general term back into (3.20) with $\rho = \bar\rho$, the $\bar\rho^{-2t}$ coefficient causes the sum to telescope and we are left with $\bar\rho^{-2k}q_k$, which is nonnegative from (3.16). This completes the proof. ∎

Remark 11 *In implementing the weighted off-by-one IQC, one can simply set $\bar\rho = \rho$. However, a less conservative approach is to keep $\bar\rho$ as an additional degree of freedom. In Theorem 4, the IQC constraint is included in (3.9) in the final term and is multiplied by the constant $\lambda \ge 0$. When using the weighted off-by-one IQC, this amounts to:*

$$\lambda\left((1-\bar\rho^2)s_t + \bar\rho^2 p_t\right) \qquad \text{with the constraints: } 0 \le \bar\rho \le \rho \text{ and } \lambda \ge 0$$

By defining $\lambda_1 = \lambda(1-\bar\rho^2)$ and $\lambda_2 = \lambda\bar\rho^2$, an equivalent expression is

$$\lambda_1 s_t + \lambda_2 p_t \qquad \text{with the constraints: } \lambda_1, \lambda_2 \ge 0 \text{ and } \lambda_2 \le \rho^2(\lambda_1 + \lambda_2)$$

3.4 Historical context of IQCs and Lyapunov theory

Constructing Lyapunov functions has a long history in control and dynamical systems, and the central focus of this paper is borrowing tools from this literature to see how we can generalize our analysis from quadratic functions to more general, nonlinear convex functions.

One of the most fundamental problems in control theory is certifying the stability of nonlinear systems. In interconnected systems such as electric circuits or chemical plants, individual components are typically modeled using differential (or difference) equations. Interconnected systems often contain nonlinearities or components that are otherwise difficult to model. The earliest results on such systems date back to the work of Lur'e and Postnikov [17]. The goal was to prove stability under a wide range of admissible uncertainties. This notion of robust stability was called *absolute stability*. Indeed, Lur'e studied precisely the model we are concerned with: a known linear system interconnected in feedback to an uncertain nonlinear system.

In the 1960's and 70's, several sufficient conditions for absolute stability were expressed as frequency-domain conditions. In other words, the main objects of interest are ratios of the Laplace transforms of the outputs to the inputs, also known as *transfer functions*. Examples include the Popov criterion [33], the small-gain theorem, the circle criterion, and passivity theory [46]. Frequency-domain conditions were popular at the time because they could be verified graphically. The work of Willems [42] unified many of the existing results by casting them in the time domain in a framework called *dissipativity theory*. This notion is on one hand a generalization of Lyapunov functions to include systems with exogenous inputs, and on the other hand a generalization of passivity theory and the small-gain theorem. These ideas form the core of modern nonlinear control theory, and are covered in many textbooks such as Khalil [14].

With the advent of computers, graphical methods were no longer required. The connection between frequency-domain conditions and Linear Matrix Inequalities (LMIs) was made by Kalman [13] and Yakubovich [43] and culminated in the Kalman-Yakubovich-Popov (KYP) lemma, also known as the Positive-Real lemma. This paved the way for the use of modern computational tools such as semidefinite programming. Another important development is the concept of the *structured singular value* [6], also known as μ-analysis. While previous theory had been used to describe *static* nonlinearities or uncertainties, μ-analysis is a computationally tractable framework for describing a system containing multiple *dynamic* uncertainties. A survey of μ-related techniques and results is given in [28]. For a comprehensive overview of the history and development of LMIs in control theory, we refer the reader to [4].

Integral Quadratic Constraints (IQCs) were first introduced by Yakubovich, who considered the notion of imposing quadratic constraints on an infinite-horizon control problem [45], and combining multiple constraints via the S-procedure [44]. The definitive work on IQCs is Megretski and Rantzer [19]. In this seminal paper, the authors showed that dissipativity theory, as well as all the frequency-domain conditions, could be formulated as IQCs. Furthermore, the KYP lemma in conjunction with the S-procedure allows stability to be verified by solving an LMI.

The seminal paper on IQCs [19] develops the theory primarily in the frequency domain, but also alludes to time-domain versions of the results by introducing *hard* IQCs. This notion of hard IQCs is pursued in [38], where the main IQC stability theorem is rederived entirely in the time domain. In the time domain, these constraints parallel the development of Nesterov, where we are able to construct inequalities linking multiple inputs and outputs of uncertain functions. This allows us to provide a wholly self-contained development of the theory. Moreover, we are able to enhance the techniques of [38], providing new IQCs and considerably sharper rates of convergence than those discussed in the earlier work. In this sense, our work provides useful methods for control theorists interested in estimating rates of stabilization of their control systems.

4 Case studies

We now use the results of Section 3 to rederive some existing results from the literature on iterative large-scale algorithms. The IQC approach gives a unified method to analyze many different algorithms. In addition to verifying existing results, we also present a negative result that was not previously known.

4.1 Computational approach

Given an iterative algorithm, our first step is to express it as a feedback interconnection of a discrete linear time-invariant dynamical system with a nonlinearity representing ∇f. This procedure is explained in Section 2 and yields matrices (A, B, C).

The next step is to decide which IQCs will be used to characterize the nonlinearity. A simple but conservative choice is the sector IQC defined in Lemma 6. A less conservative choice is the weighted off-by-one IQC of Lemma 10. For the chosen (Ψ, M), we find the smallest ρ such that the semidefinite program (SDP) (3.9) of Theorem 4 is feasible. In the case of the sector IQC, the SDP has variables (P, λ, ρ). For the weighted off-by-one IQC, the SDP has variables $(P, \lambda_1, \lambda_2, \rho)$ as explained in Remark 11. The resulting ρ is an upper bound for the worst-case convergence rate of the algorithm. Specifically,

$$\|\xi_k - \xi_\star\| \leq \sqrt{\operatorname{cond}(P)}\, \rho^k \|\xi_0 - \xi_\star\|.$$

To solve the SDP (3.9) numerically, observe that it is a quasiconvex program. In particular, for every fixed ρ, (3.9) is an LMI. The simplest way to solve (3.9) is to use a bisection search on ρ. For a fixed ρ, the SDP (3.9) or (3.11) become an LMI and can be efficiently solved using interior-point methods. Popular implementations include SDPT3, SeDuMi, and Mosek. This approach was used for all the simulations presented herein.

More sophisticated methods exist to solve (3.9) as well. A quasiconvex program of the type (3.9) is known as a *generalized eigenvalue optimization problem* (GEVP) [4]. The GEVP is well-studied and modified interior-point methods such as the *method of centers* [3] and the *long-step method of analytic centers* [21] can be used to solve it.

4.2 Lossless dimensionality reduction

The size of the SDP in (3.9) is proportional to d, the size of the state ξ_k in the optimization algorithm. This can be problematic in cases where d is large because it can be computationally costly to solve large SDPs. In many cases of interest, however, the algorithms we wish to analyze have a block-diagonal structure. For example, Nesterov's accelerated method has the form (2.5), which is

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} (1+\beta)I_d & -\beta I_d & -\alpha I_d \\ I_d & 0_d & 0_d \\ \hline (1+\beta)I_d & -\beta I_d & 0_d \end{array}\right] \tag{4.1}$$

Each of the matrices (A, B, C) is a block matrix with repeated diagonal blocks. Using Kronecker product notation (see Section 1.1, this means for example that

$$A = \begin{bmatrix} 1+\beta & -\beta \\ 1 & 0 \end{bmatrix} \otimes I_d$$

and similarly for B and C. Moreover, the IQCs we use to describe ∇f have the same sort of structure. That is, $(A_\Psi, B_\Psi^y, B_\Psi^u, C_\Psi, D_\Psi^y, D_\Psi^u)$ are block matrices with repeated diagonal blocks. Now consider the SDP (3.9) from Theorem 4.

$$\begin{bmatrix} \hat{A}^\mathsf{T} P \hat{A} - \rho^2 P & \hat{A}^\mathsf{T} P \hat{B} \\ \hat{B}^\mathsf{T} P \hat{A} & \hat{B}^\mathsf{T} P \hat{B} \end{bmatrix} + \lambda \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix}^\mathsf{T} M \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} \preceq 0 \tag{4.2}$$

Based on the discussion above, each of the matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{D}, M)$ have the form e.g. $A_0 \otimes I_d$. Rather than looking for a general $P \in \mathbb{R}^{nd \times nd}$ with $P \succ 0_{nd}$, if we restrict our search to $P = P_0 \otimes I_d$ with $P_0 \in \mathbb{R}^{n \times n}$ and $P_0 \succ 0_n$, then the SDP reduces to

$$\begin{bmatrix} \hat{A}_0^\mathsf{T} P_0 \hat{A}_0 - \rho^2 P_0 & \hat{A}_0^\mathsf{T} P_0 \hat{B}_0 \\ \hat{B}_0^\mathsf{T} P_0 \hat{A}_0 & \hat{B}_0^\mathsf{T} P_0 \hat{B}_0 \end{bmatrix} + \lambda \begin{bmatrix} \hat{C}_0 & \hat{D}_0 \end{bmatrix}^\mathsf{T} M_0 \begin{bmatrix} \hat{C}_0 & \hat{D}_0 \end{bmatrix} \preceq 0 \tag{4.3}$$

The resulting SDP no longer depends on d and is effectively the same as if we had solved the original problem with $d = 1$. As it turns out, there is no loss of generality in assuming a P of this form. To see why this is so, first suppose $P_0 \succ 0$ satisfies (4.3). Then clearly $P = P_0 \otimes I_d$ satisfies (4.2). Conversely, suppose $P \succ 0$ satisfies (4.2). Then define the matrix $P_0 := (I_n \otimes e_1)^\mathsf{T} P (I_n \otimes e_1)$ where $e_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^\mathsf{T} \in \mathbb{R}^{d \times 1}$. Note that P_0 is an $n \times n$ principal submatrix of P, and therefore $P_0 \succ 0$ because $P \succ 0$. Multiplying the left-hand side of (4.2) by $(I_n \otimes e_1)^\mathsf{T}$ on the left and $(I_n \otimes e_1)$ on the right, we conclude that P_0 satisfies (4.3). Thus, $\hat{P} = P_0 \otimes I_d$ is also a solution to (4.2). In other words, (4.2) is feasible if and only if (4.3) is feasible.

4.3 Known bounds for first-order optimization algorithms

The following proposition summarizes some of the known bounds for optimizing strongly convex functions.

Proposition 12 *The following table gives worst-case rate bounds for different algorithms and parameter choices when applied to a class of* ***strongly convex functions****. We assume here that $f : \mathbb{R}^d \to \mathbb{R}$ where $f \in S(m, L)$. Again, we define $\kappa := L/m$.*

Method	*Parameter choice*	*Rate bound*	*Comment*
Gradient	$\alpha = \frac{1}{L}$	$\rho \le \sqrt{\frac{\kappa-1}{\kappa+1}}$	*popular choice*
Nesterov	$\alpha = \frac{1}{L}$, $\beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$	$\rho \le \sqrt{1 - \frac{1}{\sqrt{\kappa}}}$	*standard choice*
Gradient	$\alpha = \frac{2}{L+m}$	$\rho = \frac{\kappa-1}{\kappa+1}$	*optimal tuning*

The Gradient bounds in the table above follow from the bound $\rho \le \sqrt{1 - \frac{2\alpha m L}{L+m}}$, which is proven in [23]. A tighter Gradient bound $\rho \le \max\{|1 - \alpha m|, |1 - \alpha L|\}$ is proven in [32] but makes the additional assumption that f is twice differentiable. The Nesterov bound in Proposition 12 is proven in [23] using the technique of estimate sequences. There are no known global convergence guarantees for the Heavy-ball method in the case of strongly convex functions, but it is proven in [32] that the Heavy-ball method converges *locally* with the same rate as in Proposition 1.

In the following sections, we will use IQC machinery to demonstrate that the first two bounds in Proposition 12 are loose. We will construct tighter bounds for the strongly convex case without requiring additional assumptions about locality or twice-differentiability. We will then use our framework to help guide a refutation of the convergence of the Heavy-ball method.

4.4 The Gradient method

The Gradient method with constant stepsize is among the simplest optimization schemes. The recursion is given by

$$\xi_{k+1} = \xi_k - \alpha \nabla f(\xi_k) \tag{4.4}$$

We will analyze this algorithm by applying Theorem 4. Since $f \in S(m, L)$, we may use the sector IQC of Lemma 6 and (3.9) together with the dimensionality reduction of Section 4.2 yields the following SDP.

$$\begin{bmatrix} (1-\rho^2)P & -\alpha P \\ -\alpha P & \alpha^2 P \end{bmatrix} + \lambda \begin{bmatrix} -2mL & (L+m) \\ (L+m) & -2 \end{bmatrix} \preceq 0, \qquad P \succ 0, \qquad \lambda \geq 0 \tag{4.5}$$

Note that P is 1×1, so we may set $P = 1$ without loss of generality and we obtain the following LMI in (ρ^2, λ).

$$\begin{bmatrix} 1-\rho^2 & -\alpha \\ -\alpha & \alpha^2 \end{bmatrix} + \lambda \begin{bmatrix} -2mL & L+m \\ L+m & -2 \end{bmatrix} \preceq 0 \qquad \text{and} \qquad \lambda \geq 0 \tag{4.6}$$

Using Schur complements, (4.6) is equivalent to

$$\lambda \geq \frac{\alpha^2}{2} \qquad \text{and} \qquad \rho^2 \geq 1 - 2mL\lambda - \frac{(\alpha - (L+m)\lambda)^2}{2\lambda - \alpha^2} \tag{4.7}$$

By analyzing the lower bound on ρ in (4.7), we can find the optimal choice of λ as a function of the stepsize α. Omitting the details, we eventually obtain the simple expression $\rho = \max\{|1-\alpha m|, |1-\alpha L|\}$. This is precisely the bound found for the quadratic case, as derived in Appendix A. However, we have shown something much stronger here, since the only assumption we made about f is that ∇f satisfies the sector IQC of Lemma 6. In particular, the Gradient method rates in Proposition 1 hold not only for quadratics, but also for strongly convex functions, and even for functions that change or switch over time (either stochastically, adversarially, or otherwise), so long as each function satisfies the pointwise sector constraint. Note that (4.6) can be transformed using Schur complements:

$$\begin{bmatrix} -2mL\lambda - \rho^2 & (L+m)\lambda & 1 \\ (L+m)\lambda & -2\lambda & -\alpha \\ 1 & -\alpha & -1 \end{bmatrix} \preceq 0 \tag{4.8}$$

And now (4.8) is linear in $(\rho^2, \lambda, \alpha)$. This formulation allows one to directly answer questions such as "what range of stepsizes can yield a given rate?".

4.5 Nesterov's accelerated method

Nesterov's accelerated method with constant stepsize converges at a linear rate. There exists some $c > 0$ such that for any initial condition ξ_0,

$$\|\xi_k - \xi_\star\| \leq c\rho^k \|\xi_0 - \xi_\star\| \qquad \text{with} \qquad \rho = \sqrt{1 - \sqrt{\tfrac{m}{L}}}$$

when applied to functions $f \in S(m, L)$. In this case, the parameters are the standard parameters from Proposition 12, which are $\alpha := 1/L$ and $\beta := (\sqrt{L} - \sqrt{m})/(\sqrt{L} + \sqrt{m})$ [23]. Nesterov also showed that a *lower bound* on convergence rate for *any* algorithm of the form (2.2) and for any $f \in S(m, L)$ is given by

$$\|\xi_k - \xi_\star\| \geq \rho_{\text{opt}}^k \|\xi_0 - \xi_\star\| \qquad \text{with} \qquad \rho_{\text{opt}} = \frac{\sqrt{L} - \sqrt{m}}{\sqrt{L} + \sqrt{m}}. \tag{4.9}$$

Since ρ and ρ_{opt} behave similarly as $L/m \to \infty$, Nesterov's accelerated method is sometimes called "optimal" or "nearly optimal".

We computed the rate bounds using Theorem 4 using either the sector IQC of Lemma 6, or a combination of the sector IQC and the weighted off-by-one IQC of Lemma 10. It is important to note that unlike the Gradient method case, the LMI (3.9) is no longer linear in ρ^2. Therefore, we found the minimal ρ by performing a bisection search on ρ, see the first plot in Figure 3.

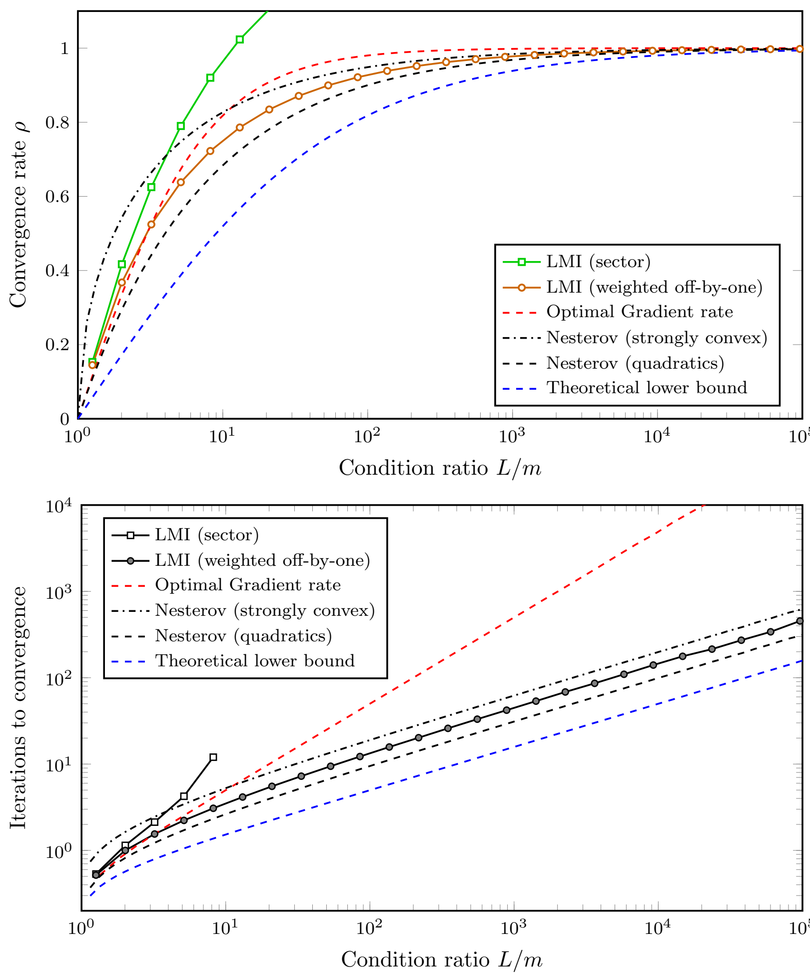


Figure 3: Upper bounds for Nesterov's accelerated method applied to $f \in S(m, L)$ using the standard tuning in Proposition 12. We tested both the sector IQC and the weighted off-by-one IQC. The first plot shows convergence rate and the second plot shows number of iterations required to achieve convergence to a specified tolerance. The *theoretical lower bound* ρ_{opt} is given in (4.9). The rate that can be certified using the LMI approach is strictly better than the rate proved in [23] using estimate sequences.

The rate obtained using the sector IQC alone is very poor. To understand why, recall from Lemma 6 that the sector IQC allows for f_k to be different at each iteration. Unlike the Gradient method, Nesterov's accelerated method is not robust to having a changing f_k. However, convergence can nevertheless be guaranteed as long as $\rho < 1$, which corresponds approximately to $L/m < 11.7$.

The rate obtained using the weighted off-by-one IQC improves upon the rate proven in [23] using the estimate sequence approach (see Proposition 12). Note that we do not have an analytical expression for the improved bound; it was found numerically by solving the LMI of Theorem 4.

Given that $\|x_k\| \leq \sqrt{\operatorname{cond}(P)}\rho^k\|x_0\|$, if we seek the smallest k such that $\|x_k\| \leq \varepsilon$, then it suffices that $\sqrt{\operatorname{cond}(P)}\rho^k\|x_0\| \leq \varepsilon$. This implies that

$$k \geq \left(-\frac{1}{2\log\rho}\right)\log\left(\frac{\operatorname{cond}(P)\|x_0\|^2}{\varepsilon^2}\right) \tag{4.10}$$

For the second plot in Figure 3, we plotted $-1/\log\rho$ versus L/m to get a sense of how the relative iteration count scales as a function of condition number. As we can see from Figure 3, Nesterov's method applied to quadratics is within a factor of 2 of the theoretical lower bound, and the bound we can prove for Nesterov's method applied to strongly convex functions is within a factor of 1.4 of the bound for quadratics.

Finally, we must also ensure that P is reasonably well-conditioned. In Figure 4, we see that $\operatorname{cond}(P)$ appears to be proportional to L/m, which agrees with the scale factor found by Nesterov [23].

If we repeat the above experiments, but instead using the optimal tuning of Nesterov's method given in Proposition 1, the resulting plots are virtually identical. The only differences are that the curves are shifted down slightly because the optimal rate for quadratics is now $1 - \frac{2}{\sqrt{3\kappa+1}}$ instead of $1 - \frac{1}{\sqrt{\kappa}}$. The sector-IQC curve goes unstable a little sooner as well, at around $L/m \approx 10$. Roughly speaking, if we use the optimal tuning we can guarantee slightly faster convergence but slightly less robustness.

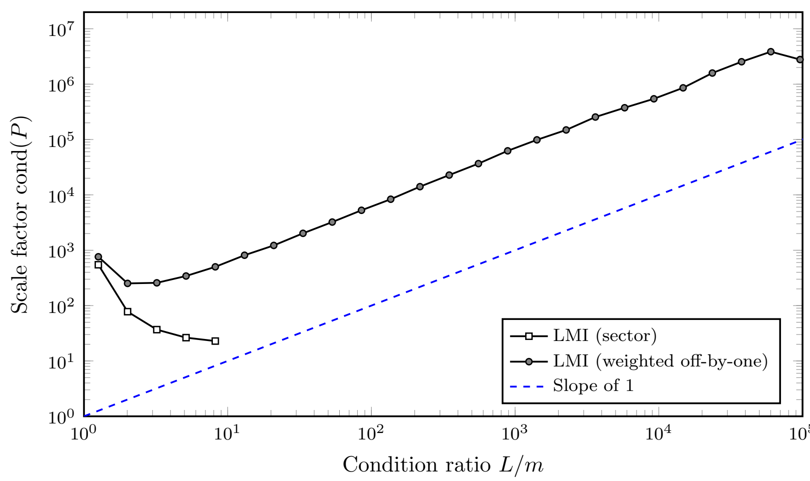


Figure 4: Condition number $\operatorname{cond}(P)$ when using the weighted off-by-one IQC. It is within a constant factor of L/m. Note that $\log\operatorname{cond}(P)$ appears in (4.10) for computing minimum iterations to convergence.

4.6 The Heavy-ball method

The optimal Heavy-ball rate for quadratics in Proposition 1 matches Nesterov's lower bound (4.9) for strongly convex functions. Although the Heavy-ball method and Nesterov's accelerated method have similar recursions, Figures 3 and 5 tell very different stories. When we allow for a different f_k at every iteration (sector IQC), we can guarantee stability when $L/m \approx 6$ or less. When we include the weighted off-by-one IQC as well, we can only guarantee stability when $L/m \approx 18$ or less. While it seems possible that using more IQCs could potentially improve this upper bound, it turns out that the poor quality of these bounds is due to something more serious: **the Heavy-ball method optimized for quadratics does not converge for general** $f \in S(m, L)$.

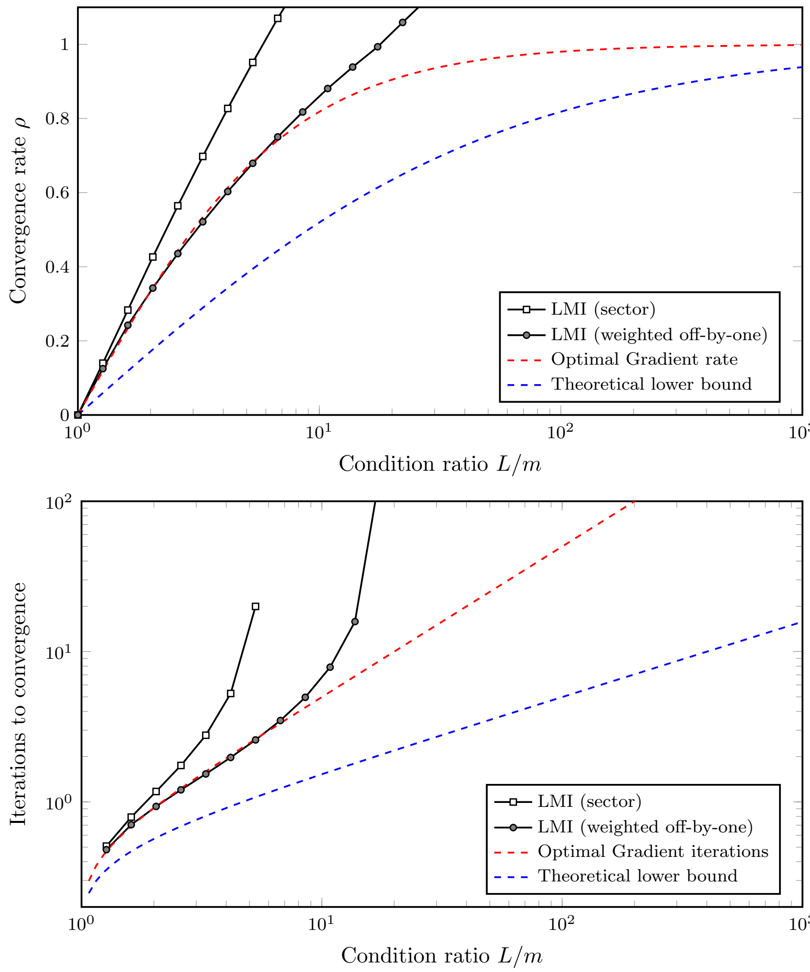


Figure 5: Upper bounds for the Heavy-ball method, using either the sector IQC or the weighted off-by-one IQC. Convergence rate (first plot) and number of iterations required to achieve convergence to a specified tolerance (second plot). Note that the theoretical lower bound is equal to the optimal Heavy-ball rate for quadratics. The *theoretical lower bound* ρ_{opt} is given in (4.9).

To find an example of an $f(x)$ that leads to a non-convergent Heavy-ball method, Figure 5 indicates that we should search for $L/m > 18$. The following one-dimensional example does the job.

$$\nabla f(x) = \begin{cases} 25x & x < 1 \\ x + 24 & 1 \le x < 2 \\ 25x - 24 & x \ge 2 \end{cases} \tag{4.11}$$

It is easy to check that $\nabla f(x)$ is continuous and monotone, and so $f \in S(m, L)$ with $m = 1$ and $L = 25$. When using an initial condition in the interval $3.07 \le x_0 \le 3.46$, the Heavy-ball method produces a limit cycle with oscillations that never damp out. The first 50 iterates for $x_0 = 3.3$ are shown in Figure 6, and a plot of $f(x)$ with the limit cycle overlaid is shown in Figure 7.

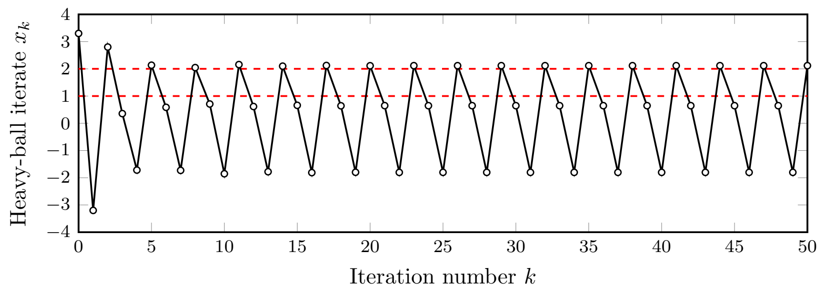


Figure 6: Iteration history of the Heavy-ball method when optimizing $f(x)$ defined in (4.11). Dashed lines separate the pieces of $f(x)$. The iterates tend to a limit cycle, so the Heavy-ball method does not converge for this particular strongly convex function.

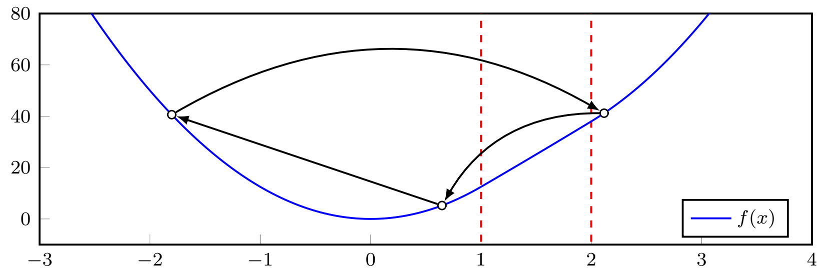


Figure 7: Graph of $f(x)$ defined in (4.11) with the limit cycle overlaid on top.

For a detailed proof that f can indeed converge to a limit cycle, see Appendix B. We further investigate the stability of the Heavy-ball method in Section 5.

5 Further applications

5.1 Stability of the Heavy-ball method

We saw in Section 4.6 that the Heavy-ball method that uses α and β optimized for quadratic functions is unstable for general strongly convex functions. A natural question to ask is whether the Heavy-ball method is stable over the class $S(m, L)$ for *some* choice of α and β. This experiment is easy to carry out in our framework, because choosing new values of α and β simply amounts to changing parameters in the LMI. We chose $\alpha = \frac{1}{L}$, and for a sampling of points in $\beta \in [0, 1]$, we evaluated the corresponding Heavy-ball method using Theorem 4 together with the weighted off-by-one IQC. See Figure 8.

The first plot shows convergence rate. When $\beta = 0$, the Heavy-ball method becomes the Gradient method, which is always convergent. However, we can improve upon the gradient rate by optimizing over β. The best achievable rate is given by the black curve. The black curve lies strictly above the optimal Heavy-ball rate for quadratics, but below the optimal gradient rate.

In the second plot, we show the iterations required to achieve convergence. Again, the black curve represents the optimal parameter choice. As L/m gets large, the envelope veers away from the optimal Heavy-ball curve and becomes parallel to the optimal gradient

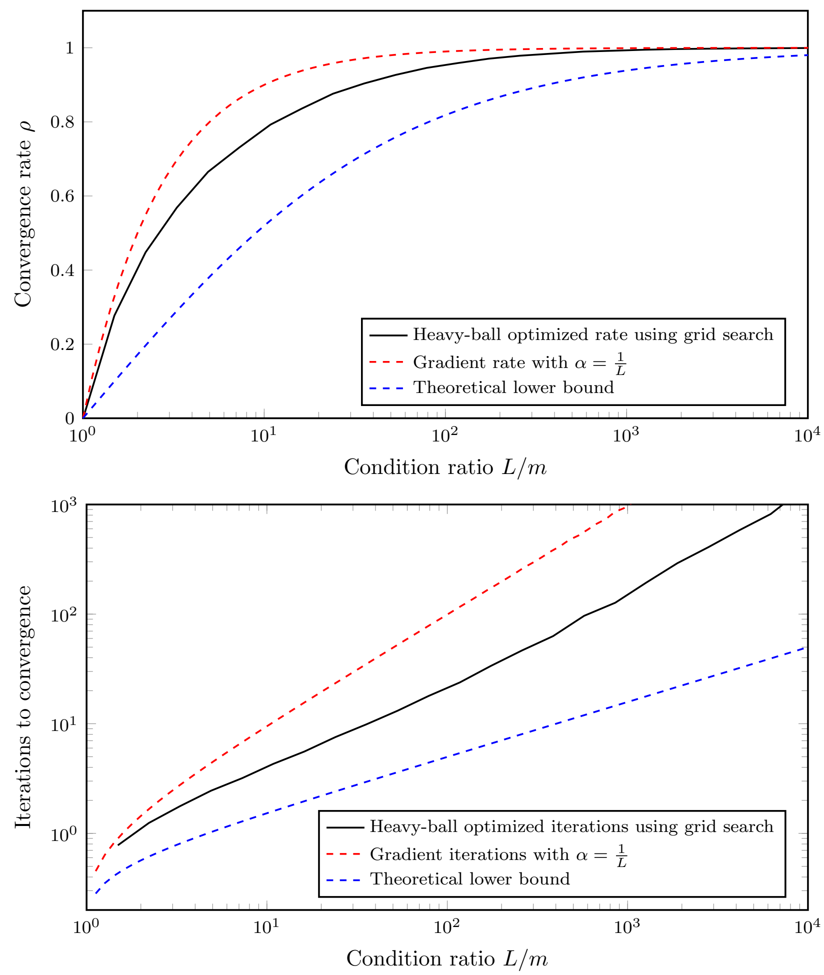


Figure 8: Upper bounds for the Heavy-ball method. We fixed $\alpha = \frac{1}{L}$, and for each L/m, we picked β that led to the optimal rate. The result is the solid black curve. We plotted convergence rate (first plot) and number of iterations required to achieve convergence to a specified tolerance (second plot). The *theoretical lower bound* ρ_{opt} is given in (4.9) and is the same as the optimal Heavy-ball rate for quadratics.

curve. So when L/m is large, even when β is chosen optimally, the Heavy-ball method is comparable to the Gradient method in worst-case for general strongly convex functions.

5.2 Multiplicative gradient noise

A common consideration is the inclusion of noise in the gradient computation. One possible model is *relative deterministic noise* where we assume the gradient error is proportional to the distance to optimality [32]. Instead of directly observing $\nabla f(y)$, we see $u_k = \nabla f(y_k) + r_k$, where

$$\|r_k\| \leq \delta \|\nabla f(y_k)\|$$

for some small nonnegative δ. The IQC framework can be used to analyze such situations to study the robustness of various algorithms to this type of noise.

If w_k is the true gradient, we actually measure $u_k = \Delta_k w_k$, where the gradient error is bounded above by a quantity proportional to the true gradient. In other words, we assume there is some $\delta > 0$ such that $\|u_k - w_k\| \le \delta \|w_k\|$. Squaring both sides of the inequality and rearranging, we obtain the IQC

$$\begin{bmatrix} w_k \\ u_k \end{bmatrix}^\mathsf{T} \begin{bmatrix} \delta^2 - 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} w_k \\ u_k \end{bmatrix} \ge 0 \qquad \text{for all } k$$

Note that this is simply the sector IQC with $m = 1 - \delta$ and $L = 1 + \delta$. We make no assumptions on how the noise is generated; it may be the output of a stochastic process, or could even be chosen adversarially. The modified block-diagram is shown in Figure 9.

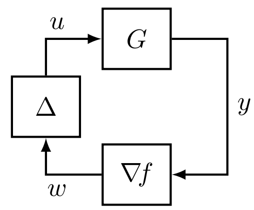


Figure 9: Block-diagram representation of the standard interconnection with an additional block Δ representing multiplicative noise.

By making a small modification, we can apply Theorem 4. We will look to show that the following inequality holds over all trajectories

$$x_{k+1}^\mathsf{T} P x_{k+1} - \rho^2 \, x_k^\mathsf{T} P x_k + \lambda_1 z_k^\mathsf{T} M z_k + \lambda_2 \begin{bmatrix} w_k \\ u_k \end{bmatrix}^\mathsf{T} \begin{bmatrix} \delta^2 - 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} w_k \\ u_k \end{bmatrix} \le 0 \tag{5.1}$$

for some $\lambda_1, \lambda_2 \ge 0$. In order to formulate an LMI that implies a solution to (5.1), we use the signal $\begin{bmatrix} x_k^\mathsf{T} & u_k^\mathsf{T} & w_k^\mathsf{T} \end{bmatrix}$. Consequently, the matrices $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ from (3.6)–(3.7) now become a map $(w_k, u_k) \mapsto z_k$. This leads to an LMI of the form (3.11) which is now block-3×3 instead of the 2×2 LMI of Theorem 4. The proof is identical to that of Theorem 4.

Gradient method Our first experiment is to test the Gradient method. We used noise values of $\delta \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$. See Figure 10.

In examining Figure 10, we observe that the Gradient method with stepsize $\frac{2}{L+m}$ is not very robust to multiplicative noise. Even with noise as low as 1% ($\delta = 0.01$), the Gradient method is no longer stable for $L/m > 100$. An explanation for this phenomenon is that in choosing the stepsize α, we are trading off convergence rate with robustness. The choice $\frac{2}{L+m}$ yields the minimum worst-case rate, but is fragile to noise. If we pick a more conservative stepsize such as the popular choice $\alpha = \frac{1}{L}$, we obtain a very different picture. See Figure 11.

Notice that with the updated stepsize of $\alpha = \frac{1}{L}$, the Gradient method is now robust to multiplicative noise. Robustness comes at the expense of a degradation in the best achievable convergence rate. This degradation manifests itself as a gap in Figure 11 between the black curves and the other ones.

Nesterov's accelerated method We can carry out an experiment similar to the one we did with the Gradient method, but now with Nesterov's method. As before, we examine the trade-off between the magnitude of the multiplicative noise and the degradation of the optimal convergence rate. This time, we use $\delta \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$. See Figure 12.

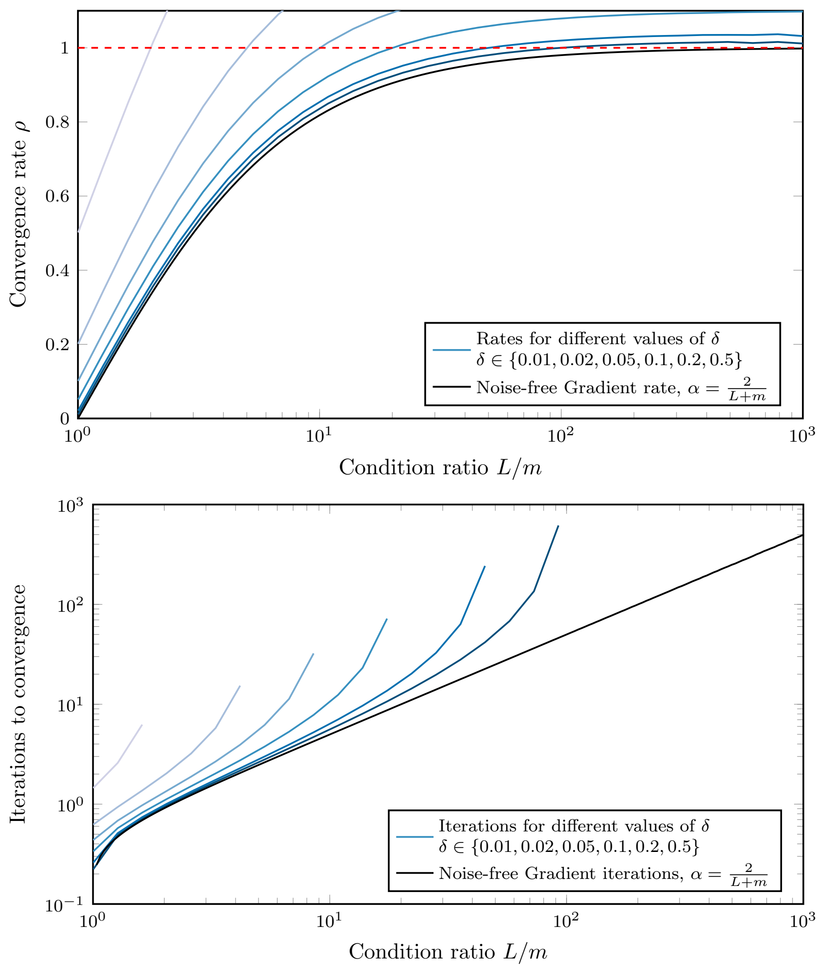


Figure 10: Convergence rate and iterations to convergence for the Gradient method with $\alpha = \frac{2}{L+m}$, for various noise parameters δ. This method is not robust to noise.

As with our first Gradient method test, Nesterov's method is not robust to multiplicative noise. For moderate L/m, the degradation is minor, but eventually leads to instability when we reach a certain threshold. The idea that accelerated methods are sensitive to noise and can lead to an accumulation of error was noted in the recent work [5], using a different notion of gradient perturbation.

Robustness of Nesterov's method can be improved by modifiying the α and β parameters. Choosing a smaller α pushes back the instability threshold, while choosing a smaller β simultaneously pushes back the instability threshold and degrades the rate. In the limit $\beta \to 0$, Nesterov's method becomes the Gradient method, so we recover the plots of Figure 11.

5.3 Proximal point methods

Suppose we are interested in solving a problem of the form

$$\begin{array}{ll} \text{minimize} & f(x) + P(x) \\ \text{subject to} & x \in \mathbb{R}^n \end{array}$$

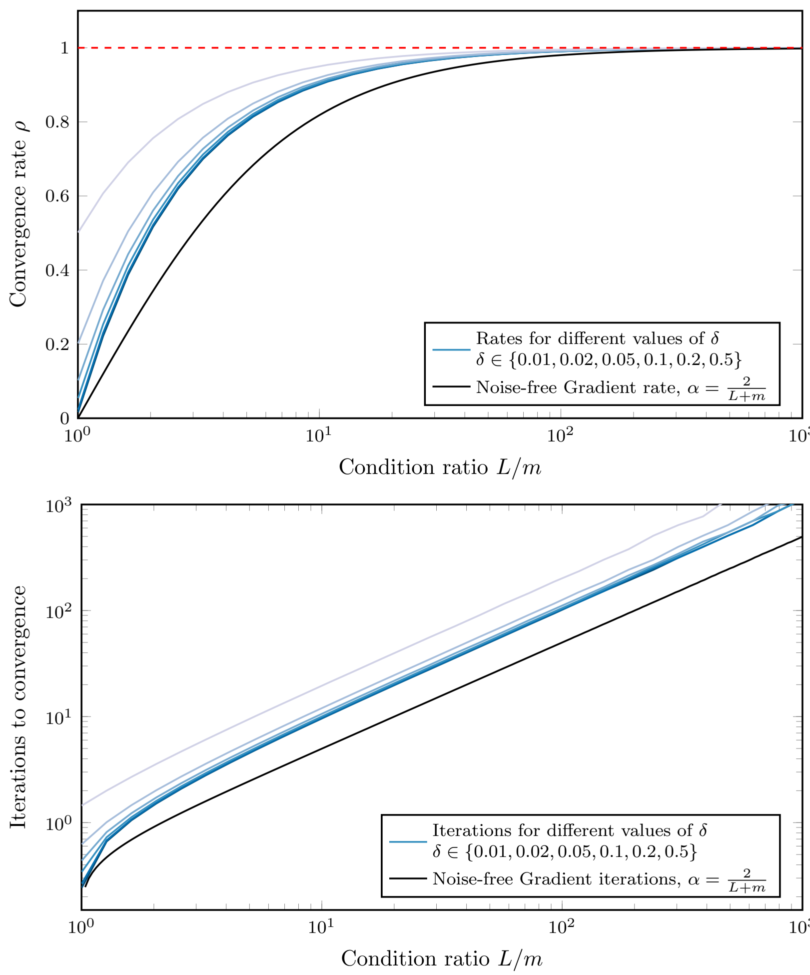


Figure 11: Convergence rate and iterations to convergence for the Gradient method with $\alpha = \frac{1}{L}$, for various noise parameters δ. This method is robust to noise, but at the expense of a gap in performance compared to the optimal stepsize of $\alpha = \frac{2}{L+m}$.

where $f \in S(m, L)$, and P is an extended-real-valued convex function on $\mathbb{R}^n$. An example of such a problem is constrained optimization, where we require that $x \in C$. In this case, we simply let P be the indicator function of C. We will now show how the IQC framework can be used to analyze algorithms involving a *proximal* operator. Define the proximal operator of P as

$$\Pi_\nu(x) := \arg\min_y \left(\tfrac{1}{2}\|x - y\|^2 + \nu P(y)\right)$$

As an illustrative example, we will show how to analyze the proximal version of Nesterov's algorithm. Iterations take the form:

$$\begin{aligned} \xi_{k+1} &= \Pi_\nu\left(y_k - \alpha \nabla f(y_k)\right) \\ y_k &= \xi_k + \beta(\xi_k - \xi_{k-1}) \end{aligned} \tag{5.2}$$

Note that when $\Pi_\nu = I$, we recover the standard Nesterov algorithm. When $\beta = 0$, we recover the proximal gradient method.

In order to analyze this algorithm, we must characterize Π_ν using IQCs. To this end, let $T := \partial P$ be the subdifferential of P. Then, $\Pi_\nu(x)$ is the unique point such that

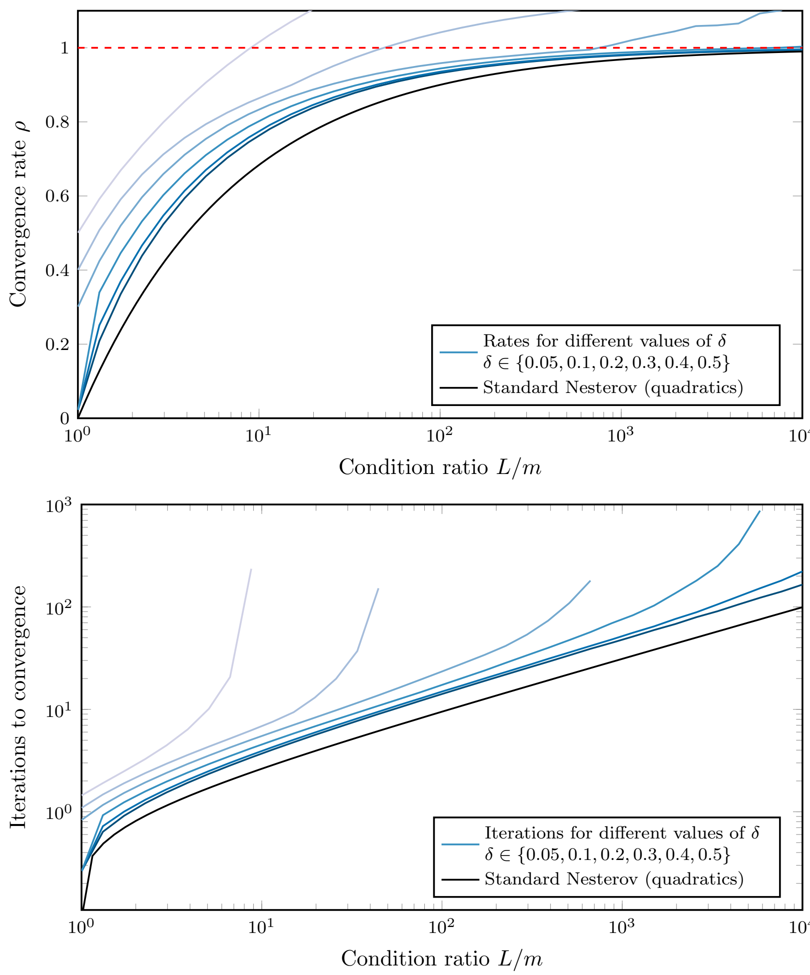


Figure 12: Convergence rate and iterations to convergence for Nesterov's method with standard tuning, for various noise parameters δ.

$x - \Pi_\nu(x) \in \nu T(\Pi_\nu(x))$. Or, written another way,

$$\Pi_\nu = (I + \nu T)^{-1} \tag{5.3}$$

Since T is a subdifferential, it satisfies the *incremental passivity* condition. Namely,

$$(T(x) - T(y))^\mathsf{T}(x - y) \geq 0 \qquad \text{for all } x, y \in \mathbb{R}^n$$

Therefore, T satisfies the sector IQC with $m = 0$ and $L = \infty$. In fact, via minor modifications of Lemma 8 and Lemma 9 using the definition of a subdifferential rather than (3.13c), T satisfies the off-by-one and weighted off-by-one IQCs as well. Now transform (5.2) by introducing the auxiliary signals $u_k := \nabla f(y_k)$, $w_k := \Pi_\nu(y_k - \alpha u_k)$, $v_k := \nu T(w_k)$. The definitions of w_k and v_k together with (5.3) immediately imply that $w_k = y_k - \alpha u_k - v_k$. Therefore, we can rewrite (5.2) as

$$\begin{aligned} \xi_{k+1} &= \xi_k + \beta(\xi_k - \xi_{k-1}) - v_k - \alpha u_k \\ w_k &= \xi_k + \beta(\xi_k - \xi_{k-1}) - v_k - \alpha u_k \\ y_k &= \xi_k + \beta(\xi_k - \xi_{k-1}) \end{aligned} \qquad \text{with:} \qquad \begin{aligned} u_k &= \nabla f(y_k) \\ v_k &= \nu T(w_k) \end{aligned}$$

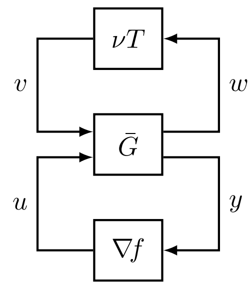


Figure 13: Block-diagram representation of the standard interconnection with an additional block νT representing a scaled subdifferential.

These equations may be succinctly represented as a block diagram, as in Figure 13.

Analyzing this interconnection is done by accounting for the IQCs for both unknown blocks. If (Ψ_1, M_1) is the IQC for ∇f with output z_k^1 and (Ψ_2, M_2) is the IQC for νT with output z_k^2, then we seek to show that for all trajectories satisfy

$$x_{k+1}^\mathsf{T} P x_{k+1} - \rho\, x_k^\mathsf{T} P x_k + \lambda_1 (z_k^1)^\mathsf{T} M_1 (z_k^1) + \lambda_2 (z_k^2)^\mathsf{T} M_2 (z_k^2) \le 0 \tag{5.4}$$

where x_k now includes the states ξ_k as well as the internal states of Ψ_1 and Ψ_2. As in the proof of Theorem 4, for each fixed ρ, we can write (5.4) as an LMI in the variables $P \succ 0$, $\lambda_1 \ge 0$, $\lambda_2 \ge 0$.

Applying this approach to the proximal version of Nesterov's accelerated method, we recover the exact same plots as in Figure 3. This is to be expected because it is known that the proximal gradient and accelerated methods achieves the same worst-case convergence rates as their unconstrained counterparts [1, 25, 40]. We conjecture that any algorithm G of the form (2.2) which converges with rate ρ has a proximal variant that converges at precisely the same rate.

5.4 Weakly convex functions

With minor modifications to our analysis, we can immediately extend our results to the case where the function to be optimized is convex, but not strongly convex. Specifically, we will assume throughout this subsection that $f \in S(0, L)$. The following development is due to Elad Hazan [10].

Suppose we want to minimize f over a compact, convex domain $\mathcal{D}$ for which we can readily compute the Euclidean projection. Let R denote the diameter of the set $\mathcal{D}$. Define the function $f_\varepsilon(x) := f(x) + \frac{\varepsilon}{2R^2}\|x\|^2$. Note that f_ε is differentiable and strongly convex; it satisfies $f_\varepsilon \in S(\frac{\varepsilon}{R^2}, L + \frac{\varepsilon}{R^2})$. Therefore, we may apply our analysis to f_ε.

Suppose we execute on f_ε an algorithm with interleaved projections as in Section 5.3. Let $x_\star$ be any minimizer of f on $\mathcal{D}$ and $x_\star^{(\varepsilon)}$ be the minimizer of f_ε. Let ρ denote the rate of convergence achieved when the condition ratio is set as $\kappa = (1 + LR^2/\varepsilon)$ and let $P_\varepsilon \succ 0$ be the associated solution to the LMI. Let $\sigma := \mathrm{cond}(P_\varepsilon)$. After k steps,

$$\begin{aligned} f(x_k) - f(x_\star) &= f_\varepsilon(x_k) - f_\varepsilon(x_\star) + \frac{\varepsilon}{2R^2}\left(\|x_\star\|^2 - \|x_k\|^2\right) \\ &\le f_\varepsilon(x_k) - f_\varepsilon(x_\star^{(\varepsilon)}) + \frac{\varepsilon}{2R^2}\left(\|x_\star\|^2 - \|x_k\|^2\right) \\ &\le f_\varepsilon(x_k) - f_\varepsilon(x_\star^{(\varepsilon)}) + \frac{\varepsilon}{2} \end{aligned}$$

Now apply (3.13a) from Proposition 5 using $(f,x,y) = (f_\varepsilon, x_\star^{(\varepsilon)}, x_k)$ and obtain

$$\begin{aligned} f(x_k) - f(x_\star) &\le \frac{LR^2+\varepsilon}{2R^2}\|x_k - x_\star^{(\varepsilon)}\|^2 + \frac{\varepsilon}{2} \\ &\le \frac{LR^2+\varepsilon}{2R^2}\,\sigma\rho^{2k}\|x_0 - x_\star^{(\varepsilon)}\|^2 + \frac{\varepsilon}{2} \\ &\le \frac{1}{2}\left((LR^2+\varepsilon)\,\sigma\rho^{2k} + \varepsilon\right). \end{aligned}$$

Where the last inequality follows from the definition of set diameter. Therefore, if

$$k \ge \frac{\log\bigl((1+LR^2/\varepsilon)\,\sigma\bigr)}{2\log(\rho^{-1})}\,, \tag{5.5}$$

then $f(x_k) - f(x_\star) \le \varepsilon$. Substituting the rates found algebraically for the quadratic case (Section 2.2) or our numerical results for the strongly convex case (Sections 4.4–4.5), the convergence rate ρ satisfies

$$\begin{aligned} &\frac{1}{\log(\rho^{-1})} \propto \kappa = (1+LR^2/\varepsilon) && \text{for the Gradient method, and} \\ &\frac{1}{\log(\rho^{-1})} \propto \kappa^{1/2} = (1+LR^2/\varepsilon)^{1/2} && \text{for Nesterov's accelerated method.} \end{aligned}$$

Finally, note that $\sigma = \operatorname{cond}(P_\varepsilon)$ also depends on ε. We can control the growth of σ directly by including a constraint of the form $I \preceq P \preceq \sigma I$ when solving the SDP of Theorem 4. Alternatively, we can observe (see Figure 4) that $\sigma \propto \kappa = (1+LR^2/\varepsilon)$. Therefore, we conclude that

$$\begin{aligned} &k = \mathcal{O}(\tfrac{1}{\varepsilon}\log\tfrac{1}{\varepsilon}) && \text{for the Gradient method, and} \\ &k = \mathcal{O}(\tfrac{1}{\sqrt{\varepsilon}}\log\tfrac{1}{\varepsilon}) && \text{for Nesterov's accelerated method.} \end{aligned}$$

This analysis matches the standard bounds up to the logarithmic terms [23].

6 Algorithm design

In this section, we show one way in which the IQC analysis framework can be used for algorithm design. We saw in Section 5.2 that the Gradient method can be very robust to noise (Figure 11), or not robust at all (Figure 10), depending on whether we use a stepsize of $\alpha = 1/L$ or $\alpha = 2/(L+m)$, respectively.

A natural question to ask is whether such a trade-off between performance and robustness exists with Nesterov's method as well. As can be seen in Figure 12, Nesterov's method is only somewhat robust to noise. In the sequel, we will synthesize variants of Nesterov's method that explore the performance-robustness trade-off space.

Consider an algorithm of the form (2.2). Based on the discussion in Section 2.1, we know A must have an eigenvalue of 1. Moreover, given any invertible T, the algorithms (A,B,C,D) and $(TAT^{-1}, TB, CT^{-1}, D)$ are *equivalent realizations* in the sense that if one is stable with rate ρ, the other is stable with rate ρ as well. Indeed, if the first algorithm has state ξ_k, the second algorithm has state $T\xi_k$. We limit our search to the case $A \in \mathbb{R}^{2\times 2}$ and $D = 0$. Three parameters are required to characterize all possible algorithms in this family (modulo equivalences due to a choice of T). One possible parameterization is given by

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} \beta_1+1 & -\beta_1 & -\alpha \\ 1 & 0 & 0 \\ \hline \beta_2+1 & -\beta_2 & 0 \end{array}\right] \qquad \text{with:}\quad (\alpha,\beta_1,\beta_2)\in\mathbb{R}^3 \tag{6.1}$$

In light of the discussion in Section 2, we see that the Gradient, Heavy-ball, and Nesterov methods are all special cases of (6.1). In particular,

$$(\alpha, \beta_1, \beta_2) \text{ is equal to: } \begin{cases} (\alpha, 0, 0) & \text{for the Gradient method} \\ (\alpha, \beta, 0) & \text{for the Heavy-ball method} \\ (\alpha, \beta, \beta) & \text{for Nesterov's method} \end{cases}$$

We may also rewrite (6.1) in more familiar recursion form as

$$\xi_{k+1} = \xi_k - \alpha \nabla f(y_k) + \beta_1(\xi_k - \xi_{k-1}) \tag{6.2a}$$
$$y_k = \xi_k + \beta_2(\xi_k - \xi_{k-1}) \tag{6.2b}$$

Our approach is straightforward: for each choice of condition ratio L/m and noise strength δ, we generate a large grid of tuples $(\alpha, \beta_1, \beta_2)$ and use the approach of Section 5.2 to evaluate each algorithm. We then choose the algorithm with the lowest ρ. In other words, given bounds on the condition ratio and noise strength, we choose the algorithm for which we can certify the best possible convergence rate over all admissible choices of f and gradient noise. The performance of each optimized algorithm is plotted in Figure 14.

By design, this new family of algorithms must have a performance superior to the Gradient method, Nesterov's method, and the Heavy-ball method for any choice of tuning parameters. In the limit $\delta \to 0$, we appear to recover the performance of Nesterov's method when it is applied to *quadratics*. That is, we have used numerical search to find an algorithm whose worst case performance guarantee is slightly better than what is guaranteed by Nesterov's method.

In the second plot of Figure 14, the algorithms robust to higher noise levels have greater slopes. When the noise level is low ($\delta = 0.01$), we approach a slope of 0.5, the same as Nesterov. When the noise level is high ($\delta = 0.5$), the slope is roughly 0.75. Note that the Gradient method, which was robust for *all* noise levels, has a slope of 1. Therefore, the new algorithms we found explore the trade-off between noise robustness and performance, and may be useful in instances where Nesterov's method would be too fragile and the Gradient method would be too slow.

7 Future work

We are only beginning to get a sense of what IQCs can tell us about optimization schemes, and there are many more control theory tools and techniques left to adapt to the context of optimization and machine learning. We conclude this paper with several interesting directions for future work.

Analytic proofs One of the drawbacks of our numerical proofs is that we are always pushing up against numerical error and conditioning error. Analytic proofs would alleviate this issue and could provide more interpretable results about how parameters of algorithms should vary to meet performance and robustness demands. To provide such analytic proofs, one would have to solve small LMIs in closed form. This amounts to solving small semidefinite programming problems, and this may be doable using analytic tools from algebraic geometry [9, 36].

Lower Bounds Our IQC conditions are merely sufficient for verifying the convergence of an optimization problem. However, as pointed out by Megretski and Rantzer, the derived conditions are necessary in a restricted sense [19]. If we fail to find a solution

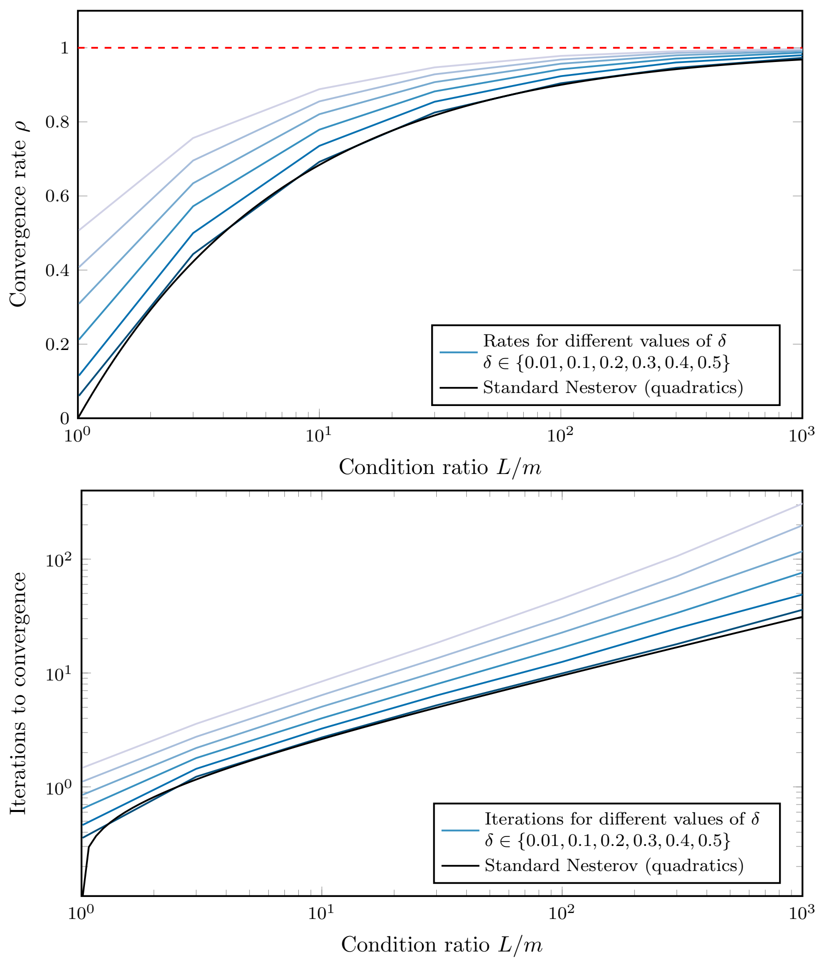


Figure 14: Upper bounds found using a brute-force search over the three-parameter family of algorithms described by (6.2). Convergence rate is shown (first plot) as is the number of iterations required to achieve convergence to a specified tolerance (second plot). Although the bounds assume strongly convex functions, we also show the worst-case rate for quadratics as a comparison.

to our LMI, then there is necessarily a sequence of point that satisfy all of the IQC constraints and that do not converge to an equilibrium [20, 39]. It is thus possible that this tool can be used to construct a convex function to serve as a counterexample for convergence. This intuition was what guided our construction of a counterexample for the convergence of the Heavy-ball method. It may be possible that this construction can be generalized to systematically produce counterexamples.

Time-varying algorithms In many practical scenarios, we know neither the Lipschitz constant L nor the strong convexity parameter m. Under such conditions, some sort of estimation scheme is used to choose the appropriate step size. This could be a simple backoff scheme to ensure a sufficient decrease, or a more intricate search method to find the appropriate parameters [27]. From our control vantage point, it may be possible to use techniques from adaptive control to certify when such line search methods are stable. In particular, these could be used to differentiate between the different sorts of schemes used to choose the parameters of the nonlinear conjugate gradient method. Useful connections are made between robustness analysis of adaptive controllers and Lyapunov theory in [37].

A related area of study is that of linear parameter varying (LPV) systems. This extension of linear systems analysis considers parameterized variations in the dynamical system matrices (A, B, C, D). Algorithms with variable stepsize are examples of LPV systems. Some recent work discussing IQCs applied to LPV systems appeared in [31]. Another possible direction would be to use optimal control techniques directly to choose algorithm parameters, possibly solving a small SDP at every iteration to choose new assignments.

Algorithm synthesis Perhaps even more ambitiously than using our framework for parameter selection, our initial results show that we can use IQCs as a way of designing new algorithms. We restricted our attention to algorithms with one-step of memory, as then we only had to search over 3 parameters. However, new techniques would be necessary to explore more complicated algorithms. Local search heuristics could be used here to probe the feasible region of the associated LMIs, but convex methods and convex relaxations may also be applicable and should be investigated for these searches.

Noise analysis Our robustness analysis only allows us to consider certain forms of deterministic noise. Expanding our techniques to study stochastic noise would expand the applicability of our techniques and could provide new insight into popular stochastic optimization algorithms such as stochastic coordinate descent and stochastic gradient descent [22, 24]. Many of the most common techniques for proving convergence of stochastic methods rely on Lyapunov-type arguments, and we may be able to generalize this approach to account for the variety of different methods. In order to expand our techniques to this space, we would need to introduce IQCs that were valid *in expectation*. Stability methods from stochastic control may be applicable to such investigations.

Beyond convexity Since our analysis decouples the derivation of constraints on function classes from the algorithm analysis, it is possible that it can be generalized to nonconvex optimization. If we can characterize the function class by reasonable quadratic constraints, our framework immediately applies, and may lead to entirely new analyses for nonconvex function classes. For example, IQCs for saturating nonlinearities are readily available in the controls literature [15, 19]. From a complementary perspective, if we know that our function is not merely convex, but has additional structure, this can be incorporated as additional IQCs. With extra constraints, it is possible that we can derive faster rates or more robustness for smaller function classes.

Non-quadratic Lyapunov functions There has been substantial work in the past decade on efficient algorithms to search over *non-quadratic* Lyapunov functions [29, 30]. These techniques use sum-of-squares hierarchies to certify that non-quadratic polynomials are nonnegative, and still reduce to solving small semidefinite programming problems. This more general class of Lyapunov functions could be better matched to certain classes of functions than quadratics, and we could perhaps analyze more complicated algorithms and interconnections.

Large-scale composite system analysis Perhaps the most ambitious goal of this program is to move beyond convex models and attempt to analyze complicated optimization systems used in science and industry. Powerful modeling languages like AMPL or GAMS allow for local analysis of large, complex systems, and certifying that the decisions about these systems are valid and safe would have impact in a variety of fields including process technology, web-scale analytics, and power management. Since our methods nicely abstract beyond two interconnected systems, it is our hope that they can be extended to analyze the variety of optimization algorithms deployed to handle large, high throughput data processing.

Acknowledgments

We would like to thank Peter Seiler for many helpful pointers on time-domain IQCs, Elad Hazan for his suggestion of how to analyze functions that are not strongly convex, and Bin Hu for pointing out a misreading of Nesterov's results in an earlier draft of this paper. We would also like to thank Ali Jadbabaie, Pablo Parrilo, and Stephen Wright for many helpful discussions and suggestions.

LL and AP are partially supported by AFOSR award FA9550-12-1-0339 and NASA Grant No. NRA NNX12AM55A. BR is generously supported by ONR awards N00014-11-1-0723 and N00014-13-1-0129, NSF award CCF-1148243, AFOSR award FA9550-13-1-0138, and a Sloan Research Fellowship. This research was also supported in part by NSF CISE Expeditions Award CCF-1139158, LBNL Award 7076018, and DARPA XData Award FA8750-12-2-0331, and gifts from Amazon Web Services, Google, SAP, The Thomas and Stacey Siebel Foundation, Adobe, Apple, Inc., C3Energy, Cisco, Cloudera, EMC, Ericsson, Facebook, GameOnTalis, Guavus, HP, Huawei, Intel, Microsoft, NetApp, Pivotal, Splunk, Virdata, Fanuc, VMware, and Yahoo!.

References

[1] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[2] S. Becker, E. J. Candès, and M. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation*, 3(3):165–218, 2011.

[3] S. P. Boyd and L. El Ghaoui. Method of centers for minimizing generalized eigenvalues. *Linear algebra and its applications*, 188:63–111, 1993.

[4] S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM, 1994.

[5] O. Devolder, F. Glineur, and Y. Nesterov. First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1-2):37–75, 2014.

[6] J. Doyle. Analysis of feedback systems with structured uncertainties. *Control Theory and Applications, IEE Proceedings D*, 129(6):242–250, 1982.

[7] Y. Drori and M. Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, pages 1–32, 2013.

[8] M. Grant and S. P. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

[9] D. R. Grayson and M. E. Stillman. Macaulay 2, a software system for research in algebraic geometry, 2002.

[10] E. Hazan. Personal Communication.

[11] W. P. Heath and A. G. Wills. Zames-Falb multipliers for quadratic programming. In *IEEE Conference on Decision and Control*, pages 963–968, 2005.

[12] U. Jönsson. A nonlinear Popov criterion. In *IEEE Conference on Decision and Control*, volume 4, pages 3523–3527, 1997.

[13] R. E. Kalman. Lyapunov functions for the problem of Lur'e in automatic control. *Proceedings of the National Academy of Sciences*, 49(2):201, 1963.

[14] H. K. Khalil. *Nonlinear systems (3rd edition)*. Prentice Hall, 2002.

[15] V. Kulkarni, S. K. Bohacek, and M. G. Safonov. Robustness of interconnected systems with controller saturation and bounded delays. In *American Control Conference*, 1999.

[16] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[17] A. I. Lur'e and V. N. Postnikov. On the theory of stability of control systems. *Applied mathematics and mechanics*, 8(3):246–248, 1944. In Russian.

[18] A. M. Lyapunov and A. T. Fuller. *General Problem of the Stability Of Motion*. Control Theory and Applications Series. Taylor & Francis, 1992. Original text in Russian, 1892.

[19] A. Megretski and A. Rantzer. System analysis via integral quadratic constraints. *IEEE Transactions on Automatic Control*, 42(6):819–830, 1997.

[20] A. Megretski and S. Treil. Power distribution inequalities in optimization and robustness of uncertain systems. *Journal of Mathematical Systems, Estimation, and Control*, 3(3):301–319, 1993.

[21] A. Nemirovski. The long-step method of analytic centers for fractional problems. *Mathematical Programming*, 77(1):191–224, 1997.

[22] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[23] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, Boston, MA, 2004.

[24] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

[25] Y. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

[26] Y. Nesterov and A. Nemirovskii. *Interior-point polynomial methods in convex programming*. SIAM, 1994.

[27] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, second edition, 2006.

[28] A. Packard and J. Doyle. The complex structured singular value. *Automatica*, 29(1):71–109, 1993.

[29] A. Papachristodoulou and S. Prajna. On the construction of Lyapunov functions using the sum of squares decomposition. In *IEEE Conference on Decision and Control*, volume 3, pages 3482–3487, 2002.

[30] P. A. Parrilo and S. Lall. Semidefinite programming relaxations and algebraic optimization in control. *European Journal of Control*, 9(2):307–321, 2003.

[31] H. Pfifer and P. Seiler. Robustness analysis of linear parameter varying systems using integral quadratic constraints. In *American Control Conference*, pages 4476–4481, 2014.

[32] B. T. Polyak. Introduction to optimization. *Optimization Software, Inc.*, 1987.

[33] V. M. Popov. Absolute stability of nonlinear systems of automatic control. *Automation and Remote Control*, 22(8):857–875, 1962. Original text in Russian, 1961.

[34] A. Rantzer and A. Megretski. Stability criteria based on Integral Quadratic Constraints. In *IEEE Conference on Decision and Control*, volume 1, pages 215–220, 1996.

[35] A. Rantzer and A. Megretski. System analysis via Integral Quadratic Constraints, part II. Technical Report ISRN LUTFD2/TFRT--7559--SE, Department of Automatic Control, Lund University, Sweden, 1997.

[36] P. Rostalski and B. Sturmfels. Dualities in convex algebraic geometry. *arXiv preprint arXiv:1006.4894*, 2010.

[37] S. Sastry and M. Bodson. *Adaptive control: stability, convergence and robustness.* Courier Dover Publications, 2011.

[38] P. Seiler. Stability analysis with dissipation inequalities and integral quadratic constraints. *IEEE Transactions on Automatic Control*, 60(6):1704–1709, 2015.

[39] J. S. Shamma. Robustness analysis for time-varying systems. In *IEEE Conference on Decision and Control*, 1992.

[40] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. *submitted to SIAM Journal on Optimization*, 2008.

[41] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, 1986.

[42] J. C. Willems. Dissipative dynamical systems—Part I: General theory and Part II: Linear systems with quadratic supply rates. *Archive for Rational Mechanics and Analysis*, 45(5):321–351,352–393, 1972.

[43] V. A. Yakubovich. Frequency conditions for the absolute stability of control systems with several nonlinear or linear nonstationary units. *Avtomatika i Telemekhanika*, pages 5–30, 1967. In Russian.

[44] V. A. Yakubovich. S-procedure in nonlinear control theory. *Vestnik Leningrad University*, 4:73–93, 1977. Original text in Russian, 1971.

[45] V. A. Yakubovich. Nonconvex optimization problem: The infinite-horizon linear-quadratic control problem with quadratic constraints. *Systems & Control Letters*, 19(1):13–22, 1992.

[46] G. Zames. On the input-output stability of time-varying nonlinear feedback systems—Part I: Conditions derived using concepts of loop gain, conicity, and positivity, and Part II: Conditions involving circles in the frequency plane and sector nonlinearities. *IEEE Transactions on Automatic Control*, 11(2,3):228–238,465–476, 1966.

[47] G. Zames and P. L. Falb. Stability conditions for systems with monotone and slope-restricted nonlinearities. *SIAM Journal on Control*, 6(1):89–108, 1968.

A Proof of Proposition 1

Suppose Q has eigenvalues that satisfy $0 < m \le \lambda_d \le \lambda_{d-1} \le \cdots \le \lambda_2 \le \lambda_1 \le L$. Throughout, we assume (A, B, C) are the state transition matrices of the algorithm we would like to analyze (see Section 2). The state transition matrices are functions of the algorithm parameters (e.g. α and β for the Heavy-ball method). Let T be the closed loop system $T := A + BQC$. The worst-case convergence rate is found by maximizing the spectral radius over all admissible Q. In other words,

$$\rho_{\text{worst}} = \underset{mI_d \preceq Q \preceq LI_d}{\text{maximize}}\ \rho(T)$$

The first observation is that for our algorithms of interest, we may assume $d = 1$. To see why, take for example the Heavy-ball method, where

$$T = \begin{bmatrix} (1+\beta)I_d - \alpha Q & -\beta I_d \\ I_d & 0_d \end{bmatrix}$$

Write the eigenvalue decomposition of Q as $Q = U\Lambda U^{\mathsf{T}}$, where $\Lambda = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d)$ and U is orthogonal. Then,

$$T = \begin{bmatrix} U & 0_d \\ 0_d & U \end{bmatrix} \begin{bmatrix} (1+\beta)I_d - \alpha\Lambda & -\beta I_d \\ I_d & 0_d \end{bmatrix} \begin{bmatrix} U & 0_d \\ 0_d & U \end{bmatrix}^{\mathsf{T}}$$

Therefore, by similarity, the eigenvalues of T are the eigenvalues of all the matrices

$$T_i = \begin{bmatrix} (1+\beta) - \alpha\lambda_i & -\beta \\ 1 & 0 \end{bmatrix}, \quad i = 1, 2, \ldots, d.$$

and we may without loss of generality let $d = 1$. The simplified problem is therefore

$$\rho_{\text{worst}} = \underset{m \le \lambda \le L}{\text{maximize}}\ \rho(T_0)$$

where T_0 is defined as

$$T_0 = \begin{cases} 1 - \alpha\lambda & \text{Gradient method} \\ \begin{bmatrix} (1+\beta)(1-\alpha\lambda) & -\beta(1-\alpha\lambda) \\ 1 & 0 \end{bmatrix} & \text{Nesterov's method} \\ \begin{bmatrix} 1+\beta-\alpha\lambda & -\beta \\ 1 & 0 \end{bmatrix} & \text{Heavy-ball method} \end{cases}$$

It is now a matter of algebraic substitution to find the optimal rates for each parameter choice. For example, with the Gradient method,

$$\begin{aligned} \rho_{\max} &= \underset{m \le \lambda \le L}{\text{maximize}}\ \rho(1 - \alpha\lambda) \\ &= \max\Big\{ |1 - \alpha m|, |1 - \alpha L| \Big\} \end{aligned}$$

The second equality follows from the fact that the maximum of a convex function must occur at the boundary. We can now see that when $\alpha = 1/L$, we have $\rho_{\max} = 1 - 1/\kappa$. Finding the optimal α is straightforward in this case because the pointwise maximum of convex functions is itself convex. In this case, the minimum $\rho_{\max}$ occurs when $\alpha = \frac{2}{L+m}$ and the result is $\rho_{\max} = \frac{\kappa-1}{\kappa+1}$.

The analyses for Nesterov's method and the Heavy-ball method are similar in spirit to that of the Gradient method, but computing the spectral radius is more complicated. For Nesterov's method, we have

$$\rho_{\max} = \underset{m\le\lambda\le L}{\text{maximize}}\ \max\bigl\{|\nu_1|,|\nu_2|\bigr\}$$

where ν_1, ν_2 are the roots of the characteristic polynomial of T_0, which is

$$\nu^2 - (1+\beta)(1-\alpha\lambda)\nu + \beta(1-\alpha\lambda) = 0$$

The magnitudes of the roots satisfy:

$$\max\bigl\{|\nu_1|,|\nu_2|\bigr\} = \begin{cases} \frac{1}{2}\bigl|(1+\beta)(1-\alpha\lambda)\bigr| + \frac{1}{2}\sqrt{\Delta} & \text{if } \Delta \ge 0 \\ \sqrt{\beta(1-\alpha\lambda)} & \text{otherwise} \end{cases}$$

where $\Delta := (1+\beta)^2(1-\alpha\lambda)^2 - 4\beta(1-\alpha\lambda)$. It is straightforward to verify that if α,β are fixed, $\max\bigl\{|\nu_1|,|\nu_2|\bigr\}$ is a continuous and quasiconvex function of λ. So, the maximum over λ must occur at boundary point. For the case where $\alpha = \frac{1}{L}$ and $\beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$, choosing $\lambda = L$ yields zero, so the maximum must be achieved at $\lambda = m$, which yields

$$\rho_{\max} = \sqrt{\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\left(1-\frac{1}{\kappa}\right)} = \sqrt{\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\cdot\frac{(\sqrt{\kappa}+1)(\sqrt{\kappa}-1)}{\kappa}} = 1 - \frac{1}{\sqrt{\kappa}},$$

as required. When optimizing over quadratic functions, the above tuning of Nesterov's method is suboptimal. Finding the choice of α and β that yields the smallest $\rho_{\max}$ requires careful examination of several subcases, and we omit the details in the interest of space. The result is shown in the second last row of the table in Proposition 1.

A similar eigenvalue analysis was used in [32] to derive the optimal parameter tuning for the Heavy-ball method applied to quadratics (last row of the table).

B Proof of the Heavy-ball counterexample

We would like to minimize the function whose gradient is given by (4.11). The Heavy-ball method with $L = 25$ and $m = 1$ is given in Section 2.2:

$$x_{k+1} = \tfrac{13}{9}x_k - \tfrac{4}{9}x_{k-1} - \tfrac{1}{9}\nabla f(x_k) \tag{B.1}$$

where we use the initialization $x_{-1} = x_0$. Based on the plot of Figure 6, we will look for limit points p, q, r such that we have a cycle of period 3:

$$x_{3n} \to p, \quad x_{3n+1} \to q, \quad x_{3n+2} \to r \qquad \text{for } n = 0, 1, \dots \tag{B.2}$$

where $p < 1$, $q < 1$, and $r > 2$. Substituting the forms (B.2) and (4.11) directly into (B.1), we obtain the system of linear equations

$$\begin{bmatrix} 4 & 12 & 9 \\ 9 & 4 & 12 \\ 12 & 9 & 4 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 0 \\ 24 \\ 0 \end{bmatrix}$$

and the unique solution of these equations is

$$p = \frac{792}{1225} \approx 0.65, \quad q = -\frac{2208}{1225} \approx -1.80, \quad r = \frac{2592}{1225} \approx 2.12 \tag{B.3}$$

In other words, the trajectory (B.2) with values (B.3) is a fixed point of (B.1). Let us call this limit sequence $\{x_k^\star\}_{k\geq 0}$. In order to show that the limit cycle is attractive (nearby trajectories will eventually converge to the cycle) consider a perturbed version of this sequence $\{x_k^\star + \varepsilon_k\}_{k\geq 0}$. If we assume that the k^{th} iterate still belongs to the same piece of the function (e.g. if $x_k^\star < 1$ then $x_k^\star + \varepsilon_k < 1$, and if $x_k^\star > 2$ then $x_k^\star + \varepsilon_k > 2$) then we can use the Heavy-ball equations to compute the perturbation in the subsequent iterate. Upon doing so, we find that $\{\varepsilon_k\}_{k\geq 0}$ must satisfy

$$\begin{bmatrix}\varepsilon_{k+2}\\ \varepsilon_{k+1}\end{bmatrix} = \underbrace{\begin{bmatrix}-\frac{4}{3} & -\frac{4}{9}\\ 1 & 0\end{bmatrix}}_{P}\begin{bmatrix}\varepsilon_{k+1}\\ \varepsilon_k\end{bmatrix} \qquad \text{for all } k$$

It is immediate that $\rho(P) = \frac{2}{3} < 1$ so as long as no transient value of ε_k strays too far from zero and causes an unscheduled crossing of the dotted lines on Figure 6, then we will have $\varepsilon_k \to 0$ and the limit cycle will be attractive. Undesired transient behavior can be ruled out by ensuring that the error eventually decreases monotonically. One can easily verify that

$$\left\|P^8\right\|^2 \approx 0.46044 < \frac{1}{2}$$

where $\|\cdot\|$ is the induced 2-norm. Therefore, P^8 is a contraction, and we have:

$$\varepsilon_{k+8}^2 \leq \left\|\begin{bmatrix}\varepsilon_{k+9}\\ \varepsilon_{k+8}\end{bmatrix}\right\|^2 \leq \left\|P^8\right\|^2\left\|\begin{bmatrix}\varepsilon_{k+1}\\ \varepsilon_k\end{bmatrix}\right\|^2 < \frac{1}{2}(\varepsilon_{k+1}^2 + \varepsilon_k^2) \tag{B.4}$$

If eight consecutive perturbations $\{\varepsilon_i^2, \varepsilon_{i+1}^2, \dots, \varepsilon_{i+7}^2\}$ are each less than some $\bar{\varepsilon}^2$, then apply (B.4) twice to conclude that

$$\varepsilon_{i+8}^2 < \frac{1}{2}(\varepsilon_i^2 + \varepsilon_{i+1}^2) < \bar{\varepsilon}^2 \qquad \text{and} \qquad \varepsilon_{i+9}^2 < \frac{1}{2}(\varepsilon_{i+1}^2 + \varepsilon_{i+2}^2) < \bar{\varepsilon}^2$$

Continuing in this fashion, we conclude that the entire tail $\{\varepsilon_k^2\}_{k\geq i}$ also satisfies the bound $\varepsilon_k^2 < \bar{\varepsilon}^2$. The closest that our proposed limit cycle comes to a transition point of $f(x)$ (either 1 or 2) is $r - 2 = \frac{142}{1225} \approx 0.1159$. Therefore, if we set this number to be $\bar{\varepsilon}$, and we can find eight consecutive iterates of the Heavy-ball method that are each within $\bar{\varepsilon}$ of the limit cycle, then the remaining iterates must converge exponentially to the limit cycle. It is straightforward to check that if $x_0 = 3.3$, then the iterates $x_4, x_5, \dots, x_{11}$ are each within a distance $\bar{\varepsilon}$ of their respective limit points. Therefore, the limit cycle is attractive.