Lyapunov analyses for first-order methods

Theory, automation, and algorithm design

Upadhyaya, Manu

Lyapunov analyses for first-order methods: Theory, automation, and algorithm design

MANU UPADHYAYA
DEPARTMENT OF AUTOMATIC CONTROL | LUND UNIVERSITY

Lyapunov analyses for first-order methods

Theory, automation, and algorithm design

Manu Upadhyaya

LUND UNIVERSITY

Department of Automatic Control

Cover image by Jóna Berglind Stefánsdóttir.
Printed with permission.

PhD Thesis TFRT-1150
ISBN 978-91-8104-589-5 (print)
ISBN 978-91-8104-590-1 (web)
ISSN 0280–5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Media-Tryck.
Lund 2025

Abstract

This thesis, comprising four research papers, contributes to the field of systematic and computer-aided analyses and design of first-order methods. The first two papers focus on developing new methodologies, while the remaining two apply these techniques to refine complexity and convergence results, as well as to design new methods. A central theme of the work is the use of Lyapunov-type analyses, a structured proof technique with historical roots in the study of dynamical systems, which is widely used to establish complexity and convergence properties of first-order methods.

Acknowledgments

I am deeply grateful to my advisors for their guidance and support throughout this work. My main advisor, Pontus Giselsson, ignited my enthusiasm from the earliest days of my PhD, consistently offering inventive ideas and steady encouragement. I am likewise indebted to Sebastian Banert, whose mathematical expertise and openness to thoughtful discussions enriched my research experience.

My sincere appreciation goes to Adrien Taylor, whose deep expertise in performance estimation problems—a pivotal element of this thesis—led to many valuable discussions and insights, significantly enriching my understanding and shaping this work. I also thank him for generously hosting me at Inria Paris. Additionally, I warmly thank Puya Latafat for inspiring me to pursue novel research directions in the later stages of my PhD.

I extend heartfelt gratitude to the Department of Automatic Control at Lund University for fostering a research environment characterized by collegiality, stimulating discussions, lively coffee breaks, engaging social events, and memorable trips. Special thanks to the dedicated administrative staff and research engineers, whose tireless efforts ensure everything runs smoothly.

Lastly, I thank my friends for their camaraderie and laughter along the way and my family for their continuous encouragement and support.

Funding

This work was supported by the ELLIIT Strategic Research Area and, in part, by the Wallenberg AI, Autonomous Systems and Software Program (WASP), a program funded by the Knut and Alice Wallenberg Foundation.

Contents

1

Introduction

Mathematical optimization, or mathematical programming, is a branch of applied mathematics that provides a formal framework for, e.g., decision-making, prediction, and modeling in various fields in science and engineering. Optimization has historical roots intertwined with the development of calculus and physics—emerging naturally in concepts such as least action and minimal energy configurations—and has since evolved to tackle problems across diverse domains. Its applicability spans automatic control, where optimization methods guide feedback mechanisms and optimal control policies, as well as economics, underpinning rational resource allocation, market equilibrium modeling, and strategic interactions studied through game theory. Inverse problems, frequently applied in fields like system identification, signal processing, astronomy, and medical imaging, leverage optimization to infer unknown parameters or structures from indirect or noisy observations. Operations research and management science also extensively rely on optimization to support improved decision-making, addressing challenges in resource scheduling, logistics, facility location, and strategic planning across various sectors, including business, healthcare, transportation, and government. Moreover, machine learning and artificial intelligence fundamentally rely on optimization to train predictive models, such as neural networks, that learn patterns from large-scale datasets; recent advances include generative models, where optimization techniques facilitate the synthesis of realistic data such as text, images, and video. Given the wide range and importance of these applications, the theoretical and computational study of optimization methods has long been an important focus in mathematics.

One of the central elements behind the success of optimization methods in modern applications, especially those involving large-scale problems, is first-order methods. Relying solely on gradient or subgradient information, they have become widely adopted for their computational efficiency, scalability, and adaptability to various problem structures. As a type of iterative method, they start from an initial value and generate a sequence of progressively better approximate solutions. Significant theoretical and algorithmic advancements have led to the adoption of these techniques; these include accelerated methods and proximal splitting techniques as well as stochastic, adaptive, and non-Euclidean variants.

A critical component in understanding the effectiveness of first-order methods is their complexity and convergence properties. The work [57] formalized complexity theory in the context of optimization, establishing a framework to understand the inherent difficulty of optimization tasks by defining bounds on the number of computations required to solve optimization problems within a specified accuracy. A more recent reference for these types of complexity analyses is given in [58]. These analyses are typically done in a black-box setting where the functional components that define the optimization problem are constrained to lie in some predefined function classes, and these functional components can only be queried at each iteration via some oracle model that provides quantitative local information such as function values and (sub)gradients. This allows us to characterize upper and lower complexity bounds and optimal optimization methods with respect to these bounds. Related to these complexity analyses, but not always the same, is whether the sequence of iterates produced by an optimization algorithm converges to a solution.

Starting with the seminal work [30], new computer-aided approaches have emerged in the literature for performing these analyses. These tools automate the most challenging aspects of the analysis process, specifically creating mathematical proofs, a task previously performed manually by experts. By formulating complexity and convergence analyses as structured problems that a computer can solve, the tools can produce results significantly faster than a human could. Moreover, they can identify proofs that, in practice, are beyond the capabilities of an expert, as they systematically explore all possible proofs within a certain structure. This means an enormous number of alternatives can be examined, something that would be practically impossible to do manually. Moreover, these approaches have not only validated and extended existing complexity and convergence results but also inspired the design of entirely new classes of methods that advance the state of the art. This research direction highlights a rapidly evolving interface between classical theory, computational experimentation, and algorithmic innovation. This thesis aims to contribute precisely to this promising research direction by advancing both the theoretical foundations and practical methodologies.

1.1 Organization

The thesis is organized as follows. Section 1.2 introduces the notation and mathematical background used throughout the introductory chapters, largely following the book [8]. Chapter 2 provides high-level context that situates the papers included in this thesis. Chapter 3 summarizes the main contributions of each included paper and describes the author's specific role in each. Chapter 4 discusses limitations and outlines directions for future research. The remainder of the thesis comprises four self-contained papers, each of which can be read independently.

1.2 Notation and preliminaries

Let $\mathbb{N}_0$ denote the set of nonnegative integers, $\mathbb{R}$ the set of real numbers, $\mathbb{R}_+$ the set of nonnegative real numbers, $\mathbb{R}_{++}$ the set of positive real numbers, $\mathbb{R}^n$ the set of all n-tuples of elements of $\mathbb{R}$, $\mathbb{R}^{m\times n}$ the set of real-valued matrices of size $m \times n$, and if $M \in \mathbb{R}^{m\times n}$ then $[M]_{i,j}$ denotes the i,j-th element of M.

Let $\mathcal{H}$ denote a real Hilbert space[1], i.e., a complete real inner product space. If not explicitly stated otherwise, norms $\|\cdot\|$ are canonical norms, i.e., $\|\cdot\|^2 = \langle\cdot,\cdot\rangle$, where the inner product $\langle\cdot,\cdot\rangle$ will be clear from the context. The sum of subsets of $\mathcal{H}$ is interpreted in the Minkowski sense, i.e., if $X, Y \subseteq \mathcal{H}$, then $X + Y = \{x + y \mid x \in X, y \in Y\}$. The power set of $\mathcal{H}$ is denoted by $2^{\mathcal{H}}$, i.e., $2^{\mathcal{H}} = \{X \mid X \subseteq \mathcal{H}\}$. A sequence $(x^k)_{k\in\mathbb{N}_0} \in \mathcal{H}^{\mathbb{N}_0}$ is said to *converge weakly* to $x \in \mathcal{H}$, denoted $x^k \rightharpoonup x$, if $\lim_{k\to\infty}\langle x^k, y\rangle = \langle x, y\rangle$ for each $y \in \mathcal{H}$, and

[1] For readers unfamiliar with Hilbert spaces, it suffices to consider $\mathcal{H} = \mathbb{R}^n$ equipped with the standard dot product.

to *converge strongly* to x, denoted $x^k \to x$, if $\lim_{k\to\infty}\|x^k - x\| = 0$. If $\dim \mathcal{H} < +\infty$, then weak and strong convergence are equivalent, and we simply say that $(x^k)_{k\in\mathbb{N}_0}$ converges to x. We denote the identity mapping $x \mapsto x$ on $\mathcal{H}$ by Id.

Let $\mathcal{G}$ be another real Hilbert space, and let $L : \mathcal{H} \to \mathcal{G}$ be a linear operator. We define the *operator norm* of L as $\|L\| = \sup \{\|Lx\| \mid x \in \mathcal{H} \text{ such that } \|x\| \leq 1\}$. We say that L is *bounded* if $\|L\| < +\infty$, which holds if and only if L is continuous. If L is bounded, there exists a unique bounded linear operator $L^* : \mathcal{G} \to \mathcal{H}$, called the *adjoint* of L, such that $\langle Lx, y\rangle = \langle x, L^*y\rangle$ for each $x \in \mathcal{H}$ and $y \in \mathcal{G}$.

If $M \in \mathbb{R}^{m\times n}$, we define the tensor product $M \otimes \mathrm{Id}$ to be the mapping $(M \otimes \mathrm{Id}) : \mathcal{H}^n \to \mathcal{H}^m$ such that

$$(M \otimes \mathrm{Id})\boldsymbol{z} = \left(\sum_{j=1}^{n}[M]_{1,j}z_j, \ldots, \sum_{j=1}^{n}[M]_{m,j}z_j\right)$$

for each $\boldsymbol{z} = (z_1, \ldots, z_n) \in \mathcal{H}^n$, i.e., for each row in M, we take a linear combination of the elements of $\boldsymbol{z}$ with the coefficients given by the corresponding row.

Functions

Below, we define several properties of functions that are commonly assumed when analyzing and designing first-order methods. Although weaker assumptions may suffice, the listed conditions often simplify the analysis and make the methods less complicated; therefore, we adopt them to make the presentation more accessible. The first two conditions exclude trivial or pathological cases, and the last three ensure that the function is well-behaved.

Definition 1.2.1
Let $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ and $\beta, \sigma \in \mathbb{R}_+$. The function f is said to be

(i) proper *if* $\operatorname{dom} f \neq \emptyset$, *where the set* $\operatorname{dom} f = \{x \in \mathcal{H} \mid f(x) < +\infty\}$ *is called the* effective domain *of f,*

(ii) lower semicontinuous *if* $\liminf_{y\to x} f(y) \geq f(x)$ *for each* $x \in \mathcal{H}$,

(iii) convex *if* $f((1-\lambda)x + \lambda y) \leq (1-\lambda)f(x) + \lambda f(y)$ *for each* $x, y \in \mathcal{H}$ *and* $0 \leq \lambda \leq 1$,

(iv) σ-strongly convex *if* $f - (\sigma/2)\|\cdot\|^2$ *is convex, and*

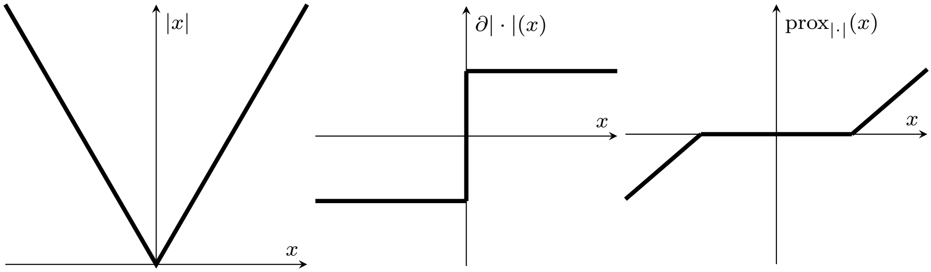
(v) β-smooth *if f is Fréchet differentiable and the gradient $\nabla f : \mathcal{H} \to \mathcal{H}$ is β-Lipschitz continuous, i.e.,* $\|\nabla f(x) - \nabla f(y)\| \leq \beta\|x - y\|$ *for each* $x, y \in \mathcal{H}$.

Next, we define the concept of a subdifferential, which can be seen as a generalization of the gradient for convex functions that may not be differentiable.

Definition 1.2.2
The subdifferential *of a function $f : \mathcal{H} \to \mathbb{R}\cup\{+\infty\}$ is the set-valued operator $\partial f : \mathcal{H} \to 2^{\mathcal{H}}$ defined as the mapping*

$$x \mapsto \{u \in \mathcal{H} \mid \forall y \in \mathcal{H},\ f(y) \geq f(x) + \langle u, y - x\rangle\}.$$

The following fact justifies the comment above. If the function $f : \mathcal{H} \to \mathbb{R}$ is convex and Fréchet differentiable, then $\partial f(x) = \{\nabla f(x)\}$ for each $x \in \mathcal{H}$ [8, Proposition 17.31]. We call elements $u \in \partial f(x)$ *subgradients* of f at x.



(a) Graph of $|\cdot|$ **(b)** Graph of $\partial|\cdot|$ **(c)** Graph of $\mathrm{prox}_{|\cdot|}$

Figure 1.1 (a) The absolute-value function, (b) its subdifferential mapping, and (c) its proximal operator.

Example 1.2.3
The subdifferential of the absolute-value function $|\cdot|:\mathbb{R}\to\mathbb{R}$ *is given by*

$$\partial|\cdot|(x) = \begin{cases} \{1\}, & x>0, \\ [-1,1], & x=0, \\ \{-1\}, & x<0. \end{cases}$$

The graphs of the absolute-value function and its subdifferential mapping are shown in Figures 1.1a and 1.1b, respectively.

Fermat's rule states that $\mathrm{Argmin}_{x\in\mathcal{H}} f(x) = \{x\in\mathcal{H} \mid 0\in\partial f(x)\}$ [8, Theorem 16.3], and gives an optimality condition for minimization. For example, we see from Figure 1.1b that $x=0$ is a minimizer of the absolute-value function, since $0\in\partial|\cdot|(0)$.

The *proximal operator*, defined below, is particularly useful for minimizing nonsmooth functions, as we will see in Chapter 2.

Definition 1.2.4
Let $f:\mathcal{H}\to\mathbb{R}\cup\{+\infty\}$ *be proper, convex, and lower semicontinuous, and* $\gamma\in\mathbb{R}_{++}$*. Then, the* proximal operator *of* f *with* step size γ*, denoted* $\mathrm{prox}_{\gamma f}:\mathcal{H}\to\mathcal{H}$*, is defined as the single-valued operator (see, e.g., [8, Proposition 12.15]) given by*

$$\mathrm{prox}_{\gamma f}(x) = \underset{z\in\mathcal{H}}{\mathrm{argmin}}\left(f(z)+\frac{1}{2\gamma}\|x-z\|^2\right)$$

for each $x\in\mathcal{H}$*.*

There are fast and exact methods available for computing the proximal operator of many commonly used functions[2]. Below, we give one such example.

Example 1.2.5
The proximal operator of the absolute-value function $|\cdot|:\mathbb{R}\to\mathbb{R}$ *with step size* $\gamma\in\mathbb{R}_{++}$ *is given by*

$$\mathrm{prox}_{\gamma|\cdot|}(x) = \mathrm{sign}(x)\max\{|x|-\gamma,0\}$$

for each $x\in\mathbb{R}$*. The graph of the proximal operator of the absolute-value function with step size* 1 *is shown in Figure 1.1c.*

[2] E.g., see `https://proximity-operator.net` and `https://juliafirstorder.github.io/ProximalOperators.jl`.

An interesting observation is that the proximal operator implicitly samples a subgradient. In particular, under the same assumptions as in Definition 1.2.4, if $x, p \in \mathcal{H}$, then (e.g., see [8, Proposition 16.44, Proposition 16.6])

$$p = \operatorname{prox}_{\gamma f}(x) \quad \Leftrightarrow \quad \gamma^{-1}(x - p) \in \partial f(p).$$

Transformations enable us to study problems from alternative perspectives and can often simplify their analysis. In particular, primal and dual formulations of optimization problems offer complementary insights, with the convex conjugate playing a central role in defining dual problems. Moreover, a combined primal-dual viewpoint can inform the design of efficient methods tailored to the problem's structure.

Definition 1.2.6
Let $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ be proper, convex, and lower semicontinuous. Then, the convex conjugate *of f, denoted $f^* : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, is the proper, convex, and lower semicontinuous function given by*

$$f^*(u) = \sup_{x \in \mathcal{H}} \left(\langle u, x \rangle - f(x) \right)$$

for each $u \in \mathcal{H}$ [8, Corollary 13.38].

For example, under the assumption of Definition 1.2.6, if $x, u \in \mathcal{H}$, then (e.g., see [8, Theorem 16.29])

$$u \in \partial f(x) \quad \Leftrightarrow \quad x \in \partial f^*(u).$$

Operators

Recall that the subdifferential of a function is a set-valued operator and that Fermat's rule states that minimizing the function is the same as solving an inclusion problem with respect to its subdifferential. This abstract point of view has motivated the analysis and design of first-order methods that solve inclusion problems of set-valued operators, as we will see in Chapter 2. First, we give two basic definitions.

Definition 1.2.7
Let $G : \mathcal{H} \to 2^{\mathcal{H}}$ be a set-valued operator. Then,

(i) the set of zeros *of G is denoted by $\operatorname{zer} G = \{x \in \mathcal{H} \mid 0 \in G(x)\}$, and*

(ii) the graph *of G is denoted by $\operatorname{gra} G = \{(x, y) \in \mathcal{H} \times \mathcal{H} \mid y \in G(x)\}$.*

For example, if $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, then Fermat's rule can be written as $\operatorname{Argmin}_{x \in \mathcal{H}} f(x) = \operatorname{zer} \partial f$.

As with functions, we need to impose some suitable assumptions on the operators to develop methods whose convergence can be rigorously guaranteed.

Definition 1.2.8
Let $G : \mathcal{H} \to 2^{\mathcal{H}}$ and $\mu \in \mathbb{R}_+$. The operator G is said to be

(i) monotone *if $\langle u - v, x - y \rangle \geq 0$ for each $(x, u), (y, v) \in \operatorname{gra} G$,*

(ii) maximally monotone *if G is monotone and there does not exist a monotone operator $H : \mathcal{H} \to 2^{\mathcal{H}}$ such that $\operatorname{gra} G \subsetneq \operatorname{gra} H$, and*

(iii) μ-strongly monotone *if $\langle u - v, x - y \rangle \geq \mu \|x - y\|^2$ for each $(x, u), (y, v) \in \operatorname{gra} G$.*

The definitions above are motivated by the following facts: Suppose that $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$. Then the following hold:

(i) if f is proper, then ∂f is monotone [8, Example 20.3],

(ii) if f is proper, convex, and lower semicontinuous, then ∂f is maximally monotone [8, Theorem 20.25], and

(iii) if f is proper and μ-strongly convex, then ∂f is μ-strongly monotone [8, Example 22.4].

Next, we introduce the concept of a resolvent, which can be viewed as a generalization of the proximal operator for maximally monotone operators. To this end, we define the *inverse* of $G : \mathcal{H} \to 2^{\mathcal{H}}$, denoted by $G^{-1} : \mathcal{H} \to 2^{\mathcal{H}}$, which is defined through its graph

$$\operatorname{gra} G^{-1} = \{(y, x) \in \mathcal{H} \times \mathcal{H} \mid (x, y) \in \operatorname{gra} G\}.$$

Definition 1.2.9
Suppose that $G : \mathcal{H} \to 2^{\mathcal{H}}$ is maximally monotone and let $\gamma \in \mathbb{R}_{++}$. Then, the resolvent *of G with* step size *γ, denoted $J_{\gamma G} : \mathcal{H} \to \mathcal{H}$, is defined by*

$$(\operatorname{Id} + \gamma G)^{-1}(x) = \{J_{\gamma G}(x)\}$$

for each $x \in \mathcal{H}$, since $(\operatorname{Id} + \gamma G)^{-1}$ is singleton-valued in this case [8, Proposition 23.10].

The following fact justifies the comment above. If $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is proper, convex, and lower semicontinuous, then ∂f is maximally monotone (as previously noted), and

$$J_{\gamma \partial f} = \operatorname{prox}_{\gamma f}$$

for any $\gamma \in \mathbb{R}_{++}$ [8, Example 23.3].

We will also consider single-valued operators, which are analogous to gradients in the functional setting.

Definition 1.2.10
Let $G : \mathcal{H} \to \mathcal{H}$ be a single-valued operator.

(i) Let $\beta \in \mathbb{R}_{+}$. The operator G is said to be β-Lipschitz continuous *if*

$$\|G(x) - G(y)\| \leq \beta \|x - y\|$$

for each $x, y \in \mathcal{H}$.

(ii) Let $\beta \in \mathbb{R}_{++}$. The operator G is said to be $(1/\beta)$-cocoercive *if*

$$\langle G(x) - G(y), x - y \rangle \geq (1/\beta) \|G(x) - G(y)\|^2$$

for each $x, y \in \mathcal{H}$.

The Cauchy–Schwarz inequality implies that any $(1/\beta)$-cocoercive operator is β-Lipschitz continuous. The converse does not hold in general, but it does hold for gradients of convex functions, as illustrated by the following proposition.

Proposition 1.2.11
If $\beta \in \mathbb{R}_{++}$ and $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is convex and Fréchet differentiable, then ∇f is β-Lipschitz continuous if and only if ∇f is $(1/\beta)$-cocoercive.

Proof. See [8, Corollary 18.17]. □

In Chapter 2, we will briefly discuss fixed-point algorithms, for which we need the following concept. The set of *fixed points* of $G : \mathcal{H} \to \mathcal{H}$ is

$$\operatorname{fix} G = \{x \in \mathcal{H} \mid x = G(x)\}.$$

We conclude this chapter by introducing a convention that enables us to treat single-valued and singleton-valued operators interchangeably.

(i) For notational convenience (at the expense of a slight abuse of notation), we will sometimes identify the operator $G : \mathcal{H} \to \mathcal{H}$ with the set-valued mapping $\mathcal{H} \ni x \mapsto \{G(x)\} \subseteq \mathcal{H}$, which will be clear from context. For example, if $x, y \in \mathcal{H}$, the inclusion $y \in G(x)$ should be interpreted as the equality $y = G(x)$.

(ii) Similarly, if $G : \mathcal{H} \to 2^{\mathcal{H}}$ and $T : \mathcal{H} \to \mathcal{H}$ satisfy $G(x) = \{T(x)\}$ for each $x \in \mathcal{H}$, i.e., G is a singleton-valued operator, we will sometimes identify G with the corresponding single-valued operator T.

2

Background

In this thesis, we consider optimization problems that can be written as

$$\underset{x\in\mathcal{H}}{\text{minimize}}\ \ f(x), \tag{2.1}$$

where $\mathcal{H}$ is a real Hilbert space and $f\colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is a function that satisfies certain properties, depending on the application. In the context of decision making, we can think of the optimization problem as a choice problem, where we want to choose an element x from a set of alternatives $\mathcal{H}$ that minimizes the cost $f(x)$.

As discussed in Chapter 1, many problems in science and engineering can be formulated as optimization problems. However, only a few of these problems can be solved in practice. Moreover, we typically cannot expect to solve such problems exactly via a finite sequence of operations; instead, we must resort to iterative schemes that generate sequences of increasingly accurate approximate solutions.

In this context, convex optimization problems, i.e., problems where the function f is convex, are particularly well-studied [8, 14, 33, 45, 53, 59, 61, 62, 75]. Many problems of this type can be solved reliably and efficiently, and they have been applied across a wide range of fields [12, 13, 16, 19, 43, 44, 58].

Due to increased computational power, the availability of large datasets, and the high dimensionality of many modern problems, the focus has shifted from small- and medium-scale problems to large-scale problems. In this setting, first-order methods, which rely only on gradient or subgradient information, have proven particularly useful due to their favorable scalability and modest per-iteration computational cost [10, 21, 31, 37, 60, 70].

In the context of first-order methods and nonsmooth convex optimization, proximal algorithms, also called splitting methods, have gained significant attention [9, 15, 24, 25, 32, 38, 47, 51]. These methods decompose complex objective functions into additive components that are easier to manage. E.g., suppose that the objective function f can be written as $f = \sum_{i=1}^{m} f_i$, where each function $f_i\colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is proper, convex, lower semicontinuous, and possibly smooth. Then, splitting methods are designed to solve the inclusion problem[1]

$$\text{find } x \in \mathcal{H} \ \text{ such that } \ 0 \in \sum_{i=1}^{m} \partial f_i(x) \tag{2.2}$$

[1] Solutions to the inclusion problem (2.2) are always solutions to the optimization problem (2.1) with $f = \sum_{i=1}^{m} f_i$. The converse holds under mild assumptions. See, e.g., [8, Corollary 16.50].

by designing iterative schemes that utilize the proximal operator $\operatorname{prox}_{f_i}$ for nonsmooth components and the gradient ∇f_i for smooth components.

For example, if $m = 2$, f_1 is β-smooth for some $\beta \in \mathbb{R}_{++}$, and f_2 is nonsmooth, then the proximal gradient method is given by

$$x^{k+1} = \operatorname{prox}_{\gamma f_2}\left(x^k - \gamma \nabla f_1(x^k)\right) \tag{2.3}$$

for some step size $\gamma \in (0, 2/\beta)$, and the sequence $(x^k)_{k\in\mathbb{N}_0}$ generated by (2.3) converges weakly to a solution of the inclusion problem (2.2) (if a solution exists) [8, Theorem 26.14]. Suppose instead that both f_1 and f_2 are nonsmooth. Then, the Douglas–Rachford method [27, 32, 51] is applicable and is given by

$$\begin{aligned} y^k &= \operatorname{prox}_{\gamma f_1}\left(x^k\right), \\ z^k &= \operatorname{prox}_{\gamma f_2}\left(2y^k - x^k\right), \\ x^{k+1} &= x^k + \lambda(z^k - y^k) \end{aligned} \tag{2.4}$$

for some step size $\gamma \in \mathbb{R}_{++}$ and relaxation parameter $\lambda \in (0, 2)$. In particular, the sequences $(y^k)_{k\in\mathbb{N}_0}$ and $(z^k)_{k\in\mathbb{N}_0}$ generated by (2.4) converge weakly to a solution of the inclusion problem (2.2) (if a solution exists) [8, Theorem 26.11].

Convergence rates and complexity

Knowing that an algorithm will eventually reach a solution is only half the story. Convergence rates quantify how many iterations are required to attain a desired accuracy, can guide the choice of algorithm parameters, and help compare different methods.

For example, for the proximal gradient method (2.3), [10, Theorem 10.16] implies that

$$\|x^{n+1} - x^\star\|^2 \leq \|x^n - x^\star\|^2 - \frac{2}{\beta}(f_1(x^{n+1}) + f_2(x^{n+1}) - f_1(x^\star) - f_2(x^\star)) \tag{2.5}$$

for any solution[2] $x^\star$ to the inclusion problem (2.2), if $\gamma = 1/\beta$. Summing inequality (2.5) over $n = 0, \ldots, k$, rearranging terms, and using $\|x^{k+1} - x^\star\|^2 \geq 0$, we get that

$$\sum_{n=0}^{k} (f_1(x^{n+1}) + f_2(x^{n+1}) - f_1(x^\star) - f_2(x^\star)) \leq \frac{\beta\|x^0 - x^\star\|^2}{2}. \tag{2.6}$$

Using the observation that (e.g., see [10, Remark 10.20])

$$f_1(x^{n+1}) + f_2(x^{n+1}) \leq f_1(x^n) + f_2(x^n), \tag{2.7}$$

we can conclude from (2.6) that

$$f_1(x^{k+1}) + f_2(x^{k+1}) - f_1(x^\star) - f_2(x^\star) \leq \frac{\beta\|x^0 - x^\star\|^2}{2(k+1)}, \tag{2.8}$$

and we say that the function-value suboptimality $(f_1(x^k) + f_2(x^k) - f_1(x^\star) - f_2(x^\star))_{k\in\mathbb{N}_0}$ converges to zero at a sublinear rate of $\mathcal{O}(1/k)$. Moreover, (2.8) implies that the proximal

[2] Equivalently, any $x^\star \in \operatorname{Argmin}_{x\in\mathcal{H}} (f_1(x) + f_2(x))$

gradient method (2.3) requires $k+1 \geq (\beta\|x^0 - x^\star\|^2)/(2\epsilon)$ iterations to reach an accuracy of $f_1(x^k) + f_2(x^k) - f_1(x^\star) - f_2(x^\star) \leq \epsilon$, and we say that the proximal gradient method has an iteration complexity of $\mathcal{O}(1/\epsilon)$.

The key inequalities in the analysis above are (2.5) and (2.7). These inequalities are commonly referred to as Lyapunov inequalities, since they ensure descent in critical quantities, such as distance to a solution or function-value suboptimality. When used to establish convergence rates, such as in (2.8), this approach is called a Lyapunov analysis. Such Lyapunov analyses form a fundamental component in the study of many first-order methods.

Computer-aided analyses

An important question is whether the convergence rate in (2.8) for the proximal gradient method can be improved, and more generally, whether convergence-rate analyses for other methods can be strengthened. The performance estimation problem (PEP) methodology, first introduced in [30] and formalized in [68, 69], provides a systematic way to obtain unimprovable (also known as tight) convergence rates and complexity analyses for a large class of first-order methods. The PEP methodology poses the search of a worst-case example from a predefined class of problems for the algorithm and performance measure under consideration as an optimization problem (called the performance estimation problem). This is then reformulated in a sequence of steps to arrive at a semidefinite program, whose exact solution can be recovered up to numerical precision using existing software. This can then sometimes be used to obtain closed-form convergence rates extracted from the numerical solution of the semidefinite program, or using computer algebra software.

The standard PEP methodology is based on a fixed iteration count (or horizon) $k \in \mathbb{N}$, meaning the performance estimation problem must be solved for $k = 1, 2, 3, \ldots$. This approach faces two main limitations. First, the number of variables and constraints in the semidefinite program grows quadratically with the number of iterations k, making numerical solutions prohibitive even for moderate values of k. Consequently, deriving closed-form convergence rates also becomes increasingly challenging as k grows. Second, results obtained for a fixed iteration count k may provide limited insight, as they do not directly generalize or guarantee algorithmic behavior beyond this fixed horizon.

These limitations have motivated the development of Lyapunov-based approaches to analyzing the performance of first-order methods, which partially or fully overcome these limitations [54, 66, 67]. The key idea is to restrict convergence-rate proof patterns to the search for key Lyapunov inequalities, which can then equivalently be reformulated as more tractable semidefinite programs, where the number of variables and constraints is either independent of the horizon k or grows only linearly with k. Compared to the standard PEP approach, this provides a more tractable framework for deriving closed-form convergence rates and producing proofs that are generally more concise and accessible. Although these types of Lyapunov analyses introduce some a priori conservatism, extracting closed-form results from the standard PEP approach can often be practically infeasible. Lyapunov-based proof patterns thus represent a pragmatic compromise with higher chances of obtaining closed-form convergence rates, while still being sufficient to yield tight convergence rates for many methods and settings.

Another closely related Lyapunov-based approach involves integral quadratic constraints (IQCs), a technique from robust control theory [52]. IQCs were first adapted for analyzing first-order methods in [50] and subsequently extended in various works [36, 49, 65, 73]. These approaches share a common feature; they represent first-order methods as linear systems interconnected through feedback with nonlinear mappings.

Such representations, widely used in nonlinear systems analysis, offer compact algorithm descriptions.

The methodological developments presented in this thesis build upon both methodologies: the worst-case analysis and tightness guarantees provided by PEP, and the compact algorithm representations offered by IQCs. A first step towards combining these methodologies was taken in [67], which served as a foundation for further formalization, broader applicability, and accessibility through a software package.

Generalization to operator inclusions

Under the assumptions in (2.2), each subdifferential ∂f_i is maximally monotone. Thus, the inclusion problem

$$\text{find } x \in \mathcal{H} \text{ such that } 0 \in \sum_{i=1}^{m} G_i(x), \tag{2.9}$$

where each operator $G_i\colon \mathcal{H} \to 2^{\mathcal{H}}$ is maximally monotone is a generalization of (2.2), i.e., (2.9) provides a formalism that covers convex optimization problems [8, 64], but also certain equilibrium problems [17, 18, 22] and so-called variational inequalities [34, 35, 48].

Similar to above, it is possible to design iterative splitting schemes where resolvents J_{G_i} are utilized for set-valued operators and direct evaluations for single-valued operators (e.g., operators that are Lipschitz continuous or cocoercive) [6, 7, 8, 23]. For example, both the proximal gradient method (2.3) (called the forward-backward method in the operator setting) and the Douglas–Rachford method (2.4) have counterparts in this more general setting, see, e.g., [8, Section 26.5 and 26.3].

Moreover, the traditional PEP methodology has been extended to this setting [63].

Fixed-point theory

The design and analysis of many splitting methods, including both the function and operator cases, can be understood through the lens of fixed-point theory in nonlinear analysis, where the goal is to find fixed points of an algorithmic operator [6, 7, 8, 23], i.e., some operator $\mathcal{T} : \mathcal{H} \to \mathcal{H}$ and point $x \in \operatorname{fix} \mathcal{T}$, and then potentially perform some simple transformation of x to obtain a solution. For example, the proximal gradient method (2.3) can be interpreted as a fixed-point iteration of the operator

$$\mathcal{T}^{\mathrm{PG}} = \operatorname{prox}_{\gamma f_2} \circ (\mathrm{Id} - \gamma \nabla f_1)$$

where $x \in \operatorname{fix} \mathcal{T}^{\mathrm{PG}}$ is a solution to the inclusion problem (2.2) in this case. Similarly, the Douglas–Rachford method (2.4) can be interpreted as a fixed-point iteration of the operator

$$\mathcal{T}^{\mathrm{DR}} = \mathrm{Id} + \lambda(\operatorname{prox}_{\gamma f_2} \circ (2\operatorname{prox}_{\gamma f_1} - \mathrm{Id}) - \operatorname{prox}_{\gamma f_1}).$$

where the point $\operatorname{prox}_{\gamma f_1}(x)$ for any $x \in \operatorname{fix} \mathcal{T}^{\mathrm{DR}}$ is a solution to the inclusion problem (2.2) in this case.

3

Contributions

This chapter presents the thesis's main contributions, comprising four research papers, and describes the author's specific role in every paper.

3.1 Scientific contributions

Paper I

M. Upadhyaya, S. Banert, A. B. Taylor, and P. Giselsson. "Automated tight Lyapunov analysis for first-order methods". *Mathematical Programming* **209** (2025), pp. 133–170
Status: Accepted

This paper considers inclusion problems of the form

$$\text{find } y \in \mathcal{H} \text{ such that } 0 \in \sum_{i=1}^{m} \partial f_i(y), \tag{3.1}$$

where each function $f_i : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is proper, (possibly strongly) convex, lower semicontinuous, and possibly smooth, and corresponding algorithms that can be written as[1]

$$\begin{aligned} \boldsymbol{x}^{k+1} &= (A \otimes \mathrm{Id})\boldsymbol{x}^k + (B \otimes \mathrm{Id})\boldsymbol{u}^k, \\ \boldsymbol{y}^k &= (C \otimes \mathrm{Id})\boldsymbol{x}^k + (D \otimes \mathrm{Id})\boldsymbol{u}^k, \\ \boldsymbol{u}^k &\in \boldsymbol{\partial f}(\boldsymbol{y}^k) \triangleq \prod_{i=1}^{m} \partial f_i(y_i^k), \\ \boldsymbol{F}^k &= \boldsymbol{f}(\boldsymbol{y}^k) \triangleq (f_1(y_1^k), \ldots, f_m(y_m^k)), \end{aligned} \tag{3.2}$$

where $\boldsymbol{\xi}^k = (\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{y}^k, \boldsymbol{F}^k) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$ contains the algorithm variables, and

$$A \in \mathbb{R}^{n\times n}, \qquad B \in \mathbb{R}^{n\times m}, \qquad C \in \mathbb{R}^{m\times n}, \qquad D \in \mathbb{R}^{m\times m} \tag{3.3}$$

[1] In this context, the symbol Π is used for Cartesian products.

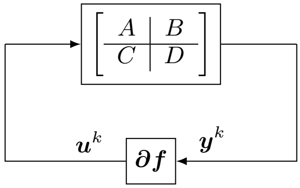


Figure 3.1 Block diagram of the state-space representation (3.2), which can be interpreted as a discrete-time linear system in feedback interconnection with the potentially nonlinear and set-valued subdifferentials that define the inclusion problem (3.1).

are fixed matrices containing the parameters of the method at hand. Figure 3.1 illustrates the state-space representation (3.2) in a block diagram.

We call points $\boldsymbol{\xi}^\star = (\boldsymbol{x}^\star, \boldsymbol{u}^\star, \boldsymbol{y}^\star, \boldsymbol{F}^\star) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$ such that

$$\begin{aligned}
\boldsymbol{x}^\star &= (A \otimes \mathrm{Id})\boldsymbol{x}^\star + (B \otimes \mathrm{Id})\boldsymbol{u}^\star, \\
\boldsymbol{y}^\star &= (C \otimes \mathrm{Id})\boldsymbol{x}^\star + (D \otimes \mathrm{Id})\boldsymbol{u}^\star, \\
\boldsymbol{u}^\star &\in \boldsymbol{\partial f}(\boldsymbol{y}^\star), \\
\boldsymbol{F}^\star &= \boldsymbol{f}(\boldsymbol{y}^\star),
\end{aligned} \tag{3.4}$$

fixed points.

The paper's first two contributions are as follows:

(i) We give a necessary and sufficient condition on the matrices (3.3) such that solving the inclusion problem (3.1) is equivalent to finding a fixed point (3.4).

(ii) We give a sufficient condition for the update equation in (3.2) to be well-posed in the sense that there exist variables $(\boldsymbol{x}^{k+1}, \boldsymbol{u}^k, \boldsymbol{y}^k)$ for any $\boldsymbol{x}^k$. In doing so, we show that the state-space representation (3.2) encompasses all first-order methods that use fixed linear combinations of a sliding window of previously computed quantities and sample each subdifferential at most once per iteration, either through an explicit gradient evaluation or an implicit proximal evaluation.

Next, we consider Lyapunov inequalities, which can be used to establish both linear and sublinear convergence rates, of the form

$$\mathcal{V}(\boldsymbol{\xi}^{k+1}, \boldsymbol{\xi}^\star) \leq \rho \mathcal{V}(\boldsymbol{\xi}^k, \boldsymbol{\xi}^\star) - \mathcal{R}(\boldsymbol{\xi}^k, \boldsymbol{\xi}^\star), \tag{3.5}$$

where $\mathcal{V}$ and $\mathcal{R}$ are nonnegative functions called the Lyapunov function and residual function, respectively, and $\rho \in [0, 1]$. For example, if $\rho \in [0, 1)$, then an inductive argument shows that

$$\mathcal{V}(\boldsymbol{\xi}^k, \boldsymbol{\xi}^\star) \leq \rho^k \mathcal{V}(\boldsymbol{\xi}^0, \boldsymbol{\xi}^\star),$$

from which we can conclude that the Lyapunov function converges to zero linearly (or geometrically). Instead, if $\rho = 1$, then a telescoping summation argument shows that $\mathcal{R}(\boldsymbol{\xi}^k, \boldsymbol{\xi}^\star)$ is summable (and therefore converges to zero) and, e.g.,

$$\min_{i=0,\ldots,k} \mathcal{R}(\boldsymbol{\xi}^i, \boldsymbol{\xi}^\star) \leq \frac{\mathcal{V}(\boldsymbol{\xi}^0, \boldsymbol{\xi}^\star)}{k+1},$$

i.e., the best-so-far residual function converges to zero at a sublinear rate. Additionally, if $\mathcal{R}(\boldsymbol{\xi}^{k+1},\boldsymbol{\xi}^\star) \leq \mathcal{R}(\boldsymbol{\xi}^{k},\boldsymbol{\xi}^\star)$, then

$$\mathcal{R}(\boldsymbol{\xi}^{k},\boldsymbol{\xi}^\star) \leq \frac{\mathcal{V}(\boldsymbol{\xi}^{0},\boldsymbol{\xi}^\star)}{k+1}.$$

Thus, the goal is to identify Lyapunov inequalities (3.5) where $\mathcal{V}$ and $\mathcal{R}$ are directly or indirectly related to optimality measures of the inclusion problem (3.1).

The paper's last two contributions are as follows:

(iii) We consider Lyapunov functions and residual functions, under quadratic ansatzes, that relate to optimality measures of the inclusion problem (3.1) and show that the existence of a Lyapunov inequality (3.5) is equivalent[2] to the feasibility of a semidefinite program, which can be solved efficiently using existing software.

(iv) We numerically verify (and sometimes extend) numerous established convergence results.

Paper II

M. Upadhyaya, A. B. Taylor, S. Banert, and P. Giselsson. "AutoLyap: A Python package for computer-assisted Lyapunov analyses for first-order methods". (2025)
Status: Preprint

This paper generalizes Paper I in the following ways:

(i) The inclusion problem (3.1) is extended to the more general form

$$\text{find } y \in \mathcal{H} \quad \text{such that} \quad 0 \in \sum_{i\in\mathcal{I}_{\text{func}}} \partial f_i(y) + \sum_{i\in\mathcal{I}_{\text{op}}} G_i(y), \tag{3.6}$$

where $\mathcal{I}_{\text{func}}$ and $\mathcal{I}_{\text{op}}$ are finite index sets, and each function $f_i : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ and operator $G_i : \mathcal{H} \to 2^{\mathcal{H}}$ is chosen from some user-specified function class and operator class, respectively.

(ii) The state-space representation (3.2) is extended to (1) include the operators G_i, (2) allow for multiple evaluations of the same subdifferential ∂f_i (and operator G_i) at each iteration, and (3) allow for the possibility for the matrices A, B, C, and D to be iteration dependent.

(iii) The class of Lyapunov inequalities (3.5) is extended to a more general form, and a class of iteration-dependent Lyapunov inequalities is introduced. Moreover, similar to Paper I, we show that the existence of these Lyapunov inequalities is equivalent[3] to the feasibility of specific semidefinite programs.

Besides the theoretical contributions, the paper introduces a software package called AutoLyap, which collects the theoretical results and provides a user-friendly interface for performing these Lyapunov analyses in Python.

[2] Under some mild assumptions.
[3] Under some mild assumptions.

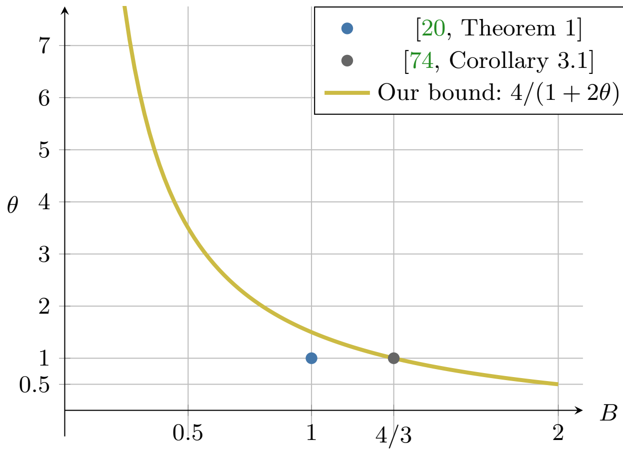


Figure 3.2 Convergence region of the Chambolle–Pock method (3.9), where convergence is guaranteed if $\tau\sigma\|L\|^2 < B$. For a fixed θ, increasing the upper bound B enlarges the convergence region.

Paper III

S. Banert, M. Upadhyaya, and P. Giselsson. "The Chambolle–Pock method converges weakly with $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$" (2023)
Status: In submission

This paper considers the Chambolle–Pock method [20], also known as the primal-dual hybrid gradient method, which solves convex-concave saddle-point problems of the form[4]

$$\underset{x\in\mathcal{H}}{\text{minimize}}\,\underset{y\in\mathcal{G}}{\text{maximize}}\;\; f(x) + \langle Lx, y\rangle - g^*(y), \tag{3.7}$$

where $f\colon \mathcal{H} \to \mathbb{R}\cup\{+\infty\}$ and $g\colon \mathcal{G} \to \mathbb{R}\cup\{+\infty\}$ are proper, convex, and lower semi-continuous functions, g^* is the convex conjugate of g, $L\colon \mathcal{H} \to \mathcal{G}$ is a bounded linear operator, and $\mathcal{H}$ and $\mathcal{G}$ are real Hilbert spaces, under the assumption that there exists a solution $(x^\star, y^\star) \in \mathcal{H}\times\mathcal{G}$ that satisfies the Karush–Kuhn–Tucker (KKT) condition

$$\begin{aligned} -L^*y^\star &\in \partial f(x^\star), \\ Lx^\star &\in \partial g^*(y^\star). \end{aligned} \tag{3.8}$$

The Chambolle–Pock method is given by

$$\begin{aligned} x^{k+1} &= \operatorname{prox}_{\tau f}\big(x^k - \tau L^* y^k\big), \\ y^{k+1} &= \operatorname{prox}_{\sigma g^*}\big(y^k + \sigma L(x^{k+1} + \theta(x^{k+1} - x^k))\big), \end{aligned} \tag{3.9}$$

[4] The saddle-point problem (3.7) is a primal-dual formulation of the primal optimization problem

$$\underset{x\in\mathcal{H}}{\text{minimize}}\;\; f(x) + g(Lx).$$

where $\tau, \sigma \in \mathbb{R}_{++}$ are primal and dual step sizes, respectively, and $\theta \in \mathbb{R}$ is a relaxation parameter.

The main contribution of this paper is a Lyapunov analysis establishing that the iterates $((x^k, y^k))_{k\in\mathbb{N}_0}$ produced by the Chambolle–Pock method (3.9) converge weakly to a KKT point (3.8), i.e., a solution to the saddle-point problem (3.7), if $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$, and that the latter bound is unimprovable. Figure 3.2 compares this bound to other bounds in the literature.

Paper IV

M. Upadhyaya, P. Latafat, and P. Giselsson. "A Lyapunov analysis of Korpelevich's extragradient method with fast and flexible extensions" (2025)
Status: In submission

This paper considers Korpelevich's extragradient method [72], i.e., the method

$$\begin{aligned} \bar{z}^k &= \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(z^k)\right), \\ z^{k+1} &= \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(\bar{z}^k)\right), \end{aligned} \tag{3.10}$$

for some step size $\gamma \in (0, 1/L_F)$, which solves inclusion problems

$$\text{find } z \in \mathcal{H} \ \text{ such that } \ 0 \in F(z) + \partial g(z), \tag{3.11}$$

where $F : \mathcal{H} \to \mathcal{H}$ is monotone and L_F-Lipschitz continuous for some $L_F \in \mathbb{R}_{++}$, $g : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is a proper, convex, and lower semicontinuous function, and $\mathcal{H}$ is a real Hilbert space.

The paper's main contributions are as follows:

(i) We propose the Lyapunov function

$$\begin{aligned} \mathcal{V}(z^k, \bar{z}^k, z^{k+1}) = 2\gamma^{-1}\langle z^k - z^{k+1}, F(z^k) - F(\bar{z}^k)\rangle + \gamma^{-2}\|z^{k+1} - \bar{z}^k\|^2 \\ + \gamma^{-2}\|z^k - z^{k+1}\|^2 \end{aligned}$$

for the extragradient method (3.10), which is a nonnegative optimality measure that fulfills the descent inequality

$$\mathcal{V}_{k+1} \leq \mathcal{V}_k - (1 - \gamma^2 L_F^2)\gamma^{-2}\|z^{k+1} - \bar{z}^k\|^2, \tag{3.12}$$

where $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$, and converges to zero with a rate $o(1/k)$. Using $\mathcal{V}_k$ enables us to recover and, in some cases, extend recent last-iterate convergence-rate results for the extragradient method.

(ii) Building on the analysis above, we propose flexible extensions that combine extragradient steps (3.10) with user-specified directions (e.g., based on quasi-second-order information), guided by a line-search procedure derived from (3.12). We show that the extensions retain global convergence under practical assumptions and can achieve superlinear rates when directions are chosen appropriately.

3.2 Author's contributions

Paper I

The author contributed to the conceptualization, theoretical developments, proofs, writing and revising the paper, and numerical experiments.

Paper II

The author contributed to the conceptualization, theoretical developments, proofs, software architectural choices, writing the paper, and numerical experiments.

Paper III

The author contributed to the conceptualization, theoretical developments, proofs, and writing the paper.

Paper IV

The author contributed to the conceptualization, theoretical developments, proofs, writing the paper, and numerical experiments.

4

Concluding remarks

This chapter provides concluding remarks and discusses possible future work.

(i) This thesis's main focus has been the development of computer-aided tools for constructive Lyapunov analyses for first-order methods. However, as is often the case with Lyapunov analyses, the reason why a systematic search for a Lyapunov analysis fails is unclear. It is possible that the class of considered Lyapunov analyses is not rich enough, or that the method actually does not converge. Recent works [39, 40] have built tools that search for cyclic trajectories; once such a trajectory is found, there is no need to further look for a Lyapunov analysis, as the method is guaranteed not to converge on a specific problem of the problem class.

(ii) Ideally, one would like to move beyond relying on numerical results as an intermediate step and instead pursue closed-form Lyapunov analyses directly. In our context, this amounts to finding closed-form solutions to parametric semidefinite programs, which is an active area of research [4, 55]. It remains an open question how far these techniques can be pushed in the context of Lyapunov analyses.

(iii) An interesting recent development is the use of so-called long-horizon Lyapunov analyses, which, e.g., have been used to analyze and design gradient methods with cyclical step-size schedules that provide acceleration without the use of momentum or any other techniques [1, 2, 41, 42]. It would be interesting to further develop these ideas for other methods and settings.

(iv) The standard PEP methodology has been used to design complexity (or worst-case) optimal first-order methods under various settings. A very nice observation is that many of these methods have simple and tight Lyapunov analyses. The most notable example is probably the optimized gradient method (OGM), first considered in [30] and formally obtained in [46], for minimizing smooth and convex functions. The work [29] showed that OGM is worst-case optimal, and the work [26] showed that there exists a tight Lyapunov analysis for OGM.

(v) Methods such as mirror descent [11, 57], mirror-prox [56], and the Bregman proximal gradient method [5] have been introduced to exploit the geometry of problems and extend applicability. A first attempt to extend the PEP methodology to include these types of methods was made in [28]. It seems worthwhile to explore this idea further, both within the Lyapunov framework and more broadly.

References

[1] J. M. Altschuler and P. A. Parrilo. "Acceleration by stepsize hedging: Multi-step descent and the silver stepsize schedule". *Journal of the ACM* **72**:2 (2025), pp. 1–38. DOI: 10.1145/3708502.

[2] J. M. Altschuler and P. A. Parrilo. "Acceleration by stepsize hedging: Silver stepsize schedule for smooth convex optimization". *Mathematical Programming* (2024). DOI: 10.1007/s10107-024-02164-2.

[3] S. Banert, M. Upadhyaya, and P. Giselsson. "The Chambolle–Pock method converges weakly with $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$" (2023). arXiv: 2309.03998v1 [math.OC].

[4] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry*. Springer Berlin Heidelberg, 2006. ISBN: 9783540330998. DOI: 10.1007/3-540-33099-2.

[5] H. H. Bauschke, J. Bolte, and M. Teboulle. "A descent lemma beyond Lipschitz gradient continuity: First-order methods revisited and applications". *Mathematics of Operations Research* **42**:2 (2017), pp. 330–348. DOI: 10.1287/moor.2016.0817.

[6] H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz (Eds.). *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer New York, 2011. ISBN: 9781441995698. DOI: 10.1007/978-1-4419-9569-8.

[7] H. H. Bauschke, R. S. Burachik, and D. R. Luke (Eds.). *Splitting Algorithms, Modern Operator Theory, and Applications*. Springer International Publishing, 2019. ISBN: 9783030259396. DOI: 10.1007/978-3-030-25939-6.

[8] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer International Publishing, 2017. ISBN: 9783319483115. DOI: 10.1007/978-3-319-48311-5.

[9] H. H. Bauschke and W. M. Moursi. *An Introduction to Convexity, Optimization, and Algorithms*. Society for Industrial and Applied Mathematics, 2023. ISBN: 9781611977806. DOI: 10.1137/1.9781611977806.

[10] A. Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, 2017. ISBN: 9781611974997. DOI: 10.1137/1.9781611974997.

[11] A. Beck and M. Teboulle. "Mirror descent and nonlinear projected subgradient methods for convex optimization". *Operations Research Letters* **31**:3 (2003), pp. 167–175. DOI: 10.1016/s0167-6377(02)00231-6.

[12] D. P. Bertsekas. *Convex Optimization Algorithms*. Athena Scientific, Belmont, Massachusetts, 2015. ISBN: 9781886529281.

[13] J. Borwein and A. Lewis. *Convex Analysis and Nonlinear Optimization*. Springer New York, 2006. ISBN: 9780387312569. DOI: 10.1007/978-0-387-31256-9.

[14] J. M. Borwein and J. D. Vanderwerff. *Convex Functions. Constructions, Characterizations and Counterexamples*. Encyclopedia of Mathematics and its Applications 109. Cambridge University Press, Cambridge, UK, 2010. ISBN: 9781139811798.

[15] R. I. Bot, E. R. Csetnek, and C. Hendrich. "Recent developments on primal-dual splitting methods with applications to convex minimization". In: *Mathematics Without Boundaries*. Springer New York, 2014, pp. 57–99. ISBN: 9781493911240. DOI: 10.1007/978-1-4939-1124-0_3.

[16] S. Boyd and L. Vandenberghe. *Convex Optimization*. Version 29. First published 2004. Cambridge University Press, Cambridge, 2023.

[17] L. M. Briceño-Arias and P. L. Combettes. "Monotone operator methods for Nash equilibria in non-potential games". In: *Computational and Analytical Mathematics*. Springer New York, 2013, pp. 143–159. ISBN: 9781461476214. DOI: 10.1007/978-1-4614-7621-4_9.

[18] M. N. Bùi and P. L. Combettes. "Analysis and numerical solution of a modular convex Nash equilibrium problem" (2021). arXiv: 2111.01781v1 [math.OC].

[19] G. C. Calafiore and L. E. Ghaoui. *Optimization Models*. 1st ed. Cambridge University Press, West Nyack, 2014. ISBN: 9781107279667.

[20] A. Chambolle and T. Pock. "A first-order primal-dual algorithm for convex problems with applications to imaging". *Journal of Mathematical Imaging and Vision* **40**:1 (2011), pp. 120–145. DOI: 10.1007/s10851-010-0251-1.

[21] A. Chambolle and T. Pock. "An introduction to continuous optimization for imaging". *Acta Numerica* **25** (2016), pp. 161–319. DOI: 10.1017/s096249291600009x.

[22] P. L. Combettes and S. A. Hirstoaga. "Equilibrium programming in Hilbert spaces". *Journal of Nonlinear and Convex Analysis* **6** (2005), pp. 117–136.

[23] P. L. Combettes and J.-C. Pesquet. "Fixed point strategies in data science". *IEEE Transactions on Signal Processing* **69** (2021), pp. 3878–3905. DOI: 10.1109/tsp.2021.3069677.

[24] P. L. Combettes and J.-C. Pesquet. "Proximal splitting methods in signal processing". In: *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer New York, 2011, pp. 185–212. ISBN: 9781441995698. DOI: 10.1007/978-1-4419-9569-8_10.

[25] L. Condat, D. Kitahara, A. Contreras, and A. Hirabayashi. "Proximal splitting algorithms for convex optimization: A tour of recent advances, with new twists". *SIAM Review* **65**:2 (2023), pp. 375–435. DOI: 10.1137/20m1379344.

[26] A. d'Aspremont, D. Scieur, and A. Taylor. "Acceleration methods". *Foundations and Trends in Optimization* **5**:1–2 (2021), pp. 1–245. DOI: 10.1561/2400000036.

[27] J. Douglas and H. H. Rachford. "On the numerical solution of heat conduction problems in two and three space variables". *Transactions of the American Mathematical Society* **82**:2 (1956), pp. 421–439. DOI: 10.1090/s0002-9947-1956-0084194-4.

[28] R.-A. Dragomir, A. B. Taylor, A. d'Aspremont, and J. Bolte. "Optimal complexity and certification of Bregman first-order methods". *Mathematical Programming* **194**:1–2 (2021), pp. 41–83. DOI: 10.1007/s10107-021-01618-1.

[29] Y. Drori. "The exact information-based complexity of smooth convex minimization". *Journal of Complexity* **39** (2017), pp. 1–16. DOI: 10.1016/j.jco.2016.11.001.

[30] Y. Drori and M. Teboulle. "Performance of first-order methods for smooth convex minimization: A novel approach". *Mathematical Programming* **145**:1–2 (2014), pp. 451–482. DOI: 10.1007/s10107-013-0653-0.

[31] P. Dvurechensky, S. Shtern, and M. Staudigl. "First-order methods for convex optimization". *EURO Journal on Computational Optimization* **9** (2021). DOI: 10.1016/j.ejco.2021.100015.

[32] J. Eckstein and D. P. Bertsekas. "On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators". *Mathematical Programming* **55**:1–3 (1992), pp. 293–318. DOI: 10.1007/bf01581204.

[33] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems*. Society for Industrial and Applied Mathematics, 1999. ISBN: 9781611971088. DOI: 10.1137/1.9781611971088.

[34] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Vol. I. Springer New York, 2004. ISBN: 9780387955803. DOI: 10.1007/b97543.

[35] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Vol. II. Springer New York, 2004. ISBN: 9780387955810. DOI: 10.1007/b97544.

[36] M. Fazlyab, A. Ribeiro, M. Morari, and V. M. Preciado. "Analysis of optimization algorithms via integral quadratic constraints: Nonstrongly convex problems". *SIAM Journal on Optimization* **28**:3 (2018), pp. 2654–2689. DOI: 10.1137/17m1136845.

[37] P. Giselsson and A. Rantzer (Eds.). *Large-Scale and Distributed Optimization*. Lecture Notes in Mathematics. Springer Cham, 2018. ISBN: 9783319974781. DOI: 10.1007/978-3-319-97478-1.

[38] R. Glowinski, S. J. Osher, and W. Yin (Eds.). *Splitting Methods in Communication, Imaging, Science, and Engineering*. Springer International Publishing, 2016. ISBN: 9783319415895. DOI: 10.1007/978-3-319-41589-5.

[39] B. Goujaud, A. Dieuleveut, and A. Taylor. "Counter-examples in first-order optimization: A constructive approach". *IEEE Control Systems Letters* **7** (2023), pp. 2485–2490. DOI: 10.1109/lcsys.2023.3286277.

[40] B. Goujaud, A. Taylor, and A. Dieuleveut. "Provable non-accelerations of the heavy-ball method" (2023). arXiv: 2307.11291v1 [math.OC].

[41] B. Grimmer. "Provably faster gradient descent via long steps". *SIAM Journal on Optimization* **34**:3 (2024), pp. 2588–2608. DOI: 10.1137/23m1588408.

[42] B. Grimmer, K. Shu, and A. L. Wang. "Accelerated objective gap and gradient norm convergence for gradient descent via long steps". *INFORMS Journal on Optimization* **7**:2 (2025), pp. 156–169. DOI: 10.1287/ijoo.2024.0057.

[43] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I: Fundamentals*. Springer Berlin Heidelberg, 1993. ISBN: 9783662027967. DOI: 10.1007/978-3-662-02796-7.

[44] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods*. Springer Berlin Heidelberg, 1993. ISBN: 9783662064092. DOI: 10.1007/978-3-662-06409-2.

[45] L. Hörmander. *Notions of Convexity*. Modern Birkhäuser Classics. Birkhäuser Boston, Boston, MA, 2007. ISBN: 9780817645854. DOI: 10.1007/978-0-8176-4585-4.

[46] D. Kim and J. A. Fessler. "Optimized first-order methods for smooth convex minimization". *Mathematical Programming* **159**:1–2 (2016), pp. 81–107. DOI: 10.1007/s10107-015-0949-3.

[47] N. Komodakis and J.-C. Pesquet. "Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems". *IEEE Signal Processing Magazine* **32**:6 (2015), pp. 31–54. DOI: 10.1109/msp.2014.2377273.

[48] G. M. Korpelevich. "The extragradient method for finding saddle points and other problems". *Matecon* **12** (1976), pp. 747–756.

[49] L. Lessard. "The analysis of optimization algorithms: A dissipativity approach". *IEEE Control Systems* **42**:3 (2022), pp. 58–72. DOI: 10.1109/mcs.2022.3157115.

[50] L. Lessard, B. Recht, and A. Packard. "Analysis and design of optimization algorithms via integral quadratic constraints". *SIAM Journal on Optimization* **26**:1 (2016), pp. 57–95. DOI: 10.1137/15m1009597.

[51] P. L. Lions and B. Mercier. "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6 (1979), pp. 964–979. DOI: 10.1137/0716071.

[52] A. Megretski and A. Rantzer. "System analysis via integral quadratic constraints". *IEEE Transactions on Automatic Control* **42**:6 (1997), pp. 819–830. DOI: 10.1109/9.587335.

[53] B. S. Mordukhovich and N. Mau Nam. *Convex Analysis and Beyond: Volume I: Basic Theory*. Springer International Publishing, 2022. ISBN: 9783030947859. DOI: 10.1007/978-3-030-94785-9.

[54] C. Moucer, A. Taylor, and F. Bach. "A systematic approach to Lyapunov analyses of continuous-time models in convex optimization". *SIAM Journal on Optimization* **33**:3 (2023), pp. 1558–1586. DOI: 10.1137/22m1498486.

[55] S. Naldi, M. S. E. Din, A. Taylor, and W. Wang. "Solving generic parametric linear matrix inequalities" (2025). arXiv: 2503.01487v1 [cs.SC].

[56] A. Nemirovski. "Prox-method with rate of convergence $\mathcal{O}(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems". *SIAM Journal on Optimization* **15**:1 (2004), pp. 229–251. DOI: 10.1137/s1052623403425629.

[57] A. S. Nemirovskij and D. B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. E. R. Dawson (Ed.). A Wiley-Interscience publication. Wiley, Chichester, 1983. 388 pp. ISBN: 0471103454.

[58] Y. Nesterov. *Lectures on Convex Optimization*. Springer International Publishing, 2018. ISBN: 9783319915784. DOI: 10.1007/978-3-319-91578-4.

[59] C. P. Niculescu and L.-E. Persson. *Convex Functions and Their Applications: A Contemporary Approach*. Springer Nature Switzerland, 2025. ISBN: 9783031719677. DOI: 10.1007/978-3-031-71967-7.

[60] J. Peypouquet. *Convex Optimization in Normed Spaces: Theory, Methods and Examples*. Springer International Publishing, 2015. ISBN: 9783319137100. DOI: 10.1007/978-3-319-13710-0.

[61] R. R. Phelps. *Convex Functions, Monotone Operators and Differentiability*. Springer Berlin Heidelberg, 1989. ISBN: 9783662215692. DOI: 10.1007/978-3-662-21569-2.

[62] R. T. Rockafellar. *Convex Analysis*. Princeton Landmarks in Mathematics and Physics. Princeton University Press, Princeton, 1970. ISBN: 9780691015866.

[63] E. K. Ryu, A. B. Taylor, C. Bergeling, and P. Giselsson. "Operator splitting performance estimation: Tight contraction factors and optimal parameter selection". *SIAM Journal on Optimization* **30**:3 (2020), pp. 2251–2271. DOI: 10.1137/19m1304854.

[64] E. K. Ryu and W. Yin. *Large-Scale Convex Optimization. Algorithms & Analyses via Monotone Operators*. Cambridge University Press, Cambridge, United Kingdom, 2022. ISBN: 9781009160865.

[65] C. W. Scherer, C. Ebenbauer, and T. Holicki. "Optimization algorithm synthesis based on integral quadratic constraints: A tutorial". In: *Conference on Decision and Control (CDC)*. 2023. DOI: 10.1109/cdc49753.2023.10384198.

[66] A. Taylor and F. Bach. "Stochastic first-order methods: Non-asymptotic and computer-aided analyses via potential functions". In: *Conference on Learning Theory (COLT)*. 2019.

[67] A. Taylor, B. Van Scoy, and L. Lessard. "Lyapunov functions for first-order methods: Tight automated convergence guarantees". In: *International Conference on Machine Learning (ICML)*. 2018.

[68] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Exact worst-case performance of first-order methods for composite convex optimization". *SIAM Journal on Optimization* **27**:3 (2017), pp. 1283–1313. DOI: 10.1137/16m108104x.

[69] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Smooth strongly convex interpolation and exact worst-case performance of first-order methods". *Mathematical Programming* **161**:1–2 (2016), pp. 307–345. DOI: 10.1007/s10107-016-1009-3.

[70] M. Teboulle. "A simplified view of first order methods for optimization". *Mathematical Programming* **170**:1 (2018), pp. 67–96. DOI: 10.1007/s10107-018-1284-2.

[71] M. Upadhyaya, S. Banert, A. B. Taylor, and P. Giselsson. "Automated tight Lyapunov analysis for first-order methods". *Mathematical Programming* **209** (2025), pp. 133–170. DOI: 10.1007/s10107-024-02061-8.

[72] M. Upadhyaya, P. Latafat, and P. Giselsson. "A Lyapunov analysis of Korpelevich's extragradient method with fast and flexible extensions" (2025). arXiv: 2502.00119v1 [math.OC].

[73] B. Van Scoy and L. Lessard. "The speed-robustness trade-off for first-order methods with additive gradient noise" (2024). arXiv: 2109.05059v2 [math.OC].

[74] M. Yan and Y. Li. "On the improved conditions for some primal-dual algorithms". *Journal of Scientific Computing* **99**:3 (2024). DOI: 10.1007/s10915-024-02537-x.

[75] C. Zalinescu. *Convex Analysis in General Vector Spaces*. World Scientific, 2002. ISBN: 9789812777096. DOI: 10.1142/5021.

Paper I

Automated tight Lyapunov analysis for first-order methods

Manu Upadhyaya Sebastian Banert Adrien B. Taylor Pontus Giselsson

Abstract

We present a methodology for establishing the existence of quadratic Lyapunov inequalities for a wide range of first-order methods used to solve convex optimization problems. In particular, we consider (i) classes of optimization problems of finite-sum form with (possibly strongly) convex and possibly smooth functional components, (ii) first-order methods that can be written as a linear system on state-space form in feedback interconnection with the subdifferentials of the functional components of the objective function, and (iii) quadratic Lyapunov inequalities that can be used to draw convergence conclusions. We present a necessary and sufficient condition for the existence of a quadratic Lyapunov inequality within a predefined class of Lyapunov inequalities, which amounts to solving a small-sized semidefinite program. We showcase our methodology on several first-order methods that fit the framework. Most notably, our methodology allows us to significantly extend the region of parameter choices that allow for duality gap convergence in the Chambolle–Pock method when the linear operator is the identity mapping.

Keywords. Performance estimation, convex optimization, first-order methods, quadratic constraints, Lyapunov functions, semidefinite programming

Originally published in M. Upadhyaya et al. "Automated tight Lyapunov analysis for first-order methods". *Mathematical Programming* **209** (2025), pp. 133–170. DOI: 10.1007/s10107-024-02061-8. Reprinted with permission.

1. Introduction

First-order methods are used to solve optimization problems and can be analyzed via Lyapunov inequalities. Such inequalities consist of a Lyapunov function that is nonincreasing from one iteration to the next and a residual function that quantifies a lower bound on the potential decrease. The traditional approach of establishing a Lyapunov inequality, which is typically done on a case-by-case basis, amounts to combining and rearranging algorithm update equations and inequalities that describe properties of the objective function. In this paper, we develop an automated methodology for finding Lyapunov inequalities that can be applied to a large class of first-order methods.

The methodology uses an algorithm representation that covers most first-order methods with fixed parameters. The structure of the algorithm representation is a linear system in state-space form in feedback interconnection with a nonlinearity, in our case the subdifferentials of the functional components of the objective function. Such representations are common in the automatic control literature [45] and have previously been used for algorithm analysis, e.g., in [24]. The algorithm representation is also closely connected to the operator splitting framework introduced in [28]. Different algorithms are obtained by instantiating the matrices that define the linear system. Some matrix choices lead to algorithms that cannot solve the optimization problem in general. A contribution of this paper is that we provide conditions on the matrices that are necessary and sufficient for the equivalence between solving an instance of the optimization problem and finding a fixed point of the algorithm.

Our methodology is based on a necessary and sufficient condition for the existence of a quadratic Lyapunov inequality within a predefined class of Lyapunov inequalities. At the core of the methodology is a necessary and sufficient condition, in terms of a semidefinite program, for the optimal value of a quadratic objective function to be nonpositive when optimized over all possible algorithm iterates, fixed points, subgradients, and function values and over the full function class under consideration. This result is applied to the three conditions that we use to define a quadratic Lyapunov inequality. The resulting semidefinite program is feasible if and only if such a quadratic Lyapunov inequality exists, and it provides associated Lyapunov functions and residual functions when feasible.

Other methodologies that analyze optimization algorithms using semidefinite programs are the performance estimation problem (PEP) methodology [15, 39] and the integral quadratic constraints (IQC) methodology [24]. The PEP methodology poses the problem of finding a worst-case function from a predefined class of functions for the algorithm under consideration as an optimization problem. This is then reformulated in a sequence of steps to arrive at a semidefinite program. The PEP methodology, first presented in [15], has been extended in a sequence of works that guarantee tightness in each step of the reformulation [38, 39], has been adapted as a tool for Lyapunov analysis [29, 36, 37], and extended to monotone inclusion problems [34]. The IQC methodology is based on integral quadratic constraints from the control literature [26], which has been adopted for automated convergence analysis of first-order methods under various settings [23, 24, 43]. The IQC methodology uses a simple algorithm representation but lacks tightness guarantees. We are inspired by the strengths of both methodologies; the worst-case analysis and tightness guarantees of PEP and the simple algorithm representation of IQC. Another work that is inspired by the PEP and IQC frameworks for tight Lyapunov function analysis is [37]. Our framework is more general as it can be applied to a wider range of algorithms, allowing, e.g., for proximal operators, and can be used to derive a broader range of convergence results.

The proposed methodology is applied in two ways. First, to find the smallest possible linear convergence rate via quadratic Lyapunov inequalities for the algorithm at hand.

This is done via a bisection search over the convergence rate $\rho \in [0,1[$. Second, to find the range of algorithm parameters for which the Lyapunov analysis can guarantee function value convergence or duality gap convergence. The algorithms we consider are the Douglas–Rachford method [13, 25], the (proximal) gradient method with heavy-ball momentum [17, 32], the three-operator splitting method by Davis and Yin [12], and the Chambolle–Pock method [7].

For the Douglas–Rachford method, we recover some of the known tight linear convergence rate results in [18, 19, 34]. For the gradient method with heavy-ball momentum, we improve, compared to [17], the linear convergence rate, and also extend the range of parameters that guarantee convergence in function-value suboptimality. We also show convergence of the duality gap for two proximal gradient methods with heavy-ball momentum. For the three-operator splitting method by Davis and Yin we provide linear convergence rate results that improve the ones found in [11, 31]. More strikingly, our methodology allows us to significantly enlarge the range of parameters that give duality gap convergence for the Chambolle–Pock method when the linear operator is assumed to be the identity operator. Traditional proofs such as in [7], allow for proximal operator step-size parameters $\tau_1, \tau_2 > 0$ to satisfy $\tau_1\tau_2 < 1$ and the coefficient θ for the linear combination of previous iterates to satisfy $\theta = 1$. Our analysis allows for a significantly wider range of parameter values, e.g., for $\theta = 1$ we allow for $\tau_1 = \tau_2 \in]0, 1.15]$, for $\theta = 0.35$ we allow for $\tau_1 = \tau_2 \in]0, 1.5]$, and for $\tau_1 = \tau_2 = 0.5$, we allow for $\theta \in [0.03, 7.5]$. We also show with the methodology that the extended range of parameters can lead to improved linear convergence rates over the traditional parameter choices.

The paper is organized as follows: in Section 2, we introduce the problem class and the algorithm representation. Section 3 discusses interpolation results and frames them in our setting. We define the notion of a quadratic Lyapunov inequality in Section 4. Section 5 contains the main result on the existence of a quadratic Lyapunov inequality. Section 6 contains numerical examples and Section 7 contains a proof of our core result. Section 8 contains the main conclusions of this work and discusses future work.

An implementation of the methodology and additional numerical examples can be found at:

https://github.com/ManuUpadhyaya/TightLyapunovAnalysis

1.1 Preliminaries

Let $\mathbb{N}_0$ denote the set of nonnegative integers, $\mathbb{Z}$ the set of integers, $[\![n, m]\!] = \{l \in \mathbb{Z} \mid n \le l \le m\}$ the set of integers inclusively between the integers n and m, $\mathbb{R}$ the set of real numbers, $\mathbb{R}_+$ the set of nonnegative real numbers, $\mathbb{R}_{++}$ the set of positive real numbers, $\mathbb{R}^n$ the set of all n-tuples of elements of $\mathbb{R}$, $\mathbb{R}^{m\times n}$ the set of real-valued matrices of size $m \times n$, if $M \in \mathbb{R}^{m\times n}$ then $[M]_{i,j}$ the i, j-th element of M, $\mathbb{S}^n$ the set of symmetric real-valued matrices of size $n \times n$, and $\mathbb{S}^n_+ \subseteq \mathbb{S}^n$ the set of positive semidefinite real-valued matrices of size $n \times n$. $\mathbf{1}$ denotes the column vector of all ones, where the size will be clear from the context.

Throughout this paper, $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ will denote a real Hilbert space. All norms $\|\cdot\|$ are canonical norms where the inner product will be clear from the context. We denote the identity mapping $x \mapsto x$ on $\mathcal{H}$ by Id. Given a function $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, the *effective domain* of f is the set $\operatorname{dom} f = \{x \in \mathcal{H} \mid f(x) < +\infty\}$. The function f is said to be *proper* if $\operatorname{dom} f \neq \emptyset$. The *subdifferential* of a proper function f is the set-valued operator $\partial f : \mathcal{H} \to 2^{\mathcal{H}}$ defined as the mapping $x \mapsto \{u \in \mathcal{H} \mid \forall y \in \mathcal{H},\ f(y) \ge f(x) + \langle u, y - x\rangle\}$.

Let $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ and $\sigma, \beta \in \mathbb{R}_+$. The function f is

(i) *convex* if $f((1-\lambda)x + \lambda y) \le (1-\lambda)f(x) + \lambda f(y)$ for each $x, y \in \mathcal{H}$ and $0 \le \lambda \le 1$,

(ii) σ-*strongly convex* if f is proper and $f - (\sigma/2)\|\cdot\|^2$ is convex, and

(iii) β-*smooth* if f is differentiable and $\|\nabla f(x) - \nabla f(y)\| \leq \beta\|x - y\|$ for each $x, y \in \mathcal{H}$.

Let $0 \leq \sigma < \beta \leq +\infty$. We let $\mathcal{F}_{\sigma,\beta}$ denote the class of all functions $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ that are

(i) β-smooth and σ-strongly convex if $\beta < +\infty$, and

(ii) lower semicontinuous and σ-strongly convex if $\beta = +\infty$.

Let $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ be proper, lower semicontinuous and convex, and let $\gamma > 0$. Then the *proximal operator* $\operatorname{prox}_{\gamma f} : \mathcal{H} \to \mathcal{H}$ is defined as the single-valued operator given by

$$\operatorname{prox}_{\gamma f}(x) = \operatorname*{argmin}_{z \in \mathcal{H}} \left(f(z) + \frac{1}{2\gamma}\|x - z\|^2 \right)$$

for each $x \in \mathcal{H}$. If $x, p \in \mathcal{H}$, then $p = \operatorname{prox}_{\gamma f}(x) \Leftrightarrow \gamma^{-1}(x - p) \in \partial f(p)$. Moreover, the *conjugate* of f, denoted $f^* : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, is the proper, lower semicontinuous and convex function given by $f^*(u) = \sup_{x \in \mathcal{H}} (\langle u, x \rangle - f(x))$ for each $u \in \mathcal{H}$. If $x, u \in \mathcal{H}$, then $u \in \partial f(x) \Leftrightarrow x \in \partial f^*(u)$ [4, Theorem 16.29].

Given any positive integer n, we let the inner product $\langle \cdot, \cdot \rangle$ on $\mathcal{H}^n$ be given by $\langle \boldsymbol{z}_1, \boldsymbol{z}_2 \rangle = \sum_{j=1}^{n} \langle z_1^{(j)}, z_2^{(j)} \rangle$ for each $\boldsymbol{z}_i = (z_i^{(1)}, \ldots, z_i^{(n)}) \in \mathcal{H}^n$ and $i \in [\![1, 2]\!]$. If $M \in \mathbb{R}^{m \times n}$, we define the tensor product $M \otimes \mathrm{Id}$ to be the mapping $(M \otimes \mathrm{Id}) : \mathcal{H}^n \to \mathcal{H}^m$ such that

$$(M \otimes \mathrm{Id})\boldsymbol{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z^{(j)}, \ldots, \sum_{j=1}^{n} [M]_{m,j} z^{(j)} \right)$$

for each $\boldsymbol{z} = (z^{(1)}, \ldots, z^{(n)}) \in \mathcal{H}^n$. The adjoint satisfies $(M \otimes \mathrm{Id})^* = M^\top \otimes \mathrm{Id}$. If $N \in \mathbb{R}^{n \times l}$, the composition rule $(M \otimes \mathrm{Id}) \circ (N \otimes \mathrm{Id}) = (MN) \otimes \mathrm{Id}$ holds. Moreover, if $M \in \mathbb{R}^{n \times n}$ is invertible, then $(M \otimes \mathrm{Id})^{-1} = M^{-1} \otimes \mathrm{Id}$ holds.

If we let $M_1 \in \mathbb{R}^{m \times n_1}$ and $M_2 \in \mathbb{R}^{m \times n_2}$, the relations above imply that $\langle (M_1 \otimes \mathrm{Id})\boldsymbol{z}_1, (M_2 \otimes \mathrm{Id})\boldsymbol{z}_2 \rangle = \langle \boldsymbol{z}_1, ((M_1^\top M_2) \otimes \mathrm{Id})\boldsymbol{z}_2 \rangle$ for each $\boldsymbol{z}_1 \in \mathcal{H}^{n_1}$ and $\boldsymbol{z}_2 \in \mathcal{H}^{n_2}$. We define the mapping[1] $\mathcal{Q} : \mathbb{S}^n \times \mathcal{H}^n \to \mathbb{R}$ by $\mathcal{Q}(M, \boldsymbol{z}) = \langle \boldsymbol{z}, (M \otimes \mathrm{Id})\boldsymbol{z} \rangle$ for each $M \in \mathbb{S}^n$ and $\boldsymbol{z} \in \mathcal{H}^n$. Note that, if $M \in \mathbb{S}^n$, $N \in \mathbb{R}^{n \times m}$ and $\boldsymbol{z} \in \mathcal{H}^m$, then $\mathcal{Q}(M, (N \otimes \mathrm{Id})\boldsymbol{z}) = \mathcal{Q}(N^\top M N, \boldsymbol{z})$.

2. Problem class and algorithm representation

In this section, we introduce the problem class and the algorithm representation. We provide conditions for when solving a problem is equivalent to finding a fixed point of an algorithm. We also provide conditions for when an algorithm can be implemented using scalar multiplications, vector additions, proximal operator evaluations, and gradient evaluations only. We conclude the section by listing a few examples of first-order methods that fit into the algorithm representation.

[1] We use the same symbol $\mathcal{Q}$ for the mapping independent of the dimension n, which will be clear from context.

2.1 Problem class

Let $0 \leq \sigma_i < \beta_i \leq +\infty$ for each $i \in [\![1, m]\!]$. Consider the convex optimization problem

$$\underset{y \in \mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^{m} f_i(y) \tag{2.1}$$

where $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$. Most first-order methods are limited to solving the related inclusion problem

$$\text{find } y \in \mathcal{H} \ \text{ such that } \ 0 \in \sum_{i=1}^{m} \partial f_i(y). \tag{2.2}$$

A solution to (2.2) is always a solution to (2.1) and the converse holds under some appropriate constraint qualification, e.g., see [6]. Moreover, it is reasonable to only consider problems such that the inclusion problem (2.2) is solvable, i.e., there exists at least one point $y \in \mathcal{H}$ such that $0 \in \sum_{i=1}^{m} \partial f_i(y)$. Thus, the problem class we consider is all solvable problems of the form (2.2) where $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1, m]\!]$. For examples of problems that can be modeled according to (2.1) or (2.2), we refer to the textbooks [5, 30].

For later convenience, we introduce the notation

$$\operatorname{zer}\left(\sum_{i=1}^{m} \partial f_i\right) = \left\{ y \in \mathcal{H} \;\middle|\; 0 \in \sum_{i=1}^{m} \partial f_i(y) \right\}.$$

That is, $\operatorname{zer}\left(\sum_{i=1}^{m} \partial f_i\right)$ is the set of zeros of the set-valued operator $\sum_{i=1}^{m} \partial f_i : \mathcal{H} \to 2^{\mathcal{H}} : y \mapsto \sum_{i=1}^{m} \partial f_i(y)$, which is the same as the set of solutions to (2.2).

2.2 Algorithm representation

We consider algorithms that solve (2.2) that can be represented as a discrete-time linear system in state-space form in feedback interconnection with the potentially nonlinear and set-valued subdifferentials that define the problem. In particular, let $\boldsymbol{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m$ and $\boldsymbol{\partial f} : \mathcal{H}^m \to 2^{\mathcal{H}^m}$ be mappings containing all functions and subdifferentials associated with (2.1) and (2.2) that satisfy

$$\boldsymbol{f}(\boldsymbol{y}) = (f_1(y^{(1)}), \ldots, f_m(y^{(m)})), \tag{2.3}$$

$$\boldsymbol{\partial f}(\boldsymbol{y}) = \prod_{i=1}^{m} \partial f_i(y^{(i)}) \tag{2.4}$$

for each $\boldsymbol{y} = (y^{(1)}, \ldots, y^{(m)}) \in \mathcal{H}^m$, respectively[2]. We consider algorithms that can be written as: pick an initial $\boldsymbol{x}_0 \in \mathcal{H}^n$ and let

$$\begin{aligned}
&\text{for } k = 0, 1, \ldots \\
&\left\lfloor \begin{aligned}
\boldsymbol{x}_{k+1} &= (A \otimes \mathrm{Id})\boldsymbol{x}_k + (B \otimes \mathrm{Id})\boldsymbol{u}_k, \\
\boldsymbol{y}_k &= (C \otimes \mathrm{Id})\boldsymbol{x}_k + (D \otimes \mathrm{Id})\boldsymbol{u}_k, \\
\boldsymbol{u}_k &\in \boldsymbol{\partial f}(\boldsymbol{y}_k), \\
\boldsymbol{F}_k &= \boldsymbol{f}(\boldsymbol{y}_k),
\end{aligned} \right.
\end{aligned} \tag{2.5}$$

[2] In this context, the symbol $\prod$ is used for Cartesian products.

where $\boldsymbol{x}_k \in \mathcal{H}^n$, $\boldsymbol{u}_k \in \mathcal{H}^m$, $\boldsymbol{y}_k \in \mathcal{H}^m$, and $\boldsymbol{F}_k \in \mathbb{R}^m$ are the algorithm variables and

$$A \in \mathbb{R}^{n\times n}, \qquad B \in \mathbb{R}^{n\times m}, \qquad C \in \mathbb{R}^{m\times n}, \qquad D \in \mathbb{R}^{m\times m}$$

are fixed matrices containing the parameters of the method at hand. For clarification, individual subgradients and function values are calculated as $u_k^{(i)} \in \partial f_i(y_k^{(i)})$ and $\boldsymbol{F}_k^{(i)} = f_i(y_k^{(i)})$, respectively, so that $\boldsymbol{u}_k = (u_k^{(1)}, \ldots, u_k^{(m)})$ and $\boldsymbol{F}_k = (f_1(y_k^{(1)}), \ldots, f_m(y_k^{(m)}))$. Moreover, representation (2.5) is a tool for analysis and does not necessarily indicate an efficient implementation, e.g., the function values are not used in the algorithm but are needed for the Lyapunov analysis. The structure in (2.5) of a linear system in feedback interconnection with a nonlinearity is common in the automatic control literature and has previously been proposed in [24, 44] as a model for algorithm analysis. It can represent a wide range of first-order methods as seen in Section 2.5.

Algorithm (2.5) searches for a *fixed point* $(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$ satisfying the fixed-point equations

$$\begin{aligned}
\boldsymbol{x}_\star &= (A \otimes \mathrm{Id})\boldsymbol{x}_\star + (B \otimes \mathrm{Id})\boldsymbol{u}_\star, \\
\boldsymbol{y}_\star &= (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star, \\
\boldsymbol{u}_\star &\in \boldsymbol{\partial f}(\boldsymbol{y}_\star), \\
\boldsymbol{F}_\star &= \boldsymbol{f}(\boldsymbol{y}_\star),
\end{aligned} \tag{2.6}$$

from which we want to recover a solution to (2.2). In particular, we want the problem of finding a fixed point of (2.5) to be equivalent to solving (2.2).

2.3 Solutions and fixed points

There are choices of the matrices A, B, C, and D such that it is not possible to extract a solution of (2.2) from fixed points of (2.5) in any practical way. To exclude such algorithms, we add the requirement that fixed points should satisfy

$$\boldsymbol{y}_\star = (y_\star, \ldots, y_\star), \qquad\qquad 0 = \sum_{i=1}^{m} u_\star^{(i)} \tag{2.7}$$

for some $y_\star \in \mathcal{H}$, where $\boldsymbol{u}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m)}) \in \mathcal{H}^m$. This implies that $y_\star$ solves (2.2) since the fixed-point equations (2.6) give that

$$0 = \sum_{i=1}^{m} u_\star^{(i)} \in \sum_{i=1}^{m} \partial f_i(y_\star).$$

We say that such fixed points are *fixed-point encodings* in line with the terminology in [33]. By defining the set of fixed points as

$$\begin{aligned}
&\Omega_{\text{fixed points}}(f_1, \ldots, f_m) \\
&\quad = \{(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m \mid (2.6) \text{ holds}\}
\end{aligned}$$

and the set fixed-point encodings as

$$\begin{aligned}
&\Omega_{\text{fixed-point encodings}}(f_1, \ldots, f_m) \\
&\quad = \{(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m \mid (2.6) \text{ and } (2.7) \text{ hold}\},
\end{aligned}$$

the requirement that all fixed points are fixed-point encodings can be written as $\Omega_{\text{fixed points}}(f_1, \ldots, f_m) = \Omega_{\text{fixed-point encodings}}(f_1, \ldots, f_m)$. Another requirement is that to each solution of (2.2), there exists a corresponding fixed point. These two requirements imply that solving (2.2) is equivalent to finding a fixed point of the algorithm. We say that such algorithms have the *fixed-point encoding property*.

Definition 2.1
We say that Algorithm (2.5) *has the* fixed-point encoding property *if*

$$y_\star \in \operatorname{zer}\left(\sum_{i=1}^{m} \partial f_i\right) \implies \exists(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{F}_\star) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathbb{R}^m \textit{ such that } (2.6) \textit{ and } (2.7) \textit{ hold} \tag{2.8}$$

and

$$\Omega_{\text{fixed points}}(f_1, \ldots, f_m) = \Omega_{\text{fixed-point encodings}}(f_1, \ldots, f_m) \tag{2.9}$$

for each $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i, \beta_i}$.

By appropriately restricting A, B, C, and D, we can exactly capture the class of algorithms with this property. For $m \geq 2$, let

$$N = \begin{bmatrix} I \\ -\mathbf{1}^\top \end{bmatrix} \in \mathbb{R}^{m\times(m-1)} \quad \text{and} \quad \hat{\boldsymbol{u}}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m-1)}). \tag{2.10}$$

The fixed-point encoding condition in (2.7) is then equivalent to $0 = (N^\top \otimes \mathrm{Id})\boldsymbol{y}_\star$ and $\boldsymbol{u}_\star = (N \otimes \mathrm{Id})\hat{\boldsymbol{u}}_\star$. In the case $m = 1$, the fixed-point encoding condition is simply $\boldsymbol{u}_\star = 0$. The matrix N enters in the restriction of A, B, C, and D to exactly capture the class of algorithms with the fixed-point encoding property.

Assumption 2.2
Suppose that

$$\operatorname{ran}\begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} \subseteq \operatorname{ran}\begin{bmatrix} I - A \\ -C \end{bmatrix} \tag{2.11}$$

with the interpretation that the block column containing N is removed when $m = 1$, and that

$$\operatorname{null}\begin{bmatrix} I - A & -B \end{bmatrix} \subseteq \operatorname{null}\begin{bmatrix} N^\top C & N^\top D \\ 0 & \mathbf{1}^\top \end{bmatrix}, \tag{2.12}$$

with the interpretation that the block row containing $N^\top$ is removed when $m = 1$.

Proposition 2.3
The following are equivalent:

(i) Assumption 2.2 holds.

(ii) Algorithm (2.5) *has the fixed-point encoding property.*

Proof. *(i)* $\Rightarrow$ *(ii):* Suppose that Assumption 2.2 holds. Let $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i,\beta_i}$.

First, we prove that (2.8) holds. Suppose that $y_\star \in \operatorname{zer}(\sum_{i=1}^{m} \partial f_i)$. This implies that there exists a $\boldsymbol{u}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m)}) \in \mathcal{H}^m$ such that

$$\boldsymbol{u}_\star \in \boldsymbol{\partial f}((\mathbf{1} \otimes \mathrm{Id})y_\star) \quad \text{and} \quad \sum_{i=1}^{m} u_\star^{(i)} = 0. \tag{2.13}$$

Note that the second part of (2.13) implies that

$$\boldsymbol{u}_\star = \begin{cases} (N \otimes \mathrm{Id})\hat{\boldsymbol{u}}_\star & \text{if } m > 1, \\ 0 & \text{if } m = 1, \end{cases}$$

for $\hat{\boldsymbol{u}}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m-1)}) \in \mathcal{H}^{m-1}$, where N is defined in (2.10). We will show that there exists an $\boldsymbol{x}_\star \in \mathcal{H}^n$ such that

$$\begin{aligned} \boldsymbol{x}_\star &= (A \otimes \mathrm{Id})\boldsymbol{x}_\star + (B \otimes \mathrm{Id})\boldsymbol{u}_\star, \\ (\mathbf{1} \otimes \mathrm{Id})y_\star &= (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star, \end{aligned} \tag{2.14}$$

i.e.,

$$(\boldsymbol{x}_\star, \boldsymbol{u}_\star, (\mathbf{1} \otimes \mathrm{Id})y_\star, \boldsymbol{f}((\mathbf{1} \otimes \mathrm{Id})y_\star)) \tag{2.15}$$

is a fixed-point encoding. This will prove the desired implication. Note that (2.14) is equivalent to

$$\left(\begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} \otimes \mathrm{Id}\right)(\hat{\boldsymbol{u}}_\star, y_\star) = \left(\begin{bmatrix} I - A \\ -C \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}_\star,$$

with the interpretation that $\hat{\boldsymbol{u}}_\star$ and the block column containing N is removed when $m = 1$. Moreover, (2.11) in Assumption 2.2 implies that there exists a matrix $U \in \mathbb{R}^{n \times m}$ such that

$$\begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} = \begin{bmatrix} I - A \\ -C \end{bmatrix} U,$$

i.e., each column of the matrix to the left in (2.11) can be written as linear combinations of the columns of the matrix to the right in (2.11). If we let $\boldsymbol{x}_\star = (U \otimes \mathrm{Id})(\hat{\boldsymbol{u}}_\star, y_\star)$, then we get

$$\begin{aligned} \left(\begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} \otimes \mathrm{Id}\right)(\hat{\boldsymbol{u}}_\star, y_\star) &= \left(\left(\begin{bmatrix} I - A \\ -C \end{bmatrix} U\right) \otimes \mathrm{Id}\right)(\hat{\boldsymbol{u}}_\star, y_\star) \\ &= \left(\begin{bmatrix} I - A \\ -C \end{bmatrix} \otimes \mathrm{Id}\right)(U \otimes \mathrm{Id})(\hat{\boldsymbol{u}}_\star, y_\star) \\ &= \left(\begin{bmatrix} I - A \\ -C \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}_\star, \end{aligned}$$

as desired.

Second, we prove that (2.9) holds. Note that the inclusion $\supseteq$ holds trivially. Therefore, we only need to prove the $\subseteq$ inclusion. Suppose that $(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \Omega_{\text{fixed points}}(f_1, \ldots, f_m)$. This implies that

$$\left(\begin{bmatrix} I - A & -B \end{bmatrix} \otimes \mathrm{Id}\right)(\boldsymbol{x}_\star, \boldsymbol{u}_\star) = 0.$$

However, (2.12) in Assumption 2.2 implies that

$$\left(\begin{bmatrix} N^\top C & N^\top D \\ 0 & \mathbf{1}^\top \end{bmatrix} \otimes \mathrm{Id}\right)(\boldsymbol{x}_\star, \boldsymbol{u}_\star) = 0,$$

with the interpretation that the block row containing $N^\top$ is removed when $m = 1$. In particular, note that this implies that

$$y_\star^{(1)} = \ldots = y_\star^{(m)} = y_\star, \quad \text{and} \quad \sum_{i=1}^{m} u_\star^{(i)} = 0,$$

where $\boldsymbol{u}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m)})$ and $\boldsymbol{y}_\star = (y_\star^{(1)}, \ldots, y_\star^{(m)})$, for some common value $y_\star \in \mathcal{H}$, since $\boldsymbol{y}_\star = (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star$ (and $(N^\top \otimes \mathrm{Id})\boldsymbol{y}_\star = (y_\star^{(1)} - y_\star^{(m)}, \ldots, y_\star^{(m-1)} - y_\star^{(m)})$ if $m > 1$). In particular, $(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \Omega_{\text{fixed-point encodings}}(f_1, \ldots, f_m)$. This proves the $\subseteq$ inclusion for (2.9).

(ii) ⇒ (i): We prove the contrapositive.

First, suppose that (2.11) does not hold, i.e., there exists $(y_\star, \hat{\boldsymbol{u}}_\star) \in \mathcal{H} \times \mathcal{H}^{m-1}$ such that

$$\left(\begin{bmatrix} I - A \\ -C \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}_\star = \left(\begin{bmatrix} BN & 0 \\ DN & -\mathbf{1} \end{bmatrix} \otimes \mathrm{Id}\right)(\hat{\boldsymbol{u}}_\star, y_\star) \tag{2.16}$$

does not hold for any $\boldsymbol{x}_\star \in \mathcal{H}^n$. Define $\boldsymbol{u}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m)}) \in \mathcal{H}^m$ such that

$$\boldsymbol{u}_\star = \begin{cases} (N \otimes \mathrm{Id})\hat{\boldsymbol{u}}_\star & \text{if } m > 1, \\ 0 & \text{if } m = 1, \end{cases}$$

and note that $\sum_{i=1}^{m} u_\star^{(i)} = 0$ holds by construction. Note that (2.16) then implies that

$$\begin{aligned} \boldsymbol{x}_\star &= (A \otimes \mathrm{Id})\boldsymbol{x}_\star + (B \otimes \mathrm{Id})\boldsymbol{u}_\star, \\ (\mathbf{1} \otimes \mathrm{Id})y_\star &= (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star \end{aligned}$$

does not hold for any $\boldsymbol{x}_\star \in \mathcal{H}^n$. Thus, if we can show that there exists $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i, \beta_i}$ such that $\boldsymbol{\partial f}((\mathbf{1} \otimes \mathrm{Id})y_\star) = \{\boldsymbol{u}_\star\}$ holds, then we are done since this shows that there exists $y_\star \in \operatorname{zer}\left(\sum_{i=1}^{m} \partial f_i\right)$ such that the implication in (2.8) fails. Let $(\delta_1, \ldots, \delta_m) \in \prod_{i=1}^{m} [\sigma_i, \beta_i]$ and $f_i : \mathcal{H} \to \mathbb{R}$ such that

$$f_i(y) = \frac{\delta_i}{2}\|y - y_\star\|^2 + \langle u_\star^{(i)}, y \rangle$$

for each $y \in \mathcal{H}$ and $i \in [\![1, m]\!]$. Then $(f_1, \ldots, f_m) \in \prod_{i=1}^{m} \mathcal{F}_{\sigma_i, \beta_i}$ is clear, and $\boldsymbol{\partial f}((\mathbf{1} \otimes \mathrm{Id})y_\star) = \{\boldsymbol{u}_\star\}$ holds since

$$\partial f_i(y_\star) = \{u_\star^{(i)}\}$$

for each $i \in [\![1, m]\!]$.

Second, suppose that (2.12) does not hold, i.e., there exists $(\boldsymbol{x}_\star, \boldsymbol{u}_\star) \in \mathcal{H}^n \times \mathcal{H}^m$ such that

$$\left(\begin{bmatrix} I - A & -B \end{bmatrix} \otimes \mathrm{Id}\right)(\boldsymbol{x}_\star, \boldsymbol{u}_\star) = 0,$$

but

$$\left(\begin{bmatrix} N^\top C & N^\top D \\ 0 & \mathbf{1}^\top \end{bmatrix} \otimes \mathrm{Id}\right)(\boldsymbol{x}_\star, \boldsymbol{u}_\star) \neq 0.$$

If we let $\boldsymbol{u}_\star = (u_\star^{(1)}, \ldots, u_\star^{(m)})$ and $\boldsymbol{y}_\star = (y_\star^{(1)}, \ldots, y_\star^{(m)}) = (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star$, then either or both of $y_\star^{(1)} = \ldots = y_\star^{(m)}$ and $\sum_{i=1}^m u_\star^{(i)} = 0$ fail. Thus, if we can show that there exists $(f_1, \ldots, f_m) \in \prod_{i=1}^m \mathcal{F}_{\sigma_i, \beta_i}$ such that (2.6) holds, then we are done since this shows that there exists $(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \Omega_{\text{fixed points}}(f_1, \ldots, f_m)$ such that $(\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \notin \Omega_{\text{fixed-point encodings}}(f_1, \ldots, f_m)$. Let $(\delta_1, \ldots, \delta_m) \in \prod_{i=1}^m [\sigma_i, \beta_i]$ and $f_i : \mathcal{H} \to \mathbb{R}$ such that

$$f_i(y) = \frac{\delta_i}{2}\|y - y_\star^{(i)}\|^2 + \langle u_\star^{(i)}, y\rangle$$

for each $y \in \mathcal{H}$ and $i \in [\![1, m]\!]$. Then $(f_1, \ldots, f_m) \in \prod_{i=1}^m \mathcal{F}_{\sigma_i, \beta_i}$, and (2.6) holds since

$$\partial f_i(y_\star^{(i)}) = \{u_\star^{(i)}\}$$

for each $i \in [\![1, m]\!]$. □

Remark 2.4

There exist many different choices of A, B, C, and D in (2.5) that can represent a given first-order method. The dimension m in $\boldsymbol{y} \in \mathcal{H}^m$ is fixed due to the number of functional components in problem (2.1), but the dimension n in $\boldsymbol{x} \in \mathcal{H}^n$ can vary among representations. In fact, there exists a minimal n such that a given first-order method can be represented as (2.5), leading to a minimal representation. *A necessary condition is that*

$$\operatorname{rank}\begin{bmatrix} I - A & -B \end{bmatrix} = n \qquad \textit{and} \qquad \operatorname{rank}\begin{bmatrix} I - A \\ -C \end{bmatrix} = n, \tag{2.17}$$

where both matrices appear in Assumption 2.2. If these do not hold, the system is not controllable *(also often called reachable) [9, Definitions 6.D1] or* observable *[9, Definitions 6.D2], respectively. This implies that the representation is not minimal [10, Theorem 25.2] and that it is possible, for instance via a Kalman decomposition [10, Section 25.2], to go from this non-minimal representation to a minimal representation that satisfies (2.17) and represents the same algorithm.*

Remark 2.5

Previously, [27, 35] derived necessary and sufficient conditions for the existence of a fixed point from which a solution can be extracted, using algorithm representations different from (2.5). Note that the existence of a fixed point from which a solution can be extracted differs from the concept of the fixed-point encoding property considered here.

2.4 Well-posedness

When analyzing existing algorithms, well-posedness is usually clear from the outset. However, when taking the more abstract point of view, as given by Algorithm (2.5), further discussion is warranted. We would like Algorithm (2.5) to be well-posed in the sense that it can be initiated at an arbitrary $\boldsymbol{x}_0 \in \mathcal{H}^n$ and produce an infinite sequence $\{(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{y}_k, \boldsymbol{F}_k)\}_{k=0}^{\infty}$ obeying the algorithm dynamics (2.5). This holds if for each $\boldsymbol{x} \in \mathcal{H}^n$, there exist $\boldsymbol{u} \in \mathcal{H}^m$ and $\boldsymbol{y} \in \mathcal{H}^m$ such that

$$\begin{aligned} \boldsymbol{y} &= (C \otimes \mathrm{Id})\boldsymbol{x} + (D \otimes \mathrm{Id})\boldsymbol{u} \\ \boldsymbol{u} &\in \boldsymbol{\partial f}(\boldsymbol{y}). \end{aligned} \tag{2.18}$$

In addition, if $\boldsymbol{u} \in \mathcal{H}^m$ and $\boldsymbol{y} \in \mathcal{H}^m$ are unique, then the generated sequence is unique.

If D has a lower-triangular structure, (2.18) can be solved using back-substitution. If $[D]_{i,i} \neq 0$, an implicit step is needed to find $y^{(i)}$ and $u^{(i)}$. If $[D]_{i,i} < 0$, this implicit step is a proximal evaluation, which implies uniqueness. If $[D]_{i,i} = 0$, $u^{(i)}$ is found via direct evaluation of $\partial f_i(y^{(i)})$ which is always unique if f_i is differentiable.

Assumption 2.6
Let

$$\begin{aligned} I_{\text{differentiable}} &= \{i \in [\![1, m]\!] : \beta_i < +\infty\}, \\ I_D &= \{i \in [\![1, m]\!] : [D]_{i,i} < 0\} \end{aligned}$$

and assume that $I_{\text{differentiable}} \cup I_D = [\![1, m]\!]$ *and* D *is lower triangular with nonpositive diagonal elements.*

The requirements in Assumption 2.6 give rise to causal algorithms that generate unique and infinite sequences that evaluate either a proximal operator or a gradient for each f_i and linearly combine results of previous evaluations to form inputs.

Proposition 2.7
Suppose that Assumption 2.6 holds. Then for any $(f_1, \ldots, f_m) \in \prod_{i=1}^m \mathcal{F}_{\sigma_i,\beta_i}$ *and* $\boldsymbol{x}_0 = (x_0^{(1)}, \ldots, x_0^{(n)}) \in \mathcal{H}^n$*, Algorithm (2.5) produces a unique sequence* $\{(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{y}_k, \boldsymbol{F}_k)\}_{k=0}^{\infty}$ *obeying the algorithm dynamics (2.5) and can be implemented as the following causal procedure:*

$$\begin{aligned}
&\textit{for } k = 0, 1, \ldots \\
&\quad \textit{for } i = 1, \ldots, m \\
&\qquad v_k^{(i)} = \sum_{j=1}^{n} [C]_{i,j} x_k^{(j)} + \sum_{j=1}^{i-1} [D]_{i,j} u_k^{(j)}, \\
&\qquad y_k^{(i)} = \begin{cases} \operatorname{prox}_{-[D]_{i,i} f_i}(v_k^{(i)}) & \textit{if } i \in I_D, \\ v_k^{(i)} & \textit{if } i \notin I_D, \end{cases} \\
&\qquad u_k^{(i)} = \begin{cases} (-[D]_{i,i})^{-1}(v_k^{(i)} - y_k^{(i)}) & \textit{if } i \in I_D, \\ \nabla f_i(y_k^{(i)}) & \textit{if } i \notin I_D, \end{cases} \\
&\qquad F_k^{(i)} = f_i(y_k^{(i)}), \\
&\quad \boldsymbol{x}_{k+1} = (x_{k+1}^{(1)}, \ldots, x_{k+1}^{(n)}) = (A \otimes \mathrm{Id})\boldsymbol{x}_k + (B \otimes \mathrm{Id})\boldsymbol{u}_k,
\end{aligned} \tag{2.19}$$

where $\boldsymbol{u}_k = (u_k^{(1)}, \ldots, u_k^{(m)})$*,* $\boldsymbol{y}_k = (y_k^{(1)}, \ldots, y_k^{(m)})$*,* $\boldsymbol{F}_k = (F_k^{(1)}, \ldots, F_k^{(m)})$ *and the empty sum is set equal to zero by convention.*

Proof. Let $(f_1, \ldots, f_m) \in \prod_{i=1}^m \mathcal{F}_{\sigma_i, \beta_i}$. Consider an arbitrary $k \in \mathbb{N}_0$ and pick any $\boldsymbol{x}_k = (x_k^{(1)}, \ldots, x_k^{(n)}) \in \mathcal{H}^n$. For $i \in [\![1, m]\!]$ in ascending order:

- $v_k^{(i)}$ in the inner loop in (2.19) is a linear combination of previously calculated/known quantities.
- If $i \in I_D$, then (2.5) and the structure of D in Assumption 2.6 give that

$$
\begin{aligned}
y_k^{(i)} \in v_k^{(i)} + [D]_{i,i} \partial f_i(y_k^{(i)}) \quad &\Leftrightarrow \quad \underbrace{(-[D]_{i,i})^{-1}(v_k^{(i)} - y_k^{(i)})}_{= u_k^{(i)}} \in \partial f_i(y_k^{(i)}) \\
&\Leftrightarrow \quad y_k^{(i)} = \operatorname{prox}_{-[D]_{i,i} f_i}(v_k^{(i)}),
\end{aligned}
$$

which is unique since the proximal operator is single-valued with full domain under our assumptions (recall that each f_i is assumed to be proper, lower semicontinuous, and convex).

- If $i \notin I_D$, then f_i is differentiable due to Assumption 2.6, and (2.5) gives that $y_k^{(i)} = v_k^{(i)}$ and $u_k^{(i)} = \nabla f_i(y_k^{(i)})$.

An inductive argument concludes the proof. □

The requirement that D is lower-triangular is for convenience. If there exists a permutation $\pi : [\![1, m]\!] \to [\![1, m]\!]$ with associated permutation matrix P_π such that $P_\pi D P_\pi^\top$ is lower-triangular, the resulting algorithm is equivalent to (2.19). Let $\bar{\boldsymbol{y}}_k = P_\pi \boldsymbol{y}_k$, $\bar{\boldsymbol{u}}_k = P_\pi \boldsymbol{u}_k$, and $\bar{\boldsymbol{f}} = \boldsymbol{f} \circ (P_\pi^\top \otimes \mathrm{Id})$ (that just reorders the inputs). Then $\boldsymbol{\partial} \bar{\boldsymbol{f}} = (P_\pi \otimes \mathrm{Id}) \circ \boldsymbol{\partial} \boldsymbol{f} \circ (P_\pi^\top \otimes \mathrm{Id})$ and the algorithm is equivalent to

$$
\begin{aligned}
\boldsymbol{x}_{k+1} &= (A \otimes \mathrm{Id}) \boldsymbol{x}_k + (B P_\pi^\top \otimes \mathrm{Id}) \bar{\boldsymbol{u}}_k, \\
\bar{\boldsymbol{y}}_k &= (P_\pi C \otimes \mathrm{Id}) \boldsymbol{x}_k + (P_\pi D P_\pi^\top \otimes \mathrm{Id}) \bar{\boldsymbol{u}}_k, \\
\bar{\boldsymbol{u}}_k &\in \boldsymbol{\partial} \bar{\boldsymbol{f}}(\bar{\boldsymbol{y}}_k),
\end{aligned}
$$

which can be implemented as in (2.19).

If no permutation matrix exists such that $P_\pi D P_\pi^\top$ is lower-triangular, then there exist $i < j$ with $i, j \in [\![1, m]\!]$ such that $[D]_{i,j} \neq 0$ and $[D]_{j,i} \neq 0$. This couples the ∂f_i and ∂f_j evaluations such that back-substitution fails and these updates cannot in general be done using only proximal operator or gradient evaluations of f_i and f_j individually.

Since the linear combinations decided by A, B, C, and D are arbitrary, all first-order methods that use fixed linear combinations of previously computed quantities and evaluate each individual subdifferentials only once per iteration and either via a proximal operator or gradient evaluation can be implemented as in (2.19), potentially after a permutation of variables. We provide a list of examples in Section 2.5 that all satisfy Assumption 2.6. They also satisfy Assumption 2.2, implying that solving (2.2) is equivalent to finding a fixed point of the algorithm, and the rank conditions in (2.17).

2.5 Examples

In this section, we provide examples of a few well-known algorithms that can be written as (2.5).

2.5.1 Douglas–Rachford method

Let $\gamma \in \mathbb{R}_{++}$, $\lambda \in \mathbb{R} \setminus \{0\}$ and $f_1, f_2 \in \mathcal{F}_{0,\infty}$. The Douglas–Rachford method [13, 16, 25] is given by

$$\begin{aligned} y_k^{(1)} &= \operatorname{prox}_{\gamma f_1}(x_k), \\ y_k^{(2)} &= \operatorname{prox}_{\gamma f_2}(2y_k^{(1)} - x_k), \\ x_{k+1} &= x_k + \lambda(y_k^{(2)} - y_k^{(1)}), \end{aligned}$$

which can equivalently be written as

$$\begin{aligned} \gamma^{-1}(x_k - y_k^{(1)}) &\in \partial f_1(y_k^{(1)}), \\ \gamma^{-1}((2y^{(1)} - x_k) - y_k^{(2)}) &\in \partial f_2(y_k^{(2)}), \\ x_{k+1} &= x_k + \lambda(y_k^{(2)} - y_k^{(1)}). \end{aligned}$$

By letting $\boldsymbol{x}_k = x_k$, $\boldsymbol{y}_k = (y_k^{(1)}, y_k^{(2)})$ and $\boldsymbol{u}_k = (\gamma^{-1}(x_k - y_k^{(1)}), \gamma^{-1}(2y_k^{(1)} - x_k - y_k^{(2)}))$, we get

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \left(\begin{bmatrix} -\gamma\lambda & -\gamma\lambda \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\ \boldsymbol{y}_k &= \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} -\gamma & 0 \\ -2\gamma & -\gamma \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\ \boldsymbol{u}_k &\in \boldsymbol{\partial f}(\boldsymbol{y}_k), \end{aligned}$$

where $\boldsymbol{\partial f}(\boldsymbol{y}) = \partial f_1(y^{(1)}) \times \partial f_2(y^{(2)})$ for each $\boldsymbol{y} = (y^{(1)}, y^{(2)}) \in \mathcal{H}^2$, which matches the form (2.5).

2.5.2 Gradient method with heavy-ball momentum

Let $\gamma, \beta_1 \in \mathbb{R}_{++}$, $\delta \in \mathbb{R}$ and $f_1 \in \mathcal{F}_{0,\beta_1}$. The gradient method with heavy-ball momentum is given by

$$x_{k+1} = x_k - \gamma \nabla f_1(x_k) + \delta(x_k - x_{k-1}).$$

By letting $\boldsymbol{x}_k = (x_k, x_{k-1})$, $\boldsymbol{y}_k = x_k$, and $\boldsymbol{u}_k = \nabla f_1(x_k)$, we get

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \left(\begin{bmatrix} 1+\delta & -\delta \\ 1 & 0 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} -\gamma \\ 0 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\ \boldsymbol{y}_k &= \left(\begin{bmatrix} 1 & 0 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k, \\ \boldsymbol{u}_k &\in \boldsymbol{\partial f}(\boldsymbol{y}_k), \end{aligned}$$

where $\boldsymbol{\partial f}(y) = \{\nabla f_1(y)\}$ for each $y \in \mathcal{H}$, which matches the form (2.5).

2.5.3 Proximal gradient method with heavy-ball momentum terms

Let $\gamma, \beta_1 \in \mathbb{R}_{++}$, $\delta_1, \delta_2 \in \mathbb{R}$, $f_1 \in \mathcal{F}_{0,\beta_1}$ and $f_2 \in \mathcal{F}_{0,\infty}$. A proximal gradient method with heavy-ball momentum terms is given by

$$x_{k+1} = \operatorname{prox}_{\gamma f_2}(x_k - \gamma \nabla f_1(x_k) + \delta_1(x_k - x_{k-1})) + \delta_2(x_k - x_{k-1}).$$

By letting $\boldsymbol{x}_k = (x_k, x_{k-1})$, $\boldsymbol{y}_k = (x_k, x_{k+1} - \delta_2(x_k - x_{k-1}))$, $\boldsymbol{u}_k = (\nabla f_1(x_k), \gamma^{-1}(x_k - \gamma\nabla f_1(x_k) + (\delta_1 + \delta_2)(x_k - x_{k-1}) - x_{k+1}))$, we get

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= \left(\begin{bmatrix} 1+\delta_1+\delta_2 & -\delta_1-\delta_2 \\ 1 & 0 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} -\gamma & -\gamma \\ 0 & 0 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k \\
\boldsymbol{y}_k &= \left(\begin{bmatrix} 1 & 0 \\ 1+\delta_1 & -\delta_1 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} 0 & 0 \\ -\gamma & -\gamma \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\
\boldsymbol{u}_k &\in \boldsymbol{\partial f}(\boldsymbol{y}_k),
\end{aligned}$$

where $\boldsymbol{\partial f}(\boldsymbol{y}) = \{\nabla f_1(y^{(1)})\} \times \partial f_2(y^{(2)})$ for each $\boldsymbol{y} = (y^{(1)}, y^{(2)}) \in \mathcal{H}^2$, which matches the form (2.5).

2.5.4 Davis–Yin three-operator splitting method

Let $\gamma, \lambda \in \mathbb{R}_{++}$, $0 \le \sigma_i < \beta_i \le +\infty$ and $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1,3]\!]$, and $\beta_2 < \infty$. The three-operator splitting method by Davis and Yin in [12] is given by

$$\begin{aligned}
x_k &= \mathrm{prox}_{\gamma f_1}(z_k), \\
z_{k+\frac{1}{2}} &= 2x_k - z_k - \gamma\nabla f_2(x_k), \\
z_{k+1} &= z_k + \lambda(\mathrm{prox}_{\gamma f_3}(z_{k+\frac{1}{2}}) - x_k).
\end{aligned}$$

By letting $\boldsymbol{x}_k = z_k$, $\boldsymbol{y}_k = (x_k, x_k, x_k + \lambda^{-1}(z_{k+1} - z_k))$ and $\boldsymbol{u}_k = (\gamma^{-1}(z_k - x_k), \gamma^{-1}(2x_k - z_k - z_{k+\frac{1}{2}}), \gamma^{-1}(z_{k+\frac{1}{2}} - x_k - \lambda^{-1}(z_{k+1} - z_k)))$, we get

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \left(\begin{bmatrix} -\gamma\lambda & -\gamma\lambda & -\gamma\lambda \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\
\boldsymbol{y}_k &= \left(\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} -\gamma & 0 & 0 \\ -\gamma & 0 & 0 \\ -2\gamma & -\gamma & -\gamma \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\
\boldsymbol{u}_k &\in \boldsymbol{\partial f}(\boldsymbol{y}_k),
\end{aligned}$$

where $\boldsymbol{\partial f}(\boldsymbol{y}) = \partial f_1(y^{(1)}) \times \{\nabla f_2(y^{(2)})\} \times \partial f_3(y^{(3)})$ for each $\boldsymbol{y} = (y^{(1)}, y^{(2)}, y^{(3)}) \in \mathcal{H}^3$, which matches the form (2.5).

2.5.5 Chambolle–Pock method

Let $\tau_1, \tau_2 \in \mathbb{R}_{++}$, $\theta \in \mathbb{R}$, $0 \le \sigma_i < \beta_i \le +\infty$ and $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ for each $i \in [\![1,2]\!]$. The method by Chambolle and Pock in [7, Algorithm 1] is given by

$$\begin{aligned}
x_{k+1} &= \mathrm{prox}_{\tau_1 f_1}(x_k - \tau_1 y_k), \\
y_{k+1} &= \mathrm{prox}_{\tau_2 f_2^*}(y_k + \tau_2(x_{k+1} + \theta(x_{k+1} - x_k))).
\end{aligned}$$

By letting $\boldsymbol{x}_k = (x_k, y_k)$, $\boldsymbol{y}_k = (x_{k+1}, \frac{1}{\tau_2}(y_k - y_{k+1}) + (1+\theta)x_{k+1} - \theta x_k)$, and $\boldsymbol{u}_k = (\frac{1}{\tau_1}(x_k - x_{k+1}) - y_k, y_{k+1})$, we get

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= \left(\begin{bmatrix} 1 & -\tau_1 \\ 0 & 0 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} -\tau_1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\
\boldsymbol{y}_k &= \left(\begin{bmatrix} 1 & -\tau_1 \\ 1 & \frac{1}{\tau_2} - \tau_1(1+\theta) \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{x}_k + \left(\begin{bmatrix} -\tau_1 & 0 \\ -\tau_1(1+\theta) & -\frac{1}{\tau_2} \end{bmatrix} \otimes \mathrm{Id}\right) \boldsymbol{u}_k, \\
\boldsymbol{u}_k &\in \boldsymbol{\partial f}(\boldsymbol{y}_k),
\end{aligned}$$

where $\boldsymbol{\partial f}(\boldsymbol{y}) = \partial f_1(y^{(1)}) \times \partial f_2(y^{(2)})$ for each $\boldsymbol{y} = (y^{(1)}, y^{(2)}) \in \mathcal{H}^2$, which matches the form (2.5).

3. Interpolation

Tightness of our methodology hinges critically on so-called *interpolation conditions* for function classes that have been developed in the PEP literature [38, 39]. The following theorem is proved in [39, Theorem 4].

Theorem 3.1
Let $0 \leq \sigma < \beta \leq +\infty$ and $\{(y_i, F_i, u_i)\}_{i\in\mathcal{I}}$ be a finite family of triplets in $\mathcal{H} \times \mathbb{R} \times \mathcal{H}$ indexed by $\mathcal{I}$. Then the following are equivalent:

(i) There exists $f \in \mathcal{F}_{\sigma,\beta}$ such that

$$f(y_i) = F_i \text{ and } u_i \in \partial f(y_i)$$

for each $i \in \mathcal{I}$.

(ii) It holds that

$$\begin{aligned} F_i \geq F_j + \langle u_j, y_i - y_j\rangle + \frac{\sigma}{2}\|y_i - y_j\|^2 \\ + \frac{1}{2(\beta-\sigma)}\|u_i - u_j - \sigma(y_i - y_j)\|^2 \end{aligned}$$

for each $i, j \in \mathcal{I}$, where $\frac{1}{2(\beta-\sigma)}$ is interpreted as 0 in the case $\beta = +\infty$.

Next, we adapt these interpolation conditions to our framework. In the following, we let $\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ denote the class of all mappings $\boldsymbol{f} : \mathcal{H}^m \to (\mathbb{R} \cup \{+\infty\})^m$ defined by (2.3) for every possible choice of $f_i \in \mathcal{F}_{\sigma_i,\beta_i}$ and $i \in [\![1, m]\!]$. Moreover, with each $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$, we associate the mapping $\boldsymbol{\partial f} : \mathcal{H}^m \to 2^{\mathcal{H}^m}$ defined by (2.4).

Corollary 3.2
Let

$$\boldsymbol{M}_l = \begin{cases} \dfrac{1}{2(\beta_l - \sigma_l)} \begin{bmatrix} \beta_l\sigma_l & -\sigma_l & \beta_l \\ -\sigma_l & 1 & -1 \\ \beta_l & -1 & 1 \end{bmatrix} \otimes \operatorname{diag}(e_l) & \text{if } \beta_l < \infty, \\ \dfrac{1}{2}\begin{bmatrix} \sigma_l & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \otimes \operatorname{diag}(e_l) & \text{if } \beta_l = +\infty, \end{cases} \tag{3.1}$$

$$\boldsymbol{a}_l = -e_l \tag{3.2}$$

for each $l \in [\![1, m]\!]$, where $\otimes$ denotes the Kronecker product and $\{e_i\}_{i=1}^m$ denotes the standard basis vectors of $\mathbb{R}^m$. Then for each finite family of triplets in $\mathcal{H}^m \times \mathbb{R}^m \times \mathcal{H}^m$ indexed by $\mathcal{I}$, $\{(\boldsymbol{y}_i, \boldsymbol{F}_i, \boldsymbol{u}_i)\}_{i\in\mathcal{I}}$, the following are equivalent:

(i) There exist $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ such that

$$\boldsymbol{f}(\boldsymbol{y}_i) = \boldsymbol{F}_i \text{ and } \boldsymbol{u}_i \in \boldsymbol{\partial f}(\boldsymbol{y}_i)$$

for each $i \in \mathcal{I}$.

(ii) It holds that

$$\boldsymbol{a}_l^\top(\boldsymbol{F}_i - \boldsymbol{F}_j) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y}_i - \boldsymbol{y}_j, \boldsymbol{u}_i, \boldsymbol{u}_j)) \leq 0$$

for each $i, j \in \mathcal{I}$ *and* $l \in [\![1, m]\!]$.

Moreover,

$$\mathcal{Q}(\boldsymbol{M}_l, (0, \boldsymbol{u}, \boldsymbol{u})) = 0$$

for each $\boldsymbol{u} \in \mathcal{H}^m$ *and* $l \in [\![1, m]\!]$.

4. Lyapunov inequalities

Convergence properties of many first-order methods can be analyzed via so-called *Lyapunov inequalities*. We consider Lyapunov inequalities of the form

$$V(\boldsymbol{\xi}_{k+1}, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}_k, \boldsymbol{\xi}_\star), \tag{4.1}$$

where $\rho \in [0, 1]$, $\boldsymbol{\xi}_k = (\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{y}_k, \boldsymbol{F}_k) \in \mathcal{S}$ contains all algorithm variables in iteration k, $\boldsymbol{\xi}_{k+1} = (\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, \boldsymbol{y}_{k+1}, \boldsymbol{F}_{k+1}) \in \mathcal{S}$ contains all algorithm variables in iteration $k+1$, $\boldsymbol{\xi}_\star = (\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star) \in \mathcal{S}$ is a fixed point, $V : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is called a *Lyapunov function*, $R : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is called a *residual function*, and $\mathcal{S} = \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathbb{R}^m$. Once such an inequality has been established, various convergence properties may be concluded depending on the properties of the functions V and R.

We consider quadratic ansatzes of the functions V and R given by

$$V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) = \mathcal{Q}(Q, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + q^\top(\boldsymbol{F} - \boldsymbol{F}_\star), \tag{4.2}$$

$$R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) = \mathcal{Q}(S, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + s^\top(\boldsymbol{F} - \boldsymbol{F}_\star) \tag{4.3}$$

for each $\boldsymbol{\xi}, \boldsymbol{\xi}_\star \in \mathcal{S}$, respectively, where $Q, S \in \mathbb{S}^{n+2m}$ and $q, s \in \mathbb{R}^m$ parameterize the functions. These are general ansatzes that allow for arbitrary linear combinations of scalar products between linear combinations of $x^{(i)} - x_\star^{(i)}$, $u^{(i)}$, and $u_\star^{(i)}$ and linear combinations of function value differences $f_i(y^{(i)}) - f_i(y_\star^{(i)})$.

To draw useful convergence conclusions from (4.1), we enforce nonnegative quadratic lower bounds on V and R given by

$$V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \geq \mathcal{Q}(P, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + p^\top(\boldsymbol{F} - \boldsymbol{F}_\star) \geq 0, \tag{4.4}$$

$$R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \geq \mathcal{Q}(T, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + t^\top(\boldsymbol{F} - \boldsymbol{F}_\star) \geq 0, \tag{4.5}$$

where $P, T \in \mathbb{S}^{n+2m}$ and $p, t \in \mathbb{R}^m$. We do not enforce these inequalities on all of $\mathcal{S} \times \mathcal{S}$ but only when the first argument is a so-called *algorithm-consistent* point and the second argument satisfies the fixed-point equations (2.6).

Definition 4.1
Consider Algorithm (2.5). *The point* $\boldsymbol{\xi} = (\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{y}, \boldsymbol{F}) \in \mathcal{S}$ *is called* algorithm-consistent for $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma}, \boldsymbol{\beta}}$ *if*

$$\begin{aligned} &\boldsymbol{y} = (C \otimes \mathrm{Id})\boldsymbol{x} + (D \otimes \mathrm{Id})\boldsymbol{u}, \\ &\boldsymbol{u} \in \boldsymbol{\partial f}(\boldsymbol{y}), \\ &\boldsymbol{F} = \boldsymbol{f}(\boldsymbol{y}). \end{aligned}$$

To restrict (4.4) and (4.5) on this subset of $\mathcal{S}\times\mathcal{S}$ gives a larger class of Lyapunov functions and residual functions compared to requiring them to hold on all of $\mathcal{S}\times\mathcal{S}$.

In the proposed methodology, the user specifies (P,p,T,t,ρ) and the methodology provides (Q,q,S,s) complying with (4.1), (4.4), and (4.5), if it exists. When such a (Q,q,S,s) exists, the choice of (P,p,T,t,ρ) decides which convergence properties the analysis implies.

(i) Suppose that $\rho\in[0,1[$. Then

$$\begin{aligned}0&\le \mathcal{Q}(P,(\boldsymbol{x}_k-\boldsymbol{x}_\star,\boldsymbol{u}_k,\boldsymbol{u}_\star))+p^\top(\boldsymbol{F}_k-\boldsymbol{F}_\star)\\&\le V(\boldsymbol{\xi}_k,\boldsymbol{\xi}_\star)\\&\le \rho^k V(\boldsymbol{\xi}_0,\boldsymbol{\xi}_\star)\to 0\end{aligned}$$

as $k\to\infty$. In particular,

$$\begin{gathered}\{\mathcal{Q}(P,(\boldsymbol{x}_k-\boldsymbol{x}_\star,\boldsymbol{u}_k,\boldsymbol{u}_\star))+p^\top(\boldsymbol{F}_k-\boldsymbol{F}_\star)\}_{k\in\mathbb{N}_0}\\ \text{converges } \rho\text{-linearly to zero.}\end{gathered}\tag{4.6}$$

(ii) Suppose that $\rho=1$. Then

$$\begin{aligned}\sum_{k=0}^{\infty}(\mathcal{Q}(T,(\boldsymbol{x}_k-\boldsymbol{x}_\star,\boldsymbol{u}_k,\boldsymbol{u}_\star))+t^\top(\boldsymbol{F}_k-\boldsymbol{F}_\star))&\le\sum_{k=0}^{\infty}R(\boldsymbol{\xi}_k,\boldsymbol{\xi}_\star)\\&\le V(\boldsymbol{\xi}_0,\boldsymbol{\xi}_\star),\end{aligned}\tag{4.7}$$

using a telescoping summation argument. In particular,

$$\begin{gathered}\{\mathcal{Q}(T,(\boldsymbol{x}_k-\boldsymbol{x}_\star,\boldsymbol{u}_k,\boldsymbol{u}_\star))+t^\top(\boldsymbol{F}_k-\boldsymbol{F}_\star)\}_{k\in\mathbb{N}_0}\\ \text{is summable and converges to zero.}\end{gathered}\tag{4.8}$$

Therefore, (P,p,T,t,ρ) needs to be chosen to extract interesting convergence results from the lower bounds. If $P=T=0$ and $p=t=0$, then V and R equal to the zero function gives a valid Lyapunov inequality (4.1) that complies with the lower bounds (4.4) and (4.5), but is of no interest. Useful choices of (P,p,T,t,ρ) that imply different specific convergence results are provided in Section 4.1.

The above requirements on the Lyapunov inequality, the Lyapunov function, and the residual function are formalized in Definition 4.3 after we define the notion of a *successor*.

Definition 4.2
Consider Algorithm (2.5). Given an algorithm-consistent point $\boldsymbol{\xi}$ for some $\boldsymbol{f}\in\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$, we define a successor *of $\boldsymbol{\xi}$ to be any point $\boldsymbol{\xi}_+=(\boldsymbol{x}_+,\boldsymbol{u}_+,\boldsymbol{y}_+,\boldsymbol{F}_+)\in\mathcal{S}$ such that*

$$\begin{aligned}\boldsymbol{x}_+&=(A\otimes\mathrm{Id})\boldsymbol{x}+(B\otimes\mathrm{Id})\boldsymbol{u},\\ \boldsymbol{y}_+&=(C\otimes\mathrm{Id})\boldsymbol{x}_++(D\otimes\mathrm{Id})\boldsymbol{u}_+,\\ \boldsymbol{u}_+&\in\boldsymbol{\partial}\boldsymbol{f}(\boldsymbol{y}_+),\\ \boldsymbol{F}_+&=\boldsymbol{f}(\boldsymbol{y}_+).\end{aligned}$$

Definition 4.3
*Let $V:\mathcal{S}\times\mathcal{S}\to\mathbb{R}$ as in (4.2), $R:\mathcal{S}\times\mathcal{S}\to\mathbb{R}$ as in (4.3), $P,T\in\mathbb{S}^{n+2m}$, $p,t\in\mathbb{R}^m$ and $\rho\in[0,1]$. We say that V and R satisfy the (P,p,T,t,ρ)-*quadratic Lyapunov inequality* for Algorithm (2.5) over the class $\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ if:*

C1 $V(\boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \leq \rho V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) - R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star)$ *for each* $\boldsymbol{\xi} \in \mathcal{S}$ *that is algorithm-consistent for* $\boldsymbol{f}$*, each successor* $\boldsymbol{\xi}_+ \in \mathcal{S}$ *of* $\boldsymbol{\xi}$*, each* $\boldsymbol{\xi}_\star \in \mathcal{S}$ *that satisfies* (2.6)*, and each* $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$*.*

C2 $V(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \geq \mathcal{Q}(P, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + p^\top(\boldsymbol{F} - \boldsymbol{F}_\star) \geq 0$ *for each* $\boldsymbol{\xi} \in \mathcal{S}$ *that is algorithm-consistent for* $\boldsymbol{f}$*, each* $\boldsymbol{\xi}_\star \in \mathcal{S}$ *that satisfies* (2.6)*, and each* $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$*.*

C3 $R(\boldsymbol{\xi}, \boldsymbol{\xi}_\star) \geq \mathcal{Q}(T, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + t^\top(\boldsymbol{F} - \boldsymbol{F}_\star) \geq 0$ *for each* $\boldsymbol{\xi} \in \mathcal{S}$ *that is algorithm-consistent for* $\boldsymbol{f}$*, each* $\boldsymbol{\xi}_\star \in \mathcal{S}$ *that satisfies* (2.6)*, and each* $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$*.*

The main result in Section 5 is a necessary and sufficient condition for the existence of a (P, p, T, t, ρ)-quadratic Lyapunov inequality expressed as a semidefinite feasibility problem over the Lyapunov function and residual function parameters (Q, q, S, s). This is done by providing a necessary and sufficient condition for each of C1, C2, and C3. Conditions C1, C2, and C3 can all be stated as the verification of a quadratic function $\Phi : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ to be nonpositive over the subset of $\mathcal{S} \times \mathcal{S}$ that includes algorithm-consistent points in the first argument and fixed points in the second. Restricting to this subset adds significant technical complication compared to verifying nonpositivity over the entirety of $\mathcal{S} \times \mathcal{S}$, but provides the added benefit of a more general Lyapunov analysis.

4.1 Lower bounds and convergence implications

In this section, we provide a few choices of (P, p, T, t, ρ) from which we can draw specific convergence results, under the assumption that there exists a Lyapunov function V and a residual function R that satisfy the (P, p, T, t, ρ)-quadratic Lyapunov inequality. Moreover, we assume that Assumptions 2.2 and 2.6 hold.

4.1.1 Linear convergence of the distance to the solution

Suppose that $\rho \in [0, 1[$ and

$$(P, p, T, t) = (\begin{bmatrix} C & D & -D \end{bmatrix}^\top e_i e_i^\top \begin{bmatrix} C & D & -D \end{bmatrix}, 0, 0, 0) \tag{4.9}$$

for some $i \in [\![1, m]\!]$, where $\{e_i\}_{i=1}^m$ denotes the standard basis vectors of $\mathbb{R}^m$. Then (4.6) implies that the *squared distance to the solution* $\{\|y_k^{(i)} - y_\star\|^2\}_{k \in \mathbb{N}_0}$ converges ρ-linearly to zero, where $y_\star$ is the solution to (2.2), since

$$\mathcal{Q}(P, (\boldsymbol{x}_k - \boldsymbol{x}_\star, \boldsymbol{u}_k, \boldsymbol{u}_\star)) + p^\top(\boldsymbol{F}_k - \boldsymbol{F}_\star) = \|y_k^{(i)} - y_\star\|^2 \geq 0.$$

Note that we exclude the case $\rho = 1$ since we can only guarantee that the squared distance to the solution remains bounded but not necessarily linearly convergent.

4.1.2 $\mathcal{O}(1/k)$ ergodic convergence

Suppose that $\rho = 1$.

Function-value suboptimality ($m = 1$). Suppose that $m = 1$ and

$$(P, p, T, t) = (0, 0, 0, 1). \tag{4.10}$$

Then (4.8) implies that the *function-value suboptimality* $\{f_1(y_k^{(1)}) - f_1(y_\star)\}_{k \in \mathbb{N}_0}$ converges to zero, since

$$\mathcal{Q}(T, (\boldsymbol{x}_k - \boldsymbol{x}_\star, \boldsymbol{u}_k, \boldsymbol{u}_\star)) + t^\top(\boldsymbol{F}_k - \boldsymbol{F}_\star) = f_1(y_k^{(1)}) - f_1(y_\star) \geq 0.$$

Moreover, (4.7) and Jensen's inequality imply that the *ergodic function-value suboptimality*

$$\left\{ f_1\left(\frac{1}{k+1}\sum_{j=0}^{k} y_j^{(1)}\right) - f_1(y_\star) \right\}_{k\in\mathbb{N}_0}$$

converges to zero with rate $\mathcal{O}(1/k)$ since

$$f_1\left(\frac{1}{k+1}\sum_{j=0}^{k} y_j^{(1)}\right) - f_1(y_\star) \leq \frac{V(\boldsymbol{\xi}_0, \boldsymbol{\xi}_\star)}{k+1}.$$

Duality gap. Suppose that

$$(P, p, T, t) = \left(0, 0, \begin{bmatrix} C & D & -D \\ 0 & 0 & I \end{bmatrix}^\top \begin{bmatrix} 0 & -\frac{1}{2}I \\ -\frac{1}{2}I & 0 \end{bmatrix} \begin{bmatrix} C & D & -D \\ 0 & 0 & I \end{bmatrix}, \mathbf{1}\right). \tag{4.11}$$

Then

$$\begin{aligned} &\mathcal{Q}(T, (\boldsymbol{x}_k - \boldsymbol{x}_\star, \boldsymbol{u}_k, \boldsymbol{u}_\star)) + t^\top(\boldsymbol{F}_k - \boldsymbol{F}_\star) \\ &\quad = \sum_{i=1}^{m} \left(f_i(y_k^{(i)}) - f_i(y_\star^{(i)}) - \langle u_\star^{(i)}, y_k^{(i)} - y_\star^{(i)}\rangle\right) \\ &\quad = \sum_{i=1}^{m} \left(f_i(y_k^{(i)}) - f_i(y_\star) - \langle u_\star^{(i)}, y_k^{(i)}\rangle\right) \geq 0, \end{aligned} \tag{4.12}$$

since $\sum_{i=1}^{m} u_\star^{(i)} = 0$ and $y_\star^{(1)} = \ldots = y_\star^{(m)} = y_\star$ (all fixed points are fixed-point encodings). The quantity in (4.12) is known as the *duality gap*. Note that if $m = 1$, the duality gap reduces to function-value suboptimality. The duality gap is in fact a natural generalization to the function-value suboptimality, which we motivate next (see also, e.g., [1, Section 3.1], [2, Theorem 3.9], and [8]). Problem (2.1) can equivalently be written as

$$\begin{aligned} \underset{(y^{(1)},\ldots,y^{(m)})\in\mathcal{H}^m}{\text{minimize}} \quad & \sum_{i=1}^{m} f_i(y^{(i)}) \\ \text{subject to} \quad & y^{(i)} = y^{(m)} \text{ for each } i \in [\![1, m]\!]. \end{aligned}$$

It has the *Lagrangian function* $\mathcal{L} : \mathcal{H}^m \times \mathcal{H}^m \to \mathbb{R}$ given by

$$\mathcal{L}(\boldsymbol{y}, \boldsymbol{u}) = \sum_{i=1}^{m} f_i(y^{(i)}) + \sum_{i=1}^{m} \langle u^{(i)}, y^{(m)} - y^{(i)}\rangle, \tag{4.13}$$

where $\boldsymbol{y} = (y^{(1)}, \ldots, y^{(m)}) \in \mathcal{H}^m$, and $\boldsymbol{u} = (u^{(1)}, \ldots, u^{(m)}) \in \mathcal{H}^m$ are the dual variables. The Lagrangian function satisfies

$$\mathcal{L}(\boldsymbol{y}_\star, \boldsymbol{u}) \leq \underbrace{\mathcal{L}(\boldsymbol{y}_\star, \boldsymbol{u}_\star)}_{=\sum_{i=1}^{m} f_i(y_\star)} \leq \mathcal{L}(\boldsymbol{y}, \boldsymbol{u}_\star)$$

for each $\boldsymbol{y}, \boldsymbol{u} \in \mathcal{H}^m$. In particular, $\mathcal{L}(\boldsymbol{y}_k, \boldsymbol{u}_\star) - \mathcal{L}(\boldsymbol{y}_\star, \boldsymbol{u}_k)$ is equal to (4.12) and (4.8) implies that the duality gap $\{\mathcal{L}(\boldsymbol{y}_k, \boldsymbol{u}_\star) - \mathcal{L}(\boldsymbol{y}_\star, \boldsymbol{u}_k)\}_{k\in\mathbb{N}_0}$ converges to zero. Moreover, (4.7) and Jensen's inequality imply that the *ergodic duality gap*

$$\left\{\mathcal{L}\left(\frac{1}{k+1}\sum_{j=0}^{k}\boldsymbol{y}_j, \boldsymbol{u}_\star\right) - \mathcal{L}\left(\boldsymbol{y}_\star, \frac{1}{k+1}\sum_{j=0}^{k}\boldsymbol{u}_j\right)\right\}_{k\in\mathbb{N}_0}$$

converges to zero with rate $\mathcal{O}(1/k)$.

5. Main result

This section provides a necessary and sufficient condition, in terms of the feasibility of a semidefinite program, for the existence of a quadratic Lyapunov inequality in the sense of Definition 4.3. First, we introduce some necessary notation. Recall $N \in \mathbb{R}^{m\times(m-1)}$ defined in (2.10) when $m > 1$. For all the matrices defined below, the interpretation is that the block column containing N is removed when $m = 1$. Let

$$\begin{aligned}
E_{\varnothing,+} &= \begin{bmatrix} C(I-A) & D-CB & -D & CBN \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \end{bmatrix}, \\
E_{\varnothing,\star} &= \begin{bmatrix} C & D & 0 & -DN \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & N \end{bmatrix}, \\
E_{+,\varnothing} &= \begin{bmatrix} C(A-I) & CB-D & D & -CBN \\ 0 & 0 & I & 0 \\ 0 & I & 0 & 0 \end{bmatrix}, \\
E_{\star,\varnothing} &= \begin{bmatrix} -C & -D & 0 & DN \\ 0 & 0 & 0 & N \\ 0 & I & 0 & 0 \end{bmatrix}, \\
E_{+,\star} &= \begin{bmatrix} CA & CB & D & -DN-CBN \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & N \end{bmatrix}, \\
E_{\star,+} &= \begin{bmatrix} -CA & -CB & -D & DN+CBN \\ 0 & 0 & 0 & N \\ 0 & 0 & I & 0 \end{bmatrix},
\end{aligned} \tag{5.1}$$

where $E_{i,j} \in \mathbb{R}^{3m\times(n+3m-1)}$ for each distinct $i, j \in \{\varnothing, +, \star\}$, and

$$\begin{aligned}
H_{\varnothing,+} &= \begin{bmatrix} I & -I \end{bmatrix}, & H_{+,\varnothing} &= \begin{bmatrix} -I & I \end{bmatrix}, & H_{\varnothing,\star} &= \begin{bmatrix} I & 0 \end{bmatrix}, \\
H_{\star,\varnothing} &= \begin{bmatrix} -I & 0 \end{bmatrix}, & H_{+,\star} &= \begin{bmatrix} 0 & I \end{bmatrix}, & H_{\star,+} &= \begin{bmatrix} 0 & -I \end{bmatrix},
\end{aligned} \tag{5.2}$$

where $H_{i,j} \in \mathbb{R}^{m\times 2m}$ for each distinct $i, j \in \{\varnothing, +, \star\}$. Define

$$\boldsymbol{M}_{(l,i,j)} = E_{i,j}^\top \boldsymbol{M}_l E_{i,j} \in \mathbb{S}^{n+3m-1} \quad \text{and} \quad \boldsymbol{a}_{(l,i,j)} = H_{i,j}^\top \boldsymbol{a}_l \in \mathbb{R}^{2m} \tag{5.3}$$

for each distinct $i,j \in \{\varnothing,+,\star\}$ and $l \in [\![1,m]\!]$, where the $\boldsymbol{M}_l$'s and $\boldsymbol{a}_l$'s are defined in (3.1) and (3.2), respectively. Moreover, let

$$\Sigma_\varnothing = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & N \end{bmatrix}, \quad \Sigma_+ = \begin{bmatrix} A & B & 0 & -BN \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & N \end{bmatrix}, \tag{5.4}$$

where $\Sigma_i \in \mathbb{R}^{(n+2m)\times(n+3m-1)}$ for each $i \in \{\varnothing,+\}$.

Theorem 5.1 (Main result)
Assume that Assumption 2.2 and Assumption 2.6 hold, let $\rho \in [0,1]$, and suppose that $P,T \in \mathbb{S}^{n+2m}$ and $p,t \in \mathbb{R}^m$ are such that

$$\begin{aligned} &\mathcal{Q}(P,(\boldsymbol{x}-\boldsymbol{x}_\star,\boldsymbol{u},\boldsymbol{u}_\star)) + p^\top(\boldsymbol{F}-\boldsymbol{F}_\star) \geq 0, \\ &\mathcal{Q}(T,(\boldsymbol{x}-\boldsymbol{x}_\star,\boldsymbol{u},\boldsymbol{u}_\star)) + t^\top(\boldsymbol{F}-\boldsymbol{F}_\star) \geq 0 \end{aligned}$$

for each $\boldsymbol{\xi} \in \mathcal{S}$ that is algorithm-consistent for $\boldsymbol{f}$, each $\boldsymbol{\xi}_\star \in \mathcal{S}$ that satisfies (2.6), and each $\boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$. Then a sufficient condition for there to exist a Lyapunov function $V : \mathcal{S}\times\mathcal{S} \to \mathbb{R}$ as in (4.2) and a residual function $R : \mathcal{S}\times\mathcal{S} \to \mathbb{R}$ as in (4.3) such that they satisfy the (P,p,T,t,ρ)-quadratic Lyapunov inequality for algorithm (2.5) over the class $\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ is that the following system of constraints

$$\text{C1}\begin{cases} \lambda^{\text{C1}}_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \{\varnothing,+,\star\}, \\ \Sigma_\varnothing^\top(\rho Q - S)\Sigma_\varnothing - \Sigma_+^\top Q \Sigma_+ + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,+,\star\} \\ i\neq j}} \lambda^{\text{C1}}_{(l,i,j)} \boldsymbol{M}_{(l,i,j)} \succeq 0, \\ \begin{bmatrix} \rho q - s \\ -q \end{bmatrix} + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,+,\star\} \\ i\neq j}} \lambda^{\text{C1}}_{(l,i,j)} \boldsymbol{a}_{(l,i,j)} = 0, \end{cases} \tag{5.5a}$$

$$\text{C2}\begin{cases} \lambda^{\text{C2}}_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \{\varnothing,\star\}, \\ \Sigma_\varnothing^\top(Q - P)\Sigma_\varnothing + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\} \\ i\neq j}} \lambda^{\text{C2}}_{(l,i,j)} \boldsymbol{M}_{(l,i,j)} \succeq 0, \\ \begin{bmatrix} q - p \\ 0 \end{bmatrix} + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\} \\ i\neq j}} \lambda^{\text{C2}}_{(l,i,j)} \boldsymbol{a}_{(l,i,j)} = 0, \end{cases} \tag{5.5b}$$

$$\text{C3}\begin{cases} \lambda^{\text{C3}}_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \{\varnothing,\star\}, \\ \Sigma_\varnothing^\top(S - T)\Sigma_\varnothing + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\} \\ i\neq j}} \lambda^{\text{C3}}_{(l,i,j)} \boldsymbol{M}_{(l,i,j)} \succeq 0, \\ \begin{bmatrix} s - t \\ 0 \end{bmatrix} + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j\in\{\varnothing,\star\} \\ i\neq j}} \lambda^{\text{C3}}_{(l,i,j)} \boldsymbol{a}_{(l,i,j)} = 0, \end{cases} \tag{5.5c}$$

$$Q, S \in \mathbb{S}^{n+2m}, \tag{5.5d}$$
$$q, s \in \mathbb{R}^{m}, \tag{5.5e}$$

is feasible for the scalars $\lambda^{\mathrm{C1}}_{(l,i,j)}$, $\lambda^{\mathrm{C2}}_{(l,i,j)}$, $\lambda^{\mathrm{C3}}_{(l,i,j)}$, *matrices* Q *and* S, *and vectors* q *and* s. *Moreover, if* $\dim(\mathcal{H}) \geq n + 3m - 1$ *and there exists* $G \in \mathbb{S}^{n+3m-1}_{++}$ *and* $\boldsymbol{\chi} \in \mathbb{R}^{2m}$ *such that*

$$\begin{gathered} \boldsymbol{a}^{\top}_{(l,i,j)} \boldsymbol{\chi} + \operatorname{trace}(\boldsymbol{M}_{(l,i,j)} G) \leq 0 \\ \textit{for each } l \in [\![1, m]\!] \textit{ and distinct } i, j \in \{\varnothing, +, \star\}, \end{gathered} \tag{5.6}$$

then the feasibility of (5.5) *is also a necessary condition.*

The proof of Theorem 5.1 is based on, for C1, C2 and C3 in Definition 4.3, finding the relevant conditions, respectively, and then combining these conditions together to give (5.5); (5.5a) correspond to C1, (5.5b) correspond to C2 and (5.5c) correspond to C3.

Finding the conditions for C1, C2 and C3 is done in the spirit of PEP by finding the worst-case behavior over algorithm-consistent points, their successors, fixed points (2.6), and mappings in the class $\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$. In particular, given some objective $\Phi : \mathcal{S}^3 \to \mathbb{R}$, the performance estimation problem we consider is

$$\begin{array}{ll} \text{maximize} & \Phi(\boldsymbol{\xi}, \boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \\ \text{subject to} & \boldsymbol{x}_+ = (A \otimes \mathrm{Id})\boldsymbol{x} + (B \otimes \mathrm{Id})\boldsymbol{u}, \\ & \boldsymbol{y} = (C \otimes \mathrm{Id})\boldsymbol{x} + (D \otimes \mathrm{Id})\boldsymbol{u}, \\ & \boldsymbol{u} \in \boldsymbol{\partial f}(\boldsymbol{y}), \\ & \boldsymbol{F} = \boldsymbol{f}(\boldsymbol{y}), \\ & \boldsymbol{y}_+ = (C \otimes \mathrm{Id})\boldsymbol{x}_+ + (D \otimes \mathrm{Id})\boldsymbol{u}_+, \\ & \boldsymbol{u}_+ \in \boldsymbol{\partial f}(\boldsymbol{y}_+), \\ & \boldsymbol{F}_+ = \boldsymbol{f}(\boldsymbol{y}_+), \\ & \boldsymbol{x}_\star = (A \otimes \mathrm{Id})\boldsymbol{x}_\star + (B \otimes \mathrm{Id})\boldsymbol{u}_\star, \\ & \boldsymbol{y}_\star = (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star, \\ & \boldsymbol{u}_\star \in \boldsymbol{\partial f}(\boldsymbol{y}_\star), \\ & \boldsymbol{F}_\star = \boldsymbol{f}(\boldsymbol{y}_\star), \\ & \boldsymbol{\xi} = (\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{y}, \boldsymbol{F}), \\ & \boldsymbol{\xi}_+ = (\boldsymbol{x}_+, \boldsymbol{u}_+, \boldsymbol{y}_+, \boldsymbol{F}_+), \\ & \boldsymbol{\xi}_\star = (\boldsymbol{x}_\star, \boldsymbol{u}_\star, \boldsymbol{y}_\star, \boldsymbol{F}_\star), \\ & \boldsymbol{f} \in \boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}, \end{array} \tag{PEP}$$

where we maximize over all variables except A, B, C, D, $\boldsymbol{\mathcal{F}}_{\boldsymbol{\sigma},\boldsymbol{\beta}}$ and Φ. Let $S^\star_\Phi$ be the optimal value of (PEP).

We consider objective functions Φ in (PEP) of the form

$$\begin{aligned} \Phi(\boldsymbol{\xi}, \boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) = {} & \mathcal{Q}(Q_\varnothing, (\boldsymbol{x} - \boldsymbol{x}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) + q_\varnothing^\top (\boldsymbol{F} - \boldsymbol{F}_\star) \\ & + \mathcal{Q}(Q_+, (\boldsymbol{x}_+ - \boldsymbol{x}_\star, \boldsymbol{u}_+, \boldsymbol{u}_\star)) + q_+^\top (\boldsymbol{F}_+ - \boldsymbol{F}_\star), \end{aligned} \tag{5.7}$$

parameterized by $Q_\varnothing, Q_+ \in \mathbb{S}^{n+2m}$ and $q_\varnothing, q_+ \in \mathbb{R}^m$. For each *cond* $\in \{\mathrm{C1}, \mathrm{C2}, \mathrm{C3}\}$ separately, the parameters $Q_\varnothing$, Q_+, $q_\varnothing$ and q_+ are chosen such that $S^\star_\Phi \leq 0$ is a necessary and sufficient condition for *cond* to hold.

Before we proceed, we reformulate (PEP), and in order to do so we introduce some helpful notation. We let

$$\boldsymbol{Q} = \Sigma_{\varnothing}^{\top} Q_{\varnothing} \Sigma_{\varnothing} + \Sigma_{+}^{\top} Q_{+} \Sigma_{+} \in \mathbb{S}^{n+3m-1}, \quad \boldsymbol{q} = (q_{\varnothing}, q_{+}) \in \mathbb{R}^{2m}, \tag{5.8}$$

where $Q_{\varnothing}$, Q_{+}, $q_{\varnothing}$, and q_{+} are the parameters in the objective function Φ given in (5.7), and $\Sigma_{\varnothing}$ and Σ_{+} are given in (5.4).

Lemma 5.2

Let $\mathcal{I} = \{\varnothing, \star\}$ or $\mathcal{I} = \{\varnothing, +, \star\}$, $S_{\Phi}^{\star}$ the optimal value of (PEP), and assume that Assumption 2.2 and Assumption 2.6 hold. Suppose that Φ is of the form (5.7) and that the right-hand side of (5.7) only depends on variables with indices in the set $\mathcal{I}$ (a variable without a subscript is interpreted to have index $\varnothing$). A sufficient condition for $S_{\Phi}^{\star} \leq 0$ is that the following system

$$\begin{cases} \lambda_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \mathcal{I}, \\ -\boldsymbol{Q} + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j \in \mathcal{I} \\ i \neq j}} \lambda_{(l,i,j)} \boldsymbol{M}_{(l,i,j)} \succeq 0, \\ -\boldsymbol{q} + \displaystyle\sum_{l=1}^{m} \sum_{\substack{i,j \in \mathcal{I} \\ i \neq j}} \lambda_{(l,i,j)} \boldsymbol{a}_{(l,i,j)} = 0, \end{cases} \tag{5.9}$$

is feasible for the scalars $\lambda_{(l,i,j)}$. Furthermore, if $\dim(\mathcal{H}) \geq n + 3m - 1$, and there exists $G \in \mathbb{S}_{++}^{n+3m-1}$ and $\boldsymbol{\chi} \in \mathbb{R}^{2m}$ such that

$$\begin{gathered} \boldsymbol{a}_{(l,i,j)}^{\top} \boldsymbol{\chi} + \operatorname{trace}(\boldsymbol{M}_{(l,i,j)} G) \leq 0 \\ \text{for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \mathcal{I}, \end{gathered} \tag{5.10}$$

then (5.9) is a necessary condition.

Proof sketch. The full proof is provided in Section 7. The proof first reformulates (PEP) as a semidefinite program, forms the dual problem, which is equal to the feasibility problem (5.9), and shows strong duality when $\dim(\mathcal{H}) \geq n + 3m - 1$ and (5.10) holds. □

Proof of Theorem 5.1. First, suppose that the parameters (Q, q, T, t) are fixed in some Lyapunov function $V : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ as in (4.2) and some residual function $R : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ as in (4.3). We consider when V and R satisfy the (P, p, T, t, ρ)-quadratic Lyapunov inequality.

C1 holds if $S_{\Phi}^{\star} \leq 0$ for the choice $Q_{\varnothing} = S - \rho Q$, $q_{\varnothing} = s - \rho q$, $Q_{+} = Q$, and $q_{+} = q$, which in turn holds if (5.5a) is feasible, according to Lemma 5.2.

C2 holds if $S_{\Phi}^{\star} \leq 0$ for the choice $Q_{\varnothing} = P - Q$, $q_{\varnothing} = p - q$, $Q_{+} = 0$ and $q_{+} = 0$, which in turn holds if (5.5b) is feasible, according to Lemma 5.2.

C3 holds if $S_{\Phi}^{\star} \leq 0$ for the choice $Q_{\varnothing} = T - S$, $q_{\varnothing} = t - s$, $Q_{+} = 0$ and $q_{+} = 0$, which in turn holds if (5.5c) is feasible, according to Lemma 5.2.

If in addition $\dim(\mathcal{H}) \geq n+3m-1$ and (5.6) hold, then Lemma 5.2 gives that feasibility of (5.5a)-(5.5c) is a necessary condition for C1, C2 and C3 to hold simultaneously.

Second, note that the proof is complete if we let the parameters (Q, q, T, t) free, as in (5.5d)-(5.5e). □

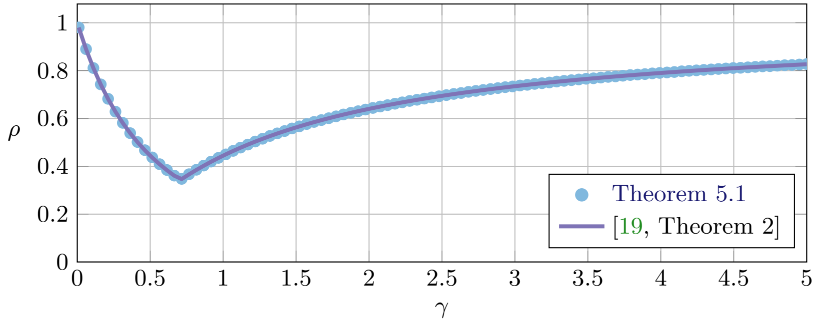


Figure 1. B1 applied to the Douglas–Rachford method (see Section 2.5.1) when $f_1 \in \mathcal{F}_{1,2}$, $f_2 \in \mathcal{F}_{0,\infty}$ and $\lambda = 1$, and the tight convergence rate given in [19, Theorem 2].

6. Numerical examples

The necessary and sufficient condition (5.5) in Theorem 5.1 for the existence of a Lyapunov inequality is a semidefinite program of size $n + 2m$ (which is below ten for all examples in Section 2.5) and is readily solved by standard solvers. We apply Theorem 5.1 to each example in Section 2.5 in two different ways:

B1 We find the smallest possible $\rho \in [0, 1[$, via bisection search, such that a (P, p, T, t, ρ)-Lyapunov inequality exists, where (P, p, T, t) is chosen as in (4.9), which implies that the squared distance to the solution convergence ρ-linearly to zero. The tolerance for the bisection search is set to 0.001 and i is set to 1 in (4.9) for all examples.

B2 We fix $\rho = 1$ and find a range of algorithm parameters for which there exists a (P, p, T, t, ρ)-Lyapunov inequality, where (P, p, T, t) is chosen as in (4.10) if $m = 1$ and (4.11) if $m > 1$, implying ($\mathcal{O}(1/k)$ ergodic) convergence of the function-value suboptimality and duality gap, respectively. The parameter range is evaluated on a square grid of size 0.01×0.01.

6.1 Douglas–Rachford method

Consider the Douglas–Rachford method in Section 2.5.1 in the case when $f_1 \in \mathcal{F}_{1,2}$, $f_2 \in \mathcal{F}_{0,\infty}$ and $\lambda = 1$. Figure 1 shows the ρ we obtain via B1. In particular, note that we recover the already known tight rates given in [19, Theorem 2].

6.2 Proximal gradient method with heavy-ball momentum

Consider the gradient method with heavy-ball momentum in Section 2.5.2 and the proximal operator extension in Section 2.5.3. Note that the method in Section 2.5.3 reduces to the one in Section 2.5.2 if the proximal operator is removed and either $\delta_1 = 0$ or $\delta_2 = 0$.

Figure 2a contains the parameter region we obtain via B2 for the gradient method with heavy-ball momentum when $f_1 \in \mathcal{F}_{0,1}$. Note that we improve on the parameter region given in [17] that guarantees $\mathcal{O}(1/k)$ ergodic convergence of the function-value suboptimality.

Figure 2b contains the parameter region we obtain via B2 for the (proximal) gradient method with heavy-ball momentum when $f_1 \in \mathcal{F}_{0,1}$ (and $f_2 \in \mathcal{F}_{0,\infty}$). In particular, note how the feasible parameter region is affected by adding a proximal term—having the momentum term inside the proximal evaluation ($\delta_2 = 0$) gives a slightly smaller region, and having it outside ($\delta_1 = 0$) makes it even smaller.

Figure 2c shows the ρ we obtain via B1 for the gradient method with heavy-ball momentum when $f_1 \in \mathcal{F}_{1,10}$. Note that we improve on the rates given in [17] and range of allowable momentum parameters δ that guarantee linear convergence.

6.3 Davis–Yin three-operator splitting method

Consider the three-operator splitting method by Davis–Yin in Section 2.5.4 in the case when $f_1 \in \mathcal{F}_{0,\beta_1}$, $f_2 \in \mathcal{F}_{1,2}$, $f_2 \in \mathcal{F}_{0,\infty}$, $\gamma = 1/2$ and $\lambda = 1$. Figure 3 shows the ρ we obtain via B1. In particular, note that we improve on the rates given in [11, Theorem D.6] and [31, Theorem 3].

6.4 Chambolle–Pock method

Consider the special case of the Chambolle–Pock method when the linear operator is restricted to be the identity operator Id, as presented in Section 2.5.5. Standard convergence proofs, e.g., the ones in [7], allow in this setting for $\theta = 1$ and $\tau_1, \tau_2 > 0$ satisfying $\tau_1\tau_2 < 1$.

Figure 4a shows the range of parameters θ, τ_1, and τ_2 when $\tau_1 = \tau_2 \geq 0.5$ that we obtain via B2 when $f_1, f_2 \in \mathcal{F}_{0,\infty}$. This is a significantly larger region than what traditional analyses allow for. In particular, we see that $\theta \neq 1$ is a valid choice and that $\tau_1\tau_2 > 1$ is also valid for many choices of θ. Moreover, for comparison, Figure 4a also contains the region if we add the additional restriction in (5.5) that

$$Q = \begin{bmatrix} Q_{xx} & 0 \\ 0 & 0 \end{bmatrix}, \tag{6.1}$$

where $Q_{xx} \in \mathbb{S}^n$ and modify P in B2 so that

$$P = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \tag{6.2}$$

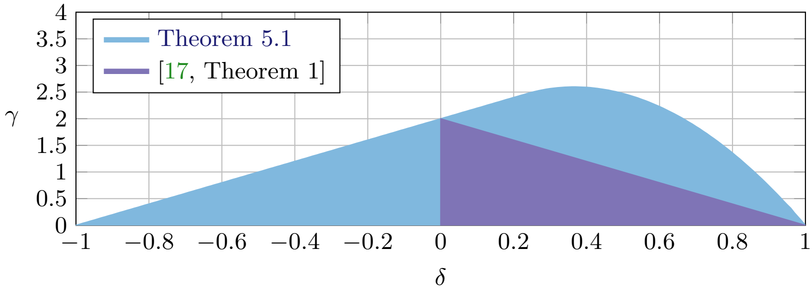
where I is the identity matrix of size $n \times n$. With these additional restrictions, we recover exactly the traditional convergence region.

Figure 4b shows the ρ that we obtain via B1 when $f_1, f_2 \in \mathcal{F}_{0.05,50}$ in the region when $\tau_1 = \tau_2 \geq 0.5$. In particular, we note that the smallest ρ is obtained for the parameters $\tau_1 = \tau_2 = 1.6$ and $\theta = 0.22$, giving a value of $\rho = 0.8812$. If we restrict to the feasible parameter region in Figure 4a, the optimal parameters are $\tau_1 = \tau_2 = 1.5$ and $\theta = 0.35$ with $\rho = 0.8891$. Both these rates are significantly better than what can be achieved with traditional parameter choices, where the optimal choice is $\tau_1 = \tau_2 = 0.99$ and $\theta = 1$ giving $\rho = 0.9266$.

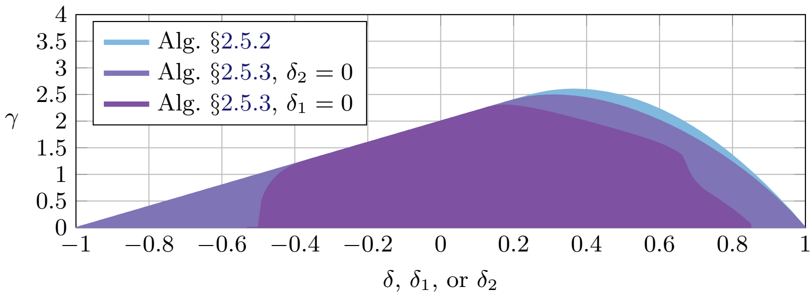
7. Proof of Lemma 5.2

We prove Lemma 5.2 only in the case $\mathcal{I} = \{\varnothing, +, \star\}$, as the case $\mathcal{I} = \{\varnothing, \star\}$ is analogous. Recall that we assume that Assumption 2.2 and Assumption 2.6 hold.

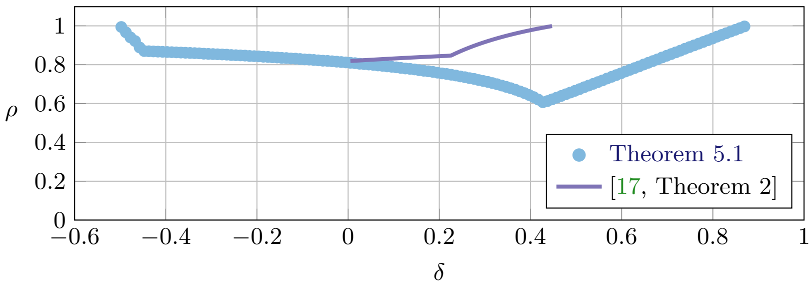
Proof of Lemma 5.2. We prove Lemma 5.2 in a sequence of steps:



(a) B2 applied to the gradient method with heavy-ball momentum when $f_1 \in \mathcal{F}_{0,1}$, and compared to the range given in [17, Theorem 1] that gives $\mathcal{O}(1/k)$ ergodic convergence of the function-value suboptimality.



(b) B2 applied to the (proximal) gradient method with heavy-ball momentum when $f_1 \in \mathcal{F}_{0,1}$ (and $f_2 \in \mathcal{F}_{0,\infty}$).



(c) B1 applied to the gradient method with heavy-ball momentum when $f_1 \in \mathcal{F}_{1,10}$ and $\gamma = 1/10$.

Figure 2. Convergence analysis of the (proximal) gradient method with heavy-ball momentum (see Sections 2.5.2 and 2.5.3).

Formulating the primal semidefinite program. Recall that $S^\star_\Phi$ is the optimal value of (PEP). By Corollary 3.2, the constraints of (PEP) can equivalently be written as

$$\boldsymbol{x}_+ = (A \otimes \mathrm{Id})\boldsymbol{x} + (B \otimes \mathrm{Id})\boldsymbol{u},$$

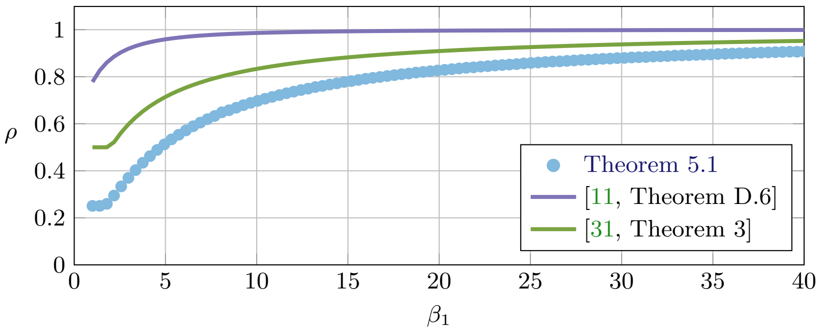
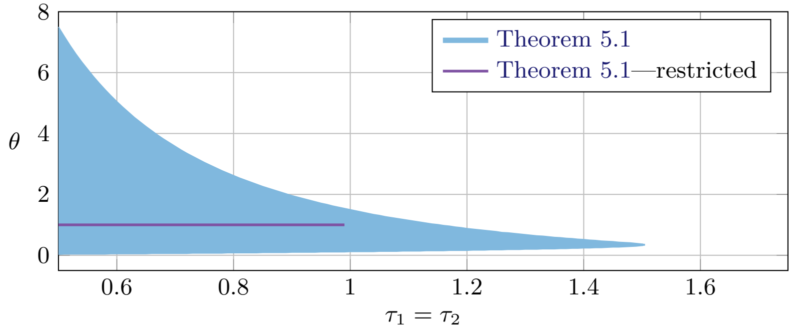


Figure 3. B1 applied to the three-operator splitting method by Davis and Yin (see Section 2.5.4) when $f_1 \in \mathcal{F}_{0,\beta_1}$, $f_2 \in \mathcal{F}_{1,2}$, $f_3 \in \mathcal{F}_{0,\infty}$, $\gamma = 1/2$ and $\lambda = 1$, the linear convergence rate given in [11, Theorem D.6], and the linear convergence rate given in [31, Theorem 3].

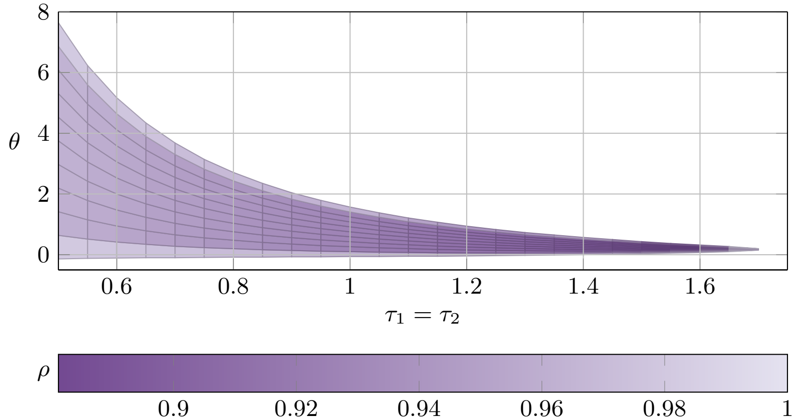
$$
\begin{aligned}
&\boldsymbol{y} = (C \otimes \mathrm{Id})\boldsymbol{x} + (D \otimes \mathrm{Id})\boldsymbol{u},\\
&\boldsymbol{y}_+ = (C \otimes \mathrm{Id})\boldsymbol{x}_+ + (D \otimes \mathrm{Id})\boldsymbol{u}_+,\\
&\boldsymbol{x}_\star = (A \otimes \mathrm{Id})\boldsymbol{x}_\star + (B \otimes \mathrm{Id})\boldsymbol{u}_\star,\\
&\boldsymbol{y}_\star = (C \otimes \mathrm{Id})\boldsymbol{x}_\star + (D \otimes \mathrm{Id})\boldsymbol{u}_\star,\\
&\textbf{for each} \quad l \in [\![1, m]\!]\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F} - \boldsymbol{F}_+) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y} - \boldsymbol{y}_+, \boldsymbol{u}, \boldsymbol{u}_+)) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_+ - \boldsymbol{F}) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y}_+ - \boldsymbol{y}, \boldsymbol{u}_+, \boldsymbol{u})) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F} - \boldsymbol{F}_\star) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y} - \boldsymbol{y}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_\star - \boldsymbol{F}) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y}_\star - \boldsymbol{y}, \boldsymbol{u}_\star, \boldsymbol{u})) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_+ - \boldsymbol{F}_\star) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y}_+ - \boldsymbol{y}_\star, \boldsymbol{u}_+, \boldsymbol{u}_\star)) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_\star - \boldsymbol{F}_+) + \mathcal{Q}(\boldsymbol{M}_l, (\boldsymbol{y}_\star - \boldsymbol{y}_+, \boldsymbol{u}_\star, \boldsymbol{u}_+)) \leq 0,\\
&\quad \mathcal{Q}(\boldsymbol{M}_l, (0, \boldsymbol{u}, \boldsymbol{u})) \leq 0,\\
&\quad \mathcal{Q}(\boldsymbol{M}_l, (0, \boldsymbol{u}_+, \boldsymbol{u}_+)) \leq 0,\\
&\quad \mathcal{Q}(\boldsymbol{M}_l, (0, \boldsymbol{u}_\star, \boldsymbol{u}_\star)) \leq 0.\\
&\textbf{end}
\end{aligned} \tag{7.1}
$$

By Corollary 3.2, the last three constraints can be dropped since they encode $0 \leq 0$. By inserting the $\boldsymbol{y}$, $\boldsymbol{y}_+$, and $\boldsymbol{y}_\star$ equalities and using the notation $X_{\mathrm{Id}} = (X \otimes \mathrm{Id})$, the constraints in (7.1) can be written as

$$
\begin{aligned}
&\boldsymbol{x}_+ = A_{\mathrm{Id}}\boldsymbol{x} + B_{\mathrm{Id}}\boldsymbol{u},\\
&\boldsymbol{x}_\star = A_{\mathrm{Id}}\boldsymbol{x}_\star + B_{\mathrm{Id}}\boldsymbol{u}_\star,\\
&\textbf{for each} \quad l \in [\![1, m]\!]\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F} - \boldsymbol{F}_+) + \mathcal{Q}(\boldsymbol{M}_l, (C_{\mathrm{Id}}(\boldsymbol{x} - \boldsymbol{x}_+) + D_{\mathrm{Id}}\boldsymbol{u} - D_{\mathrm{Id}}\boldsymbol{u}_+, \boldsymbol{u}, \boldsymbol{u}_+)) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_+ - \boldsymbol{F}) + \mathcal{Q}(\boldsymbol{M}_l, (C_{\mathrm{Id}}(\boldsymbol{x}_+ - \boldsymbol{x}) - D_{\mathrm{Id}}\boldsymbol{u} + D_{\mathrm{Id}}\boldsymbol{u}_+, \boldsymbol{u}_+, \boldsymbol{u})) \leq 0,\\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F} - \boldsymbol{F}_\star) + \mathcal{Q}(\boldsymbol{M}_l, (C_{\mathrm{Id}}(\boldsymbol{x} - \boldsymbol{x}_\star) + D_{\mathrm{Id}}\boldsymbol{u} - D_{\mathrm{Id}}\boldsymbol{u}_\star, \boldsymbol{u}, \boldsymbol{u}_\star)) \leq 0,
\end{aligned} \tag{7.2}
$$



(a) B2 applied to the Chambolle–Pock method in the region $\tau_1 = \tau_2 \geq 0.5$ when $f_1, f_2 \in \mathcal{F}_{0,\infty}$, and also with the additional restrictions in (6.1) and (6.2).



(b) B1 applied to the Chambolle–Pock method in the region where $\tau_1 = \tau_2 \geq 0.5$ when $f_1, f_2 \in \mathcal{F}_{0.05,50}$.

Figure 4. Convergence analysis of the Chambolle–Pock method (see Section 2.5.5).

$$
\begin{aligned}
&\boldsymbol{a}_l^\top(\boldsymbol{F}_\star - \boldsymbol{F}) + \mathcal{Q}(\boldsymbol{M}_l, (-C_{\mathrm{Id}}(\boldsymbol{x} - \boldsymbol{x}_\star) - D_{\mathrm{Id}}\boldsymbol{u} + D_{\mathrm{Id}}\boldsymbol{u}_\star, \boldsymbol{u}_\star, \boldsymbol{u})) \leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}_+ - \boldsymbol{F}_\star) + \mathcal{Q}(\boldsymbol{M}_l, (C_{\mathrm{Id}}(\boldsymbol{x}_+ - \boldsymbol{x}_\star) + D_{\mathrm{Id}}\boldsymbol{u}_+ - D_{\mathrm{Id}}\boldsymbol{u}_\star, \boldsymbol{u}_+, \boldsymbol{u}_\star)) \leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}_\star - \boldsymbol{F}_+) + \mathcal{Q}(\boldsymbol{M}_l, (-C_{\mathrm{Id}}(\boldsymbol{x}_+ - \boldsymbol{x}_\star) - D_{\mathrm{Id}}\boldsymbol{u}_+ + D_{\mathrm{Id}}\boldsymbol{u}_\star, \boldsymbol{u}_\star, \boldsymbol{u}_+)) \leq 0.
\end{aligned}
$$

end

Using the equality $\boldsymbol{x} - \boldsymbol{x}_+ = (\boldsymbol{x} - \boldsymbol{x}_\star) - (A_{\mathrm{Id}}(\boldsymbol{x} - \boldsymbol{x}_\star) + B_{\mathrm{Id}}(\boldsymbol{u} - \boldsymbol{u}_\star))$ in the first two inequalities and inserting the $\boldsymbol{x}_+$ and $\boldsymbol{x}_\star$ equalities in the last two inequalities, (7.2) can equivalently be written as

$$
\begin{aligned}
&\boldsymbol{x}_+ = A_{\mathrm{Id}}\boldsymbol{x} + B_{\mathrm{Id}}\boldsymbol{u},\\
&\boldsymbol{x}_\star = A_{\mathrm{Id}}\boldsymbol{x}_\star + B_{\mathrm{Id}}\boldsymbol{u}_\star,
\end{aligned}
$$

for each $l \in [\![1, m]\!]$

$$
\begin{aligned}
&\boldsymbol{a}_l^\top(\boldsymbol{F}-\boldsymbol{F}_+)+\mathcal{Q}(\boldsymbol{M}_l,((C(I-A))_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)\\
&\quad+(D-CB)_{\mathrm{Id}}\boldsymbol{u}-D_{\mathrm{Id}}\boldsymbol{u}_+ +(CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u},\boldsymbol{u}_+))\leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}_+-\boldsymbol{F})+\mathcal{Q}(\boldsymbol{M}_l,((C(A-I))_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)\\
&\quad+(CB-D)_{\mathrm{Id}}\boldsymbol{u}+D_{\mathrm{Id}}\boldsymbol{u}_+ -(CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_+,\boldsymbol{u}))\leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}-\boldsymbol{F}_\star)+\mathcal{Q}(\boldsymbol{M}_l,(C_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)+D_{\mathrm{Id}}\boldsymbol{u}-D_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u},\boldsymbol{u}_\star))\leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}_\star-\boldsymbol{F})+\mathcal{Q}(\boldsymbol{M}_l,(-C_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)-D_{\mathrm{Id}}\boldsymbol{u}+D_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_\star,\boldsymbol{u}))\leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}_+-\boldsymbol{F}_\star)+\mathcal{Q}(\boldsymbol{M}_l,(CA)_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)+(CB)_{\mathrm{Id}}\boldsymbol{u}\\
&\quad+D_{\mathrm{Id}}\boldsymbol{u}_+ -(D+CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_+,\boldsymbol{u}_\star))\leq 0,\\
&\boldsymbol{a}_l^\top(\boldsymbol{F}_\star-\boldsymbol{F}_+)+\mathcal{Q}(\boldsymbol{M}_l,(-(CA)_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)\\
&\quad-(CB)_{\mathrm{Id}}\boldsymbol{u}-D_{\mathrm{Id}}\boldsymbol{u}_+ +(D+CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_\star,\boldsymbol{u}_+))\leq 0,\\
&\textbf{end}
\end{aligned}
\tag{7.3}
$$

and using the same equality $\boldsymbol{x}_+-\boldsymbol{x}_\star=A_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)+B_{\mathrm{Id}}(\boldsymbol{u}-\boldsymbol{u}_\star)$, the objective function $\Phi(\boldsymbol{\xi},\boldsymbol{\xi}_+,\boldsymbol{\xi}_\star)$ of (PEP), given in (5.7), can be written as

$$
\begin{aligned}
&\Phi(\boldsymbol{\xi},\boldsymbol{\xi}_+,\boldsymbol{\xi}_\star)\\
&=\mathcal{Q}(Q_\varnothing,(\boldsymbol{x}-\boldsymbol{x}_\star,\boldsymbol{u},\boldsymbol{u}_\star))+q_\varnothing^\top(\boldsymbol{F}-\boldsymbol{F}_\star)\\
&\quad+\mathcal{Q}(Q_+,(A_{\mathrm{Id}}(\boldsymbol{x}-\boldsymbol{x}_\star)+B_{\mathrm{Id}}\boldsymbol{u}-B_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_+,\boldsymbol{u}_\star))+q_+^\top(\boldsymbol{F}_+-\boldsymbol{F}_\star).
\end{aligned}
\tag{7.4}
$$

Therefore, the first equality in (7.3) can be dropped since nothing else in (7.3) and (7.4) depend on $\boldsymbol{x}_+$. Moreover, by replacing $\boldsymbol{x}-\boldsymbol{x}_\star$ with $\Delta\boldsymbol{x}$, we get that (7.3) can equivalently be written as

$$
\begin{aligned}
&\boldsymbol{x}_\star=A_{\mathrm{Id}}\boldsymbol{x}_\star+B_{\mathrm{Id}}\boldsymbol{u}_\star,\\
&\textbf{for each}\quad l\in[\![1,m]\!]\\
&\quad\boldsymbol{a}_l^\top(\boldsymbol{F}-\boldsymbol{F}_+)+\mathcal{Q}(\boldsymbol{M}_l,((C(I-A))_{\mathrm{Id}}\Delta\boldsymbol{x}+(D-CB)_{\mathrm{Id}}\boldsymbol{u}\\
&\quad\quad-D_{\mathrm{Id}}\boldsymbol{u}_+ +(CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u},\boldsymbol{u}_+))\leq 0,\\
&\quad\boldsymbol{a}_l^\top(\boldsymbol{F}_+-\boldsymbol{F})+\mathcal{Q}(\boldsymbol{M}_l,((C(A-I))_{\mathrm{Id}}\Delta\boldsymbol{x}+(CB-D)_{\mathrm{Id}}\boldsymbol{u}\\
&\quad\quad+D_{\mathrm{Id}}\boldsymbol{u}_+ -(CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_+,\boldsymbol{u}))\leq 0,\\
&\quad\boldsymbol{a}_l^\top(\boldsymbol{F}-\boldsymbol{F}_\star)+\mathcal{Q}(\boldsymbol{M}_l,(C_{\mathrm{Id}}\Delta\boldsymbol{x}+D_{\mathrm{Id}}\boldsymbol{u}-D_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u},\boldsymbol{u}_\star))\leq 0,\\
&\quad\boldsymbol{a}_l^\top(\boldsymbol{F}_\star-\boldsymbol{F})+\mathcal{Q}(\boldsymbol{M}_l,(-C_{\mathrm{Id}}\Delta\boldsymbol{x}-D_{\mathrm{Id}}\boldsymbol{u}+D_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_\star,\boldsymbol{u}))\leq 0,\\
&\quad\boldsymbol{a}_l^\top(\boldsymbol{F}_+-\boldsymbol{F}_\star)+\mathcal{Q}(\boldsymbol{M}_l,(CA)_{\mathrm{Id}}\Delta\boldsymbol{x}+(CB)_{\mathrm{Id}}\boldsymbol{u}+D_{\mathrm{Id}}\boldsymbol{u}_+\\
&\quad\quad-(D+CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_+,\boldsymbol{u}_\star))\leq 0,\\
&\quad\boldsymbol{a}_l^\top(\boldsymbol{F}_\star-\boldsymbol{F}_+)+\mathcal{Q}(\boldsymbol{M}_l,(-(CA)_{\mathrm{Id}}\Delta\boldsymbol{x}-(CB)_{\mathrm{Id}}\boldsymbol{u}-D_{\mathrm{Id}}\boldsymbol{u}_+\\
&\quad\quad+(D+CB)_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_\star,\boldsymbol{u}_+))\leq 0,\\
&\textbf{end}
\end{aligned}
\tag{7.5}
$$

and that (7.4) can equivalently be written as

$$
\begin{aligned}
&\Phi(\boldsymbol{\xi},\boldsymbol{\xi}_+,\boldsymbol{\xi}_\star)\\
&=\mathcal{Q}(Q_\varnothing,(\Delta\boldsymbol{x},\boldsymbol{u},\boldsymbol{u}_\star))+q_\varnothing^\top(\boldsymbol{F}-\boldsymbol{F}_\star)\\
&\quad+\mathcal{Q}(Q_+,(A_{\mathrm{Id}}\Delta\boldsymbol{x}+B_{\mathrm{Id}}\boldsymbol{u}-B_{\mathrm{Id}}\boldsymbol{u}_\star,\boldsymbol{u}_+,\boldsymbol{u}_\star))+q_+^\top(\boldsymbol{F}_+-\boldsymbol{F}_\star).
\end{aligned}
\tag{7.6}
$$

The first line in (7.5) and (2.12) in Assumption 2.2 imply that

$$\boldsymbol{u}_\star = \begin{cases} 0 & \text{if } m = 1, \\ N_{\mathrm{Id}} \hat{\boldsymbol{u}}_\star & \text{if } m > 1. \end{cases}$$

for some $\hat{\boldsymbol{u}}_\star \in \mathcal{H}^{m-1}$, where N is defined in (2.10). This implies that the first line in (7.5) can be written as $\boldsymbol{x}_\star = A_{\mathrm{Id}}\boldsymbol{x}_\star$ if $m = 1$ and $\boldsymbol{x}_\star = A_{\mathrm{Id}}\boldsymbol{x}_\star + (BN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star$ if $m > 1$. Moreover, note that nothing else in (7.5) and (7.6) depend on $\boldsymbol{x}_\star$. Therefore, $\boldsymbol{x}_\star = 0$ is a valid choice in the $m = 1$ case and in the $m > 1$ case (2.11) in Assumption 2.2 gives that the first line in (7.5) can be dropped since for each $\hat{\boldsymbol{u}}_\star \in \mathcal{H}^{m-1}$ there exists an $\boldsymbol{x}_\star \in \mathcal{H}^n$ such that $\boldsymbol{x}_\star = A_{\mathrm{Id}}\boldsymbol{x}_\star + (BN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star$ is satisfied. Therefore, (7.5) can equivalently be written as

$$\begin{aligned}
&\textbf{for each} \quad l \in [\![1, m]\!] \\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F} - \boldsymbol{F}_+) + \mathcal{Q}(\boldsymbol{M}_l, ((C(I-A))_{\mathrm{Id}}\Delta\boldsymbol{x} + (D - CB)_{\mathrm{Id}}\boldsymbol{u} - D_{\mathrm{Id}}\boldsymbol{u}_+ \\
&\qquad + (CBN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u}, \boldsymbol{u}_+)) \leq 0, \\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_+ - \boldsymbol{F}) + \mathcal{Q}(\boldsymbol{M}_l, ((C(A-I))_{\mathrm{Id}}\Delta\boldsymbol{x} + (CB - D)_{\mathrm{Id}}\boldsymbol{u} + D_{\mathrm{Id}}\boldsymbol{u}_+, \\
&\qquad - (CBN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u}_+, \boldsymbol{u})) \leq 0, \\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F} - \boldsymbol{F}_\star) + \mathcal{Q}(\boldsymbol{M}_l, (C_{\mathrm{Id}}\Delta\boldsymbol{x} + D_{\mathrm{Id}}\boldsymbol{u} - (DN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u}, N_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star)) \leq 0, \\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_\star - \boldsymbol{F}) + \mathcal{Q}(\boldsymbol{M}_l, (-C_{\mathrm{Id}}\Delta\boldsymbol{x} - D_{\mathrm{Id}}\boldsymbol{u} + (DN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, N_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u})) \leq 0, \\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_+ - \boldsymbol{F}_\star) + \mathcal{Q}(\boldsymbol{M}_l, (CA)_{\mathrm{Id}}\Delta\boldsymbol{x} + (CB)_{\mathrm{Id}}\boldsymbol{u} + D_{\mathrm{Id}}\boldsymbol{u}_+ \\
&\qquad - ((D + CB)N)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u}_+, N_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star)) \leq 0, \\
&\quad \boldsymbol{a}_l^\top(\boldsymbol{F}_\star - \boldsymbol{F}_+) + \mathcal{Q}(\boldsymbol{M}_l, (-(CA)_{\mathrm{Id}}\Delta\boldsymbol{x} - (CB)_{\mathrm{Id}}\boldsymbol{u} - D_{\mathrm{Id}}\boldsymbol{u}_+ \\
&\qquad + ((D + CB)N)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, N_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u}_+)) \leq 0, \\
&\textbf{end}
\end{aligned} \tag{7.7}$$

and (7.6) can equivalently be written as

$$\begin{aligned}
&\Phi(\boldsymbol{\xi}, \boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) \\
&= \mathcal{Q}(Q_\varnothing, (\Delta\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{u}_\star)) + q_\varnothing^\top(\boldsymbol{F} - \boldsymbol{F}_\star) \\
&\quad + \mathcal{Q}(Q_+, (A_{\mathrm{Id}}\Delta\boldsymbol{x} + B_{\mathrm{Id}}\boldsymbol{u} - (BN)_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star, \boldsymbol{u}_+, N_{\mathrm{Id}}\hat{\boldsymbol{u}}_\star)) + q_+^\top(\boldsymbol{F}_+ - \boldsymbol{F}_\star).
\end{aligned} \tag{7.8}$$

If we let

$$\begin{aligned}
\boldsymbol{\zeta} &= (\Delta\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{u}_+, \hat{\boldsymbol{u}}_\star) \in \mathcal{H}^n \times \mathcal{H}^m \times \mathcal{H}^m \times \mathcal{H}^{m-1}, \\
\boldsymbol{\chi} &= (\boldsymbol{F} - \boldsymbol{F}_\star, \boldsymbol{F}_+ - \boldsymbol{F}_\star) \in \mathbb{R}^m \times \mathbb{R}^m,
\end{aligned}$$

and use $\Sigma_\varnothing$ and Σ_+ defined in (5.4), (7.8) can equivalently be written as

$$\begin{aligned}
\Phi(\boldsymbol{\xi}, \boldsymbol{\xi}_+, \boldsymbol{\xi}_\star) &= \mathcal{Q}(Q_\varnothing, (\Sigma_\varnothing)_{\mathrm{Id}}\boldsymbol{\zeta}) + \mathcal{Q}(Q_+, (\Sigma_+)_{\mathrm{Id}}\boldsymbol{\zeta}) + \boldsymbol{q}^\top\boldsymbol{\chi} \\
&= \mathcal{Q}(\Sigma_\varnothing^\top Q_\varnothing \Sigma_\varnothing + \Sigma_+^\top Q_+ \Sigma_+, \boldsymbol{\zeta}) + \boldsymbol{q}^\top\boldsymbol{\chi} \\
&= \mathcal{Q}(\boldsymbol{Q}, \boldsymbol{\zeta}) + \boldsymbol{q}^\top\boldsymbol{\chi},
\end{aligned} \tag{7.9}$$

where $\boldsymbol{Q}$ and $\boldsymbol{q}$ are defined in (5.8). Using $E_{i,j}$ and $H_{i,j}$ defined in (5.1) and (5.2), respectively, (7.7) can equivalently be written as

$$\begin{aligned}&\textbf{for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \mathcal{I}\\ &\quad (H_{i,j}^\top \boldsymbol{a}_l)^\top \boldsymbol{\chi} + \mathcal{Q}(\boldsymbol{M}_l, (E_{i,j})_{\mathrm{Id}}\boldsymbol{\zeta}) \leq 0,\\ &\textbf{end}\end{aligned}$$

which with $\boldsymbol{M}_{(l,i,j)} = E_{i,j}^\top \boldsymbol{M}_l E_{i,j}$ and $\boldsymbol{a}_{(l,i,j)} = H_{i,j}^\top \boldsymbol{a}_l$ (also defined in (5.3)) is equivalent to

$$\begin{aligned}&\textbf{for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \mathcal{I}\\ &\quad \boldsymbol{a}_{(l,i,j)}^\top \boldsymbol{\chi} + \mathcal{Q}(\boldsymbol{M}_{(l,i,j)}, \boldsymbol{\zeta}) \leq 0.\\ &\textbf{end}\end{aligned} \tag{7.10}$$

The equivalent reformulations (7.9) and (7.10) give that (PEP) can be written as

$$\begin{aligned}\text{maximize} \quad & \mathcal{Q}(\boldsymbol{Q},\boldsymbol{\zeta}) + \boldsymbol{q}^\top \boldsymbol{\chi}\\ \text{subject to} \quad & \textbf{for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \mathcal{I}\\ & \quad \boldsymbol{a}_{(l,i,j)}^\top \boldsymbol{\chi} + \mathcal{Q}(\boldsymbol{M}_{(l,i,j)}, \boldsymbol{\zeta}) \leq 0,\\ & \textbf{end}\\ & \boldsymbol{\zeta} \in \mathcal{H}^{n+3m-1},\ \boldsymbol{\chi} \in \mathbb{R}^{2m}.\end{aligned} \tag{7.11}$$

We define the *Gramian function* $g : \mathcal{H}^k \to \mathbb{S}_+^k$ such that $[g(\boldsymbol{z})]_{i,j} = \langle z^{(i)}, z^{(j)} \rangle$ for each $i,j \in [\![1,k]\!]$ and $\boldsymbol{z} = (z^{(1)},\ldots,z^{(k)}) \in \mathcal{H}^k$. If $M \in \mathbb{S}^k$ and $\boldsymbol{z} \in \mathcal{H}^k$, then $\mathcal{Q}(M,\boldsymbol{z}) = \operatorname{trace}(Mg(\boldsymbol{z}))$. Using this identity, (7.11) can be written as

$$\begin{aligned}\text{maximize} \quad & \operatorname{trace}(\boldsymbol{Q}g(\boldsymbol{\zeta})) + \boldsymbol{q}^\top \boldsymbol{\chi}\\ \text{subject to} \quad & \textbf{for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \mathcal{I}\\ & \quad \boldsymbol{a}_{(l,i,j)}^\top \boldsymbol{\chi} + \operatorname{trace}(\boldsymbol{M}_{(l,i,j)} g(\boldsymbol{\zeta})) \leq 0,\\ & \textbf{end}\\ & \boldsymbol{\zeta} \in \mathcal{H}^{n+3m-1},\ \boldsymbol{\chi} \in \mathbb{R}^{2m},\end{aligned} \tag{7.12}$$

with optimal value equal to $S_\Phi^\star$. The problem

$$\begin{aligned}\text{maximize} \quad & \operatorname{trace}(\boldsymbol{Q}G) + \boldsymbol{q}^\top \boldsymbol{\chi}\\ \text{subject to} \quad & \textbf{for each } l \in [\![1,m]\!] \text{ and distinct } i,j \in \mathcal{I}\\ & \quad \boldsymbol{a}_{(l,i,j)}^\top \boldsymbol{\chi} + \operatorname{trace}(\boldsymbol{M}_{(l,i,j)} G) \leq 0,\\ & \textbf{end}\\ & G \in \mathbb{S}_+^{n+3m-1},\ \boldsymbol{\chi} \in \mathbb{R}^{2m},\end{aligned} \tag{7.13}$$

is a relaxation of (7.12), and therefore, has optimal value greater or equal to $S_\Phi^\star$.

We will make use of the following fact: If $\dim \mathcal{H} \geq k$, then $G \in \mathbb{S}_+^k$ if and only if there exists $\boldsymbol{z} \in \mathcal{H}^k$ such that $G = g(\boldsymbol{z})$. [34, Lemma 3.1] shows the result for the case $k = 4$ and is based on the Cholesky decomposition of positive semidefinite matrices. The general case is a straightforward extension. This fact implies that if $\dim(\mathcal{H}) \geq n+3m-1$, then (7.13) has optimal value equal to $S_\Phi^\star$. Note that (7.13) is a semidefinite program.

Dual problem and strong duality. First, we derive the dual problem of (7.13). If we introduce dual variables $\lambda_{(l,i,j)} \geq 0$ for each $l \in [\![1, m]\!]$ and distinct $i, j \in \mathcal{I}$ for the inequality constraints, the objective function of the dual problem becomes

$$
\begin{aligned}
&\sup_{G\in\mathbb{S}_{+}^{n+3m-1},\, \boldsymbol{\chi}\in\mathbb{R}^{2m}} \Bigg(\operatorname{trace}(\boldsymbol{Q}G) + \boldsymbol{q}^{\top}\boldsymbol{\chi} \\
&\qquad\qquad - \sum_{l=1}^{m}\sum_{\substack{i,j\in\mathcal{I}\\ i\neq j}} \lambda_{(l,i,j)}\big(\boldsymbol{a}_{(l,i,j)}^{\top}\boldsymbol{\chi} + \operatorname{trace}(\boldsymbol{M}_{(l,i,j)}G)\big)\Bigg) \\
&= \sup_{G\in\mathbb{S}_{+}^{n+3m-1}} \operatorname{trace}\left(\left(\boldsymbol{Q} - \sum_{l=1}^{m}\sum_{\substack{i,j\in\mathcal{I}\\ i\neq j}} \lambda_{(l,i,j)}\boldsymbol{M}_{(l,i,j)}\right)G\right) \\
&\quad + \sup_{\boldsymbol{\chi}\in\mathbb{R}^{2m}} \left(\boldsymbol{q} - \sum_{l=1}^{m}\sum_{\substack{i,j\in\mathcal{I}\\ i\neq j}} \lambda_{(l,i,j)}\boldsymbol{a}_{(l,i,j)}\right)^{\top}\boldsymbol{\chi}.
\end{aligned}
$$

Since the dual problem is a minimization problem over the dual variables $\lambda_{(l,i,j)}$, we conclude that it can be written as

$$
\begin{array}{ll}
\text{minimize} & 0 \\
\text{subject to} & \lambda_{(l,i,j)} \geq 0 \text{ for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \mathcal{I}, \\
& -\boldsymbol{Q} + \displaystyle\sum_{l=1}^{m}\sum_{\substack{i,j\in\mathcal{I}\\ i\neq j}} \lambda_{(l,i,j)}\boldsymbol{M}_{(l,i,j)} \succeq 0, \\
& -\boldsymbol{q} + \displaystyle\sum_{l=1}^{m}\sum_{\substack{i,j\in\mathcal{I}\\ i\neq j}} \lambda_{(l,i,j)}\boldsymbol{a}_{(l,i,j)} = 0,
\end{array}
\tag{7.14}
$$

which is a feasibility problem.

Next, suppose that the primal problem (7.13) has a Slater point, i.e., there exists $G \in \mathbb{S}_{++}^{n+3m-1}$ and $\boldsymbol{\chi} \in \mathbb{R}^{2m}$ such that

$$
\begin{gathered}
\boldsymbol{a}_{(l,i,j)}^{\top}\boldsymbol{\chi} + \operatorname{trace}(\boldsymbol{M}_{(l,i,j)}G) \leq 0 \\
\text{for each } l \in [\![1, m]\!] \text{ and distinct } i, j \in \mathcal{I}.
\end{gathered}
\tag{7.15}
$$

Then there is no duality gap, i.e., strong duality holds, between the primal problem (7.13) and the dual problem (7.14).

Alternatives. The last step of the proof compares the optimal values of (PEP) and the dual problem (7.14). We have established that $S_{\Phi}^{\star}$ is less than or equal to the optimal value of (7.14). Thus, a sufficient condition for $S_{\Phi} \leq 0$ is that the dual problem (7.14) is feasible. In addition, if $\dim(\mathcal{H}) \geq n + 3m - 1$ and there exists $G \in \mathbb{S}_{++}^{n+3m-1}$ and $\boldsymbol{\chi} \in \mathbb{R}^{2m}$ such that (7.15) holds, the above condition also becomes a necessary condition.

This concludes the proof. $\square$

8. Conclusions

We developed a flexible methodology for automated convergence analysis of a large class of first-order methods for solving convex optimization problems. The main result is a necessary and sufficient condition for the existence of a quadratic Lyapunov inequality within a predefined class of Lyapunov inequalities, which amounts to solving a small-sized semidefinite program. The applicability and efficacy of the methodology are demonstrated by providing several new convergence results in Section 6.

We mention a few possible modifications that can be made to extend or modify the applicability and possibly improve the convergence results of the methodology. These were not pursued in the current work in order to maintain accessibility and not introduce unnecessary burdensome notation, but do constitute proper avenues for future works. First, each functional component f_i in (2.2) can be modified to be from any function class that has quadratic interpolation constraints, e.g., the class of smooth functions [38], the class of convex and quadratically upper bounded functions [20], the class of convex and Lipschitz continuous functions [38], etc. Second, the algorithm representation (2.5) can be extended to allow for more types of oracles (including, e.g., Frank–Wolfe-type oracles [38], Bregman-type oracles [14], or approximate proximal point oracles [3]) but also multiple evaluations of the same subdifferential ∂f_i during the same iteration, enabling the analysis of, e.g., the forward–backward–forward splitting method of Tseng [40]. Third, similar to [24, 37], it is possible to extend the quadratic Lyapunov function and the quadratic residual function ansatzes to not only contain the current iterate $\boldsymbol{\xi}_k$, but some history $\boldsymbol{\xi}_k, \boldsymbol{\xi}_{k-1}, \ldots, \boldsymbol{\xi}_{k+1-h}$ for some integer $h \geq 1$. This would allow exploring a greater class of Lyapunov inequalities that may lead to improved convergence results.

Finally, the methodology can be used in the process of finding analytical Lyapunov inequalities, convergence results, and optimal algorithm parameters. Indeed, finding a Lyapunov inequality is equivalent to solving a parametric semidefinite program. Obtaining a Lyapunov inequality involves discovering a closed-form solution for this semidefinite program, which can then be utilized to derive convergence results and select algorithm parameters. Works that aim to enable the obtaining of closed-form solutions include [21, 22], while a previous work focused on selecting algorithm parameters can be found in [42].

Acknowledgments

This work was partially supported by the ELLIIT Strategic Research Area and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. S. Banert and P. Giselsson acknowledge support from Vetenskapsrådet grant VR 2021-05710. A.B. Taylor acknowledges support from the French “Agence Nationale de la Recherche” as part of the “Investissements d’avenir” program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute), as well as support from the European Research Council (grant SEQUOIA 724063). We are grateful to Olle Kjellqvist for pointing out some of the details in Remark 2.4. We also thank Anders Rantzer for providing comments and suggestions on an earlier version of this manuscript.

References

[1] S. Banert, R. I. Bot, and E. R. Csetnek. "Fixing and extending some recent results on the ADMM algorithm". *Numerical Algorithms* **86**:3 (2021), pp. 1303–1325. DOI: 10.1007/s11075-020-00934-5.

[2] S. Banert, A. Ringh, J. Adler, J. Karlsson, and O. Öktem. "Data-driven nonsmooth optimization". *SIAM Journal on Optimization* **30**:1 (2020), pp. 102–131. DOI: 10.1137/18m1207685.

[3] M. Barré, A. B. Taylor, and F. Bach. "Principled analyses and design of first-order methods with inexact proximal operators". *Mathematical Programming* **201**:1–2 (2022), pp. 185–230. DOI: 10.1007/s10107-022-01903-7.

[4] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer International Publishing, 2017. ISBN: 9783319483115. DOI: 10.1007/978-3-319-48311-5.

[5] A. Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, 2017. ISBN: 9781611974997. DOI: 10.1137/1.9781611974997.

[6] R. I. Bot. *Conjugate Duality in Convex Optimization*. Springer Berlin Heidelberg, 2010. ISBN: 9783642049002. DOI: 10.1007/978-3-642-04900-2.

[7] A. Chambolle and T. Pock. "A first-order primal-dual algorithm for convex problems with applications to imaging". *Journal of Mathematical Imaging and Vision* **40**:1 (2011), pp. 120–145. DOI: 10.1007/s10851-010-0251-1.

[8] A. Chambolle and T. Pock. "On the ergodic convergence rates of a first-order primal-dual algorithm". *Mathematical Programming* **159**:1–2 (2016), pp. 253–287. DOI: 10.1007/s10107-015-0957-3.

[9] C.-T. Chen. *Linear System Theory and Design*. Fourth edition. Oxford University Press, New York, NY, 2013. 386 pp. ISBN: 9780199959570.

[10] M. Dahleh, M. A. Dahleh, and G. Verghese. *Lectures on Dynamic Systems and Control*. MIT OpenCourseWare, 2011.

[11] D. Davis and W. Yin. "A three-operator splitting scheme and its optimization applications" (2015). arXiv: 1504.01032v1 [math.OC].

[12] D. Davis and W. Yin. "A three-operator splitting scheme and its optimization applications". *Set-Valued and Variational Analysis* **25**:4 (2017), pp. 829–858. DOI: 10.1007/s11228-017-0421-z.

[13] J. Douglas and H. H. Rachford. "On the numerical solution of heat conduction problems in two and three space variables". *Transactions of the American Mathematical Society* **82**:2 (1956), pp. 421–439. DOI: 10.1090/s0002-9947-1956-0084194-4.

[14] R.-A. Dragomir, A. B. Taylor, A. d'Aspremont, and J. Bolte. "Optimal complexity and certification of Bregman first-order methods". *Mathematical Programming* **194**:1–2 (2021), pp. 41–83. DOI: 10.1007/s10107-021-01618-1.

[15] Y. Drori and M. Teboulle. "Performance of first-order methods for smooth convex minimization: A novel approach". *Mathematical Programming* **145**:1–2 (2014), pp. 451–482. DOI: 10.1007/s10107-013-0653-0.

[16] J. Eckstein and D. P. Bertsekas. "On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators". *Mathematical Programming* **55**:1–3 (1992), pp. 293–318. DOI: 10.1007/bf01581204.

[17] E. Ghadimi, H. R. Feyzmahdavian, and M. Johansson. "Global convergence of the heavy-ball method for convex optimization". In: *European Control Conference (ECC)*. 2015. DOI: 10.1109/ecc.2015.7330562.

[18] P. Giselsson. "Tight global linear convergence rate bounds for Douglas–Rachford splitting". *Journal of Fixed Point Theory and Applications* **19**:4 (2017), pp. 2241–2270. DOI: 10.1007/s11784-017-0417-1.

[19] P. Giselsson and S. Boyd. "Linear convergence and metric selection for Douglas–Rachford splitting and admm". *IEEE Transactions on Automatic Control* **62**:2 (2017), pp. 532–544. DOI: 10.1109/tac.2016.2564160.

[20] B. Goujaud, A. Taylor, and A. Dieuleveut. "Optimal first-order methods for convex functions with a quadratic upper bound" (2022). arXiv: 2205.15033v1 [math.OC].

[21] D. Henrion, S. Naldi, and M. S. El Din. "Exact algorithms for linear matrix inequalities". *SIAM Journal on Optimization* **26**:4 (2016), pp. 2512–2539. DOI: 10.1137/15m1036543.

[22] D. Henrion, S. Naldi, and M. Safey El Din. "SPECTRA - a Maple library for solving linear matrix inequalities in exact arithmetic". *Optimization Methods and Software* **34**:1 (2017), pp. 62–78. DOI: 10.1080/10556788.2017.1341505.

[23] L. Lessard. "The analysis of optimization algorithms: A dissipativity approach". *IEEE Control Systems* **42**:3 (2022), pp. 58–72. DOI: 10.1109/mcs.2022.3157115.

[24] L. Lessard, B. Recht, and A. Packard. "Analysis and design of optimization algorithms via integral quadratic constraints". *SIAM Journal on Optimization* **26**:1 (2016), pp. 57–95. DOI: 10.1137/15m1009597.

[25] P. L. Lions and B. Mercier. "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6 (1979), pp. 964–979. DOI: 10.1137/0716071.

[26] A. Megretski and A. Rantzer. "System analysis via integral quadratic constraints". *IEEE Transactions on Automatic Control* **42**:6 (1997), pp. 819–830. DOI: 10.1109/9.587335.

[27] S. Michalowsky, C. Scherer, and C. Ebenbauer. "Robust and structure exploiting optimisation algorithms: An integral quadratic constraint approach". *International Journal of Control* **94**:11 (2020), pp. 2956–2979. DOI: 10.1080/00207179.2020.1745286.

[28] M. Morin, S. Banert, and P. Giselsson. "Frugal splitting operators: Representation, minimal lifting, and convergence". *SIAM Journal on Optimization* **34**:2 (2024), pp. 1595–1621. DOI: 10.1137/22m1531105.

[29] C. Moucer, A. Taylor, and F. Bach. "A systematic approach to Lyapunov analyses of continuous-time models in convex optimization". *SIAM Journal on Optimization* **33**:3 (2023), pp. 1558–1586. DOI: 10.1137/22m1498486.

[30] Y. Nesterov. *Lectures on Convex Optimization*. Springer International Publishing, 2018. ISBN: 9783319915784. DOI: 10.1007/978-3-319-91578-4.

[31] F. Pedregosa and G. Gidel. "Adaptive three operator splitting". In: *International Conference on Machine Learning (ICML)*. 2018.

[32] B. T. Polyak. *Introduction to Optimization*. Translations Series in mathematics and engineering. Optimization Software, New York, 1987. 438 pp. ISBN: 0911575146.

[33] E. K. Ryu. "Uniqueness of DRS as the 2 operator resolvent-splitting and impossibility of 3 operator resolvent-splitting". *Mathematical Programming* **182**:1–2 (2020), pp. 233–273. DOI: 10.1007/s10107-019-01403-1.

[34] E. K. Ryu, A. B. Taylor, C. Bergeling, and P. Giselsson. "Operator splitting performance estimation: Tight contraction factors and optimal parameter selection". *SIAM Journal on Optimization* **30**:3 (2020), pp. 2251–2271. DOI: 10.1137/19m1304854.

[35] A. Sundararajan, B. Van Scoy, and L. Lessard. "Analysis and design of first-order distributed optimization algorithms over time-varying graphs". *IEEE Transactions on Control of Network Systems* **7**:4 (2020), pp. 1597–1608. DOI: 10.1109/tcns.2020.2988009.

[36] A. Taylor and F. Bach. "Stochastic first-order methods: Non-asymptotic and computer-aided analyses via potential functions". In: *Conference on Learning Theory (COLT)*. 2019.

[37] A. Taylor, B. Van Scoy, and L. Lessard. "Lyapunov functions for first-order methods: Tight automated convergence guarantees". In: *International Conference on Machine Learning (ICML)*. 2018.

[38] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Exact worst-case performance of first-order methods for composite convex optimization". *SIAM Journal on Optimization* **27**:3 (2017), pp. 1283–1313. DOI: 10.1137/16m108104x.

[39] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Smooth strongly convex interpolation and exact worst-case performance of first-order methods". *Mathematical Programming* **161**:1–2 (2016), pp. 307–345. DOI: 10.1007/s10107-016-1009-3.

[40] P. Tseng. "A modified forward-backward splitting method for maximal monotone mappings". *SIAM Journal on Control and Optimization* **38**:2 (2000), pp. 431–446. DOI: 10.1137/s0363012998338806.

[41] M. Upadhyaya, S. Banert, A. B. Taylor, and P. Giselsson. "Automated tight Lyapunov analysis for first-order methods". *Mathematical Programming* **209** (2025), pp. 133–170. DOI: 10.1007/s10107-024-02061-8.

[42] B. Van Scoy, R. A. Freeman, and K. M. Lynch. "The fastest known globally convergent first-order method for minimizing strongly convex functions". *IEEE Control Systems Letters* **2**:1 (2018), pp. 49–54. DOI: 10.1109/lcsys.2017.2722406.

[43] B. Van Scoy and L. Lessard. "The speed-robustness trade-off for first-order methods with additive gradient noise" (2024). arXiv: 2109.05059v2 [math.OC].

[44] S. Zhao, L. Lessard, and M. Udell. "An automatic system to detect equivalence between iterative algorithms" (2025). arXiv: 2105.04684v4 [math.OC].

[45] K. Zhou and J. C. Doyle. *Essentials of Robust Control*. Prentice Hall international editions. Prentice Hall, Upper Saddle River, NJ, 1998. 411 pp. ISBN: 0135258332.

Paper II

AutoLyap: A Python package for computer-assisted Lyapunov analyses for first-order methods

Manu Upadhyaya Adrien B. Taylor Sebastian Banert Pontus Giselsson

Abstract

We introduce AutoLyap, a Python package designed to automate Lyapunov analyses for a wide class of first-order methods for solving structured optimization and inclusion problems. Lyapunov analyses are structured proof patterns, with historical roots in the study of dynamical systems, commonly used to establish convergence results for first-order methods. Building on previous works, the core idea behind AutoLyap is to recast the verification of the existence of a Lyapunov analysis as a semidefinite program (SDP), which can then be solved numerically using standard SDP solvers. Users of the package specify (i) the class of optimization or inclusion problems, (ii) the first-order method in question, and (iii) the type of Lyapunov analysis they wish to test. Once these inputs are provided, AutoLyap handles the SDP modeling and proceeds with the numerical solution of the SDP. We leverage the package to numerically verify and extend several convergence results.

Keywords. First-order methods, operator splitting methods, performance estimation, Lyapunov analysis, semidefinite programming

1. Introduction

Lyapunov analyses have become fundamental in establishing structured convergence guarantees for first-order optimization and operator splitting methods. Despite their theoretical strength, traditional Lyapunov analyses often demand intricate, manual derivations, limiting their accessibility. To address these challenges, this paper introduces AutoLyap, a Python package designed to automate Lyapunov analyses through semidefinite programming. AutoLyap streamlines the process of verifying and deriving convergence properties by numerically solving the associated SDP formulations. In particular, AutoLyap enables researchers and practitioners to quickly and reliably obtain convergence results for a wide range of structured optimization and inclusion problems.

The documentation of AutoLyap can be found at `https://autolyap.github.io`. The package currently relies exclusively on MOSEK [2], which provides free academic licenses, to solve semidefinite programs. Support for additional solvers may be added in future releases.

1.1 Related works

The performance estimation problem (PEP) methodology, first introduced in [14] and formalized in [39, 40], provides a systematic way to obtain unimprovable (also known as tight) performance guarantees for a large class of first-order methods. The PEP methodology involves finding a worst-case example from a predefined class of problems for the algorithm and performance measure under consideration as an optimization problem (called the performance estimation problem). This is then reformulated in a sequence of steps to arrive at a semidefinite program, whose exact solution can be recovered up to numerical precision using existing software. This can then sometimes be used to obtain closed-form convergence rates extracted from the numerical solution of the semidefinite program, or using computer algebra software.

The original PEP methodology was developed for fixed iteration counts (or horizon) $k \in \mathbb{N}$. That is, the performance estimation problem must be solved for $k = 1, 2, 3, \ldots$. This approach faces two main limitations. First, the number of variables and constraints in the semidefinite program grows quadratically with the number of iterations k, making numerical solutions prohibitive even for moderate values of k. Second, results obtained for a fixed iteration count k may provide limited insight, as they do not directly generalize or guarantee algorithmic behavior beyond this fixed horizon.

These limitations have motivated the development of Lyapunov-based approaches to analyzing the performance of first-order methods, which partially or fully overcome these limitations [30, 37, 38, 42]. The key idea is to restrict convergence-rate proof patterns to the search for key Lyapunov inequalities, which can then equivalently be reformulated as more tractable semidefinite programs, where the number of variables and constraints is either independent of the horizon k or grows only linearly with k. Compared to the standard PEP approach, this provides a more tractable framework for deriving closed-form convergence rates and producing proofs that are generally more concise and accessible. Although these types of Lyapunov analyses introduce some a priori conservatism, extracting closed-form results from the standard PEP approach can often be practically infeasible. Lyapunov-based proof patterns thus represent a pragmatic compromise with higher chances of obtaining closed-form convergence rates, while still being sufficient to yield tight convergence rates for many methods and settings. One such example is the optimized gradient method (OGM), first considered in [14], formally obtained in [21], and shown to be a worst-case optimal method for minimizing smooth and convex functions in [13]. In particular, a tight Lyapunov-based analysis is provided in [10, Section 4.3.1] for OGM.

Another closely related Lyapunov-based approach involves integral quadratic constraints (IQCs), a technique from robust control theory [28]. IQCs were first adapted for analyzing first-order methods in [25] and subsequently extended in various works [18, 24, 36, 45]. These approaches share a common feature; they represent first-order methods as linear systems interconnected through feedback with nonlinear mappings. Such representations, widely used in nonlinear systems analysis, offer compact algorithm descriptions.

The methodological developments presented in this work build on both methodologies: the worst-case analysis and tightness guarantees provided by PEP and the compact algorithm representations offered by IQCs. A first step towards combining these methodologies was taken in [38] and was further developed in [42], which serves as a foundation for further formalization, broader applicability, and accessibility through a software package.

For example, the problem class and algorithm representation considered in this paper generalize those introduced in [42, Section 2.1] and [42, Section 2.2], respectively. While the framework presented in [42] addresses convex optimization with iteration-independent algorithm parameters, the approach considered here integrates several generalizations previously studied in separate works. Specifically, we incorporate (i) optimization beyond convex settings, (ii) inclusion problems beyond optimization, (iii) iteration-dependent algorithm parameters, and (iv) non-frugal algorithms, in which basic oracle calls (such as gradient/operator or proximal/resolvent evaluations) may occur multiple times per iteration.

Moreover, the original PEP methodology is already available through the software packages PESTO [41] (Matlab) and PEPit [20] (Python). While PESTO and PEPit conveniently allow verification of Lyapunov-based analyses, AutoLyap is specifically adapted to explicitly search for such analyses.

1.2 Organization

The paper is organized as follows. Section 2 provides an introductory example demonstrating how AutoLyap can be used to derive convergence results for the Douglas–Rachford method under various scenarios, offering users a first glimpse of its functionality. Section 3 details our modeling approach for structured optimization and inclusion problems, as well as the algorithms we use to solve them. In Section 4, we outline the types of Lyapunov analyses considered. Additional numerical examples and results are presented in Section 5. The complete mathematical development underlying the tools implemented in the package is given in Section 6. Finally, Section 7 offers concluding remarks and discusses avenues for future research.

2. A first example

We start by showing how AutoLyap can be used to find linear convergence rates for the Douglas–Rachford method using a few lines of code. In particular, consider the inclusion problem

$$\text{find } y \in \mathcal{H} \ \text{ such that } \ 0 \in G_1(y) + G_2(y),$$

where $G_1 : \mathcal{H} \rightrightarrows \mathcal{H}$ is a maximally monotone operator and $G_2 : \mathcal{H} \to \mathcal{H}$ is a μ-strongly monotone and L-Lipschitz continuous operator. The Douglas–Rachford method [11, 15, 26] is given by

$$(\forall k \in \mathbb{N}_0) \quad \begin{bmatrix} y_1^k = J_{\gamma G_1}(x^k), \\ y_2^k = J_{\gamma G_2}(2y_1^k - x^k), \\ x^{k+1} = x^k + \lambda(y_2^k - y_1^k), \end{bmatrix} \tag{2.1}$$

where $J_{\gamma G_i}$ is the resolvent for G_i with step-size $\gamma \in \mathbb{R}_{++}$, $\lambda \in \mathbb{R}$ is a relaxation parameter, and $x^0 \in \mathcal{H}$ is an initial point.

The code below, using the values $(\mu, L, \gamma, \lambda) = (1, 2, 1, 2)$, performs a bisection search to find the smallest possible $\rho \in [0, 1]$ such that

$$\|y_1^k - y^\star\|^2 \in \mathcal{O}(\rho^k) \quad \text{as} \quad k \to \infty \tag{2.2}$$

provable via the Lyapunov analysis in Section 4.1 under default settings, where $y^\star \in \operatorname{zer}(G_1 + G_2)$.

```
1  import autolyap.problemclass as pc
2  import autolyap.algorithms as algs
3  from autolyap import IterationIndependent
4
5  components_list = [
6      pc.MaximallyMonotone(), # G1
7      [pc.StronglyMonotone(mu=1), pc.LipschitzOperator(L=2)] # G2
8  ]
9  problem = pc.InclusionProblem(components_list)
10 algorithm = algs.DouglasRachford(gamma=1, lambda_value=2, operator_version=True)
11 (P, T) = IterationIndependent.LinearConvergence.get_parameters_distance_to_solution(
         algorithm)
12 rho = IterationIndependent.LinearConvergence.bisection_search_rho(problem, algorithm,
        P, T)
```

Repeating this for multiple values of $\gamma \in (0, 5)$ gives Figure 1a. Updating Line 7 to

```
7      [pc.StronglyMonotone(mu=1), pc.Cocoercive(beta=0.5)] # G2
```

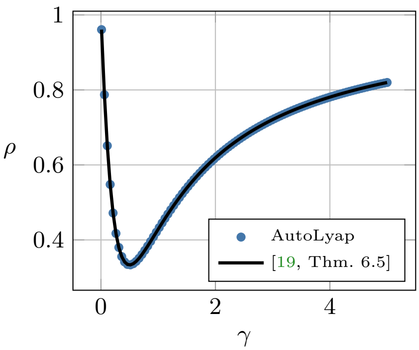
i.e., replacing L-Lipschitz continuity of G_2 with β-cocoercivity, gives Figure 1b. Instead, updating Lines 6 and 7 to

```
6      pc.Cocoercive(beta=1), # G1
7      pc.StronglyMonotone(mu=1) # G2
```

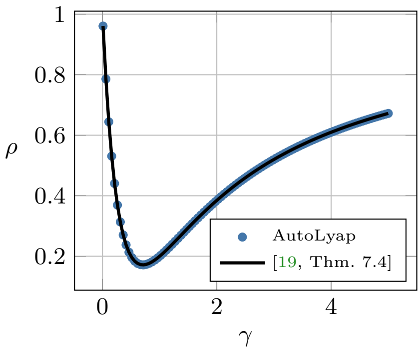
i.e., letting G_1 be β-cocoercive and G_2 be μ-strongly and maximally monotone gives Figure 1c, where we instead sweep over $\lambda \in (0, 2)$ and set $\gamma = 1$. Finally, further updating Lines 6 and 7 to

```
6      [pc.MaximallyMonotone(), pc.LipschitzOperator(L=1)], # G1
7      pc.StronglyMonotone(mu=1) # G2
```

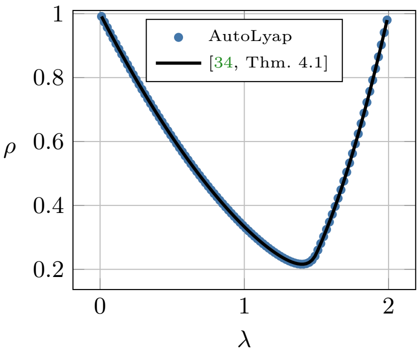
i.e., letting G_1 be monotone and L-Lipschitz continuous and G_2 be μ-strongly and maximally monotone gives Figure 1d.



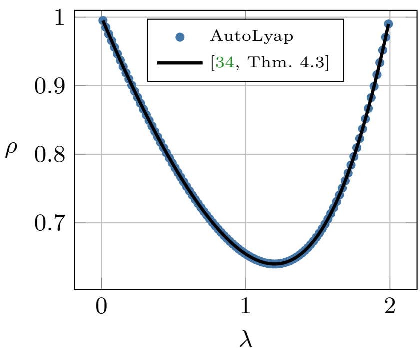
(a) $G_1 : \mathcal{H} \rightrightarrows \mathcal{H}$ maximally monotone and $G_2 : \mathcal{H} \to \mathcal{H}$ μ-strongly monotone and L-Lipschitz continuous with $(\mu, L) = (1, 2)$ and $\lambda = 2$.



(b) $G_1 : \mathcal{H} \rightrightarrows \mathcal{H}$ maximally monotone and $G_2 : \mathcal{H} \to \mathcal{H}$ μ-strongly monotone and β-cocoercive with $(\mu, \beta) = (1, 0.5)$ and $\lambda = 2$.



(c) $G_1 : \mathcal{H} \to \mathcal{H}$ β-cocoercive and $G_2 : \mathcal{H} \rightrightarrows \mathcal{H}$ μ-strongly and maximally monotone with $(\mu, \beta) = (1, 1)$ and $\gamma = 1$.



(d) $G_1 : \mathcal{H} \to \mathcal{H}$ monotone and L-Lipschitz continuous and $G_2 : \mathcal{H} \rightrightarrows \mathcal{H}$ μ-strongly and maximally monotone with $(\mu, L) = (1, 1)$ and $\gamma = 1$.

Figure 1. The linear convergence rate ρ in (2.2) for the Douglas–Rachford method (2.1), with step size γ and relaxation parameter λ, as computed by AutoLyap and compared against known tight theoretical rates.

3. Problem class and algorithm representation

In this section, we introduce the class of optimization and inclusion problems that we consider throughout the paper and the representations for algorithms that solve these problems.

3.1 Problem class

To cover both structured optimization and inclusion problems, we introduce two disjoint index sets $\mathcal{I}_{\text{func}}, \mathcal{I}_{\text{op}} \subseteq [\![1, m]\!]$, where $m \in \mathbb{N}$, such that $\mathcal{I}_{\text{func}} \cup \mathcal{I}_{\text{op}} = [\![1, m]\!]$, and consider

Table 1. Some function classes included in the `autolyap.problemclass` module of AutoLyap. See Definitions 6.1 and 6.2 for formal definitions. Further details are found in the documentation.

Class	Description
`Convex`	The class $\mathcal{F}_{0,\infty}(\mathcal{H})$.
`StronglyConvex`(μ)	The class $\mathcal{F}_{\mu,\infty}(\mathcal{H})$ for some $\mu \in \mathbb{R}_{++}$.
`WeaklyConvex`($\tilde{\mu}$)	The class $\mathcal{F}_{-\tilde{\mu},\infty}(\mathcal{H})$ for some $\tilde{\mu} \in \mathbb{R}_{++}$.
`Smooth`(L)	The class $\mathcal{F}_{-L,L}(\mathcal{H})$ for some $L \in \mathbb{R}_{++}$.
`SmoothConvex`(L)	The class $\mathcal{F}_{0,L}(\mathcal{H})$ for some $L \in \mathbb{R}_{++}$.
`SmoothStronglyConvex`(μ, L)	The class $\mathcal{F}_{\mu,L}(\mathcal{H})$ for some $\mu, L \in \mathbb{R}_{++}$ such that $\mu < L$.
`SmoothWeaklyConvex`($\tilde{\mu}, L$)	The class $\mathcal{F}_{-\tilde{\mu},L}(\mathcal{H})$ for some $\tilde{\mu}, L \in \mathbb{R}_{++}$.
`GradientDominated`(μ_{gd})	The class of function with domain $\mathcal{H}$ that are μ_{gd}-gradient dominated for some $\mu_{\mathrm{gd}} \in \mathbb{R}_{++}$. Requires that $m = 1$ and $\mathcal{I}_{\mathrm{op}} = \emptyset$ in (3.1).

inclusion problems of the form

$$\text{find } y \in \mathcal{H} \text{ such that } 0 \in \sum_{i \in \mathcal{I}_{\mathrm{func}}} \partial f_i(y) + \sum_{i \in \mathcal{I}_{\mathrm{op}}} G_i(y), \tag{3.1}$$

where the functions $f_i : \mathcal{H} \to \mathbb{R} \cup \{\pm\infty\}$ and operators $G_i : \mathcal{H} \rightrightarrows \mathcal{H}$ are chosen from some user-specified function class $\mathcal{F}_i$ and operator class $\mathcal{G}_i$, respectively, i.e.,

$$\begin{aligned} (\forall i \in \mathcal{I}_{\mathrm{func}}) \quad & f_i \in \mathcal{F}_i \subseteq \{f : \mathcal{H} \to \mathbb{R} \cup \{\pm\infty\}\}, \\ (\forall i \in \mathcal{I}_{\mathrm{op}}) \quad & G_i \in \mathcal{G}_i \subseteq \{G : \mathcal{H} \rightrightarrows \mathcal{H}\}. \end{aligned}$$

For example, if $\mathcal{I}_{\mathrm{op}} = \emptyset$, then (3.1) is a first-order optimality condition for minimizing $\sum_{i \in \mathcal{I}_{\mathrm{func}}} f_i$. Moreover, (3.1) provides a formalism that covers monotone inclusion problems [5, 35], certain equilibrium problems [6, 7, 9], so-called (mixed) variational inequalities [16, 17, 23], and beyond.

The `InclusionProblem` class from the `autolyap.problemclass` module of AutoLyap provides the interface for formulating inclusion problems (3.1). A few function and operator classes, also found in the `autolyap.problemclass` module, are presented in Tables 1 and 2, respectively. It is possible to take intersections of operator (or function) classes, as demonstrated in Section 2.

3.2 Algorithm representation

We consider first-order methods that solve (3.1) that can be represented as a discrete-time linear time-varying system in state-space form in feedback interconnection with the potentially nonlinear and set-valued operators $(\partial f_i)_{i \in \mathcal{I}_{\mathrm{func}}}$ and $(G_i)_{i \in \mathcal{I}_{\mathrm{op}}}$ that define the problem.

Before presenting the algorithm representation, we introduce some notation:

(i) $m_{\mathrm{func}} = |\mathcal{I}_{\mathrm{func}}|$ denotes the number of functional components in (3.1);

(ii) $m_{\mathrm{op}} = |\mathcal{I}_{\mathrm{op}}|$ denotes the number of operator components in (3.1);

(iii) $\bar{m}_i \in \mathbb{N}$ denotes the number of evaluations of ∂f_i per iteration, for $i \in \mathcal{I}_{\mathrm{func}}$;

Table 2. Some operator classes included in the `autolyap.problemclass` module of AutoLyap. See Definitions 6.4 and 6.6 for formal definitions. Further details are found in the documentation.

Class	Description
`MaximallyMonotone`	Class of operators $G : \mathcal{H} \rightrightarrows \mathcal{H}$ that are maximally monotone.
`StronglyMonotone`(μ)	Class of operators $G : \mathcal{H} \rightrightarrows \mathcal{H}$ that are μ-strongly and maximally monotone for some $\mu \in \mathbb{R}_{++}$.
`LipschitzOperator`(L)	Class of operators $G : \mathcal{H} \to \mathcal{H}$ that are L-Lipschitz continuous for some $L \in \mathbb{R}_{++}$.
`Cocoercive`(β)	Class of operators $G : \mathcal{H} \to \mathcal{H}$ that are β-cocoercive for some $\beta \in \mathbb{R}_{++}$.

(iv) $\bar{m}_i \in \mathbb{N}$ denotes the number of evaluations of G_i per iteration, for $i \in \mathcal{I}_{\text{op}}$;

(v) $\bar{m}_{\text{func}} = \sum_{i\in\mathcal{I}_{\text{func}}} \bar{m}_i$ denotes the total number of subdifferential evaluations per iteration;

(vi) $\bar{m}_{\text{op}} = \sum_{i\in\mathcal{I}_{\text{op}}} \bar{m}_i$ denotes the total number of operator evaluations per iteration; and

(vii) $\bar{m} = \bar{m}_{\text{func}} + \bar{m}_{\text{op}}$ denotes the combined total number of evaluations per iteration.

Since we consider methods that allow for multiple evaluation of $(\partial f_i)_{i\in\mathcal{I}_{\text{func}}}$ and $(G_i)_{i\in\mathcal{I}_{\text{op}}}$ per iteration, we define $\boldsymbol{f}_i : \mathcal{H}^{\bar{m}_i} \to (\mathbb{R} \cup \{\pm\infty\})^{\bar{m}_i}$ such that

$$\left(\begin{array}{c}\forall i \in \mathcal{I}_{\text{func}} \\ \forall \boldsymbol{y}_i = (y_{i,1}, \ldots, y_{i,\bar{m}_i}) \in \mathcal{H}^{\bar{m}_i}\end{array}\right) \quad \boldsymbol{f}_i(\boldsymbol{y}_i) = (f_i(y_{i,1}), \ldots, f_i(y_{i,\bar{m}_i})),$$

$\boldsymbol{\partial f}_i : \mathcal{H}^{\bar{m}_i} \rightrightarrows \mathcal{H}^{\bar{m}_i}$ such that[1]

$$\left(\begin{array}{c}\forall i \in \mathcal{I}_{\text{func}} \\ \forall \boldsymbol{y}_i = (y_{i,1}, \ldots, y_{i,\bar{m}_i}) \in \mathcal{H}^{\bar{m}_i}\end{array}\right) \quad \boldsymbol{\partial f}_i(\boldsymbol{y}_i) = \prod_{j=1}^{\bar{m}_i} \partial f_i(y_{i,j}),$$

and $\boldsymbol{G}_i : \mathcal{H}^{\bar{m}_i} \rightrightarrows \mathcal{H}^{\bar{m}_i}$ such that

$$\left(\begin{array}{c}\forall i \in \mathcal{I}_{\text{op}} \\ \forall \boldsymbol{y}_i = (y_{i,1}, \ldots, y_{i,\bar{m}_i}) \in \mathcal{H}^{\bar{m}_i}\end{array}\right) \quad \boldsymbol{G}_i(\boldsymbol{y}_i) = \prod_{j=1}^{\bar{m}_i} G_i(y_{i,j}).$$

We are now ready to give the algorithm representation: Pick an initial $\boldsymbol{x}_0 \in \mathcal{H}^n$, an

[1] In this context, the symbol Π is used for Cartesian products.

iteration horizon $K \in \mathbb{N}_0 \cup \{\infty\}$, and let

$$(\forall k \in [\![0, K]\!]) \quad \begin{bmatrix} \boldsymbol{x}^{k+1} = (A_k \otimes \mathrm{Id})\boldsymbol{x}^k + (B_k \otimes \mathrm{Id})\boldsymbol{u}^k, \\ \boldsymbol{y}^k = (C_k \otimes \mathrm{Id})\boldsymbol{x}^k + (D_k \otimes \mathrm{Id})\boldsymbol{u}^k, \\ (\boldsymbol{u}_i^k)_{i \in \mathcal{I}_{\text{func}}} \in \prod_{i \in \mathcal{I}_{\text{func}}} \boldsymbol{\partial} \boldsymbol{f}_i(\boldsymbol{y}_i^k), \\ (\boldsymbol{u}_i^k)_{i \in \mathcal{I}_{\text{op}}} \in \prod_{i \in \mathcal{I}_{\text{op}}} \boldsymbol{G}_i(\boldsymbol{y}_i^k), \\ \boldsymbol{F}^k = (\boldsymbol{f}_i(\boldsymbol{y}_i^k))_{i \in \mathcal{I}_{\text{func}}}, \end{bmatrix} \tag{3.2}$$

where $\boldsymbol{x}^k \in \mathcal{H}^n$, $\boldsymbol{u}^k = (\boldsymbol{u}_1^k, \ldots, \boldsymbol{u}_m^k) \in \prod_{i=1}^m \mathcal{H}^{\bar{m}_i}$, $\boldsymbol{y}^k = (\boldsymbol{y}_1^k, \ldots, \boldsymbol{y}_m^k) \in \prod_{i=1}^m \mathcal{H}^{\bar{m}_i}$, and $\boldsymbol{F}^k \in \mathbb{R}^{\bar{m}_{\text{func}}}$ are the algorithm variables and

$$A_k \in \mathbb{R}^{n \times n}, \qquad B_k \in \mathbb{R}^{n \times \bar{m}}, \qquad C_k \in \mathbb{R}^{\bar{m} \times n}, \qquad D_k \in \mathbb{R}^{\bar{m} \times \bar{m}} \tag{3.3}$$

are matrices containing the parameters of the method at hand.

The interface for (3.2) in AutoLyap is provided via the abstract base class `Algorithm`, located in the `autolyap.algorithms` submodule. Specifically, each algorithm in AutoLyap must be implemented as a concrete subclass of `Algorithm`, and must define the abstract method `get_ABCD`, as shown below.

```python
from abc import ABC, abstractmethod
import numpy as np
from typing import List, Tuple

class Algorithm(ABC):
    def __init__(self, n: int, m: int, m_bar_is: List[int], I_func: List[int], I_op:
        List[int]):
        # n: dimension n of x^k in (3.2)
        # m: number of components m in (3.1)
        # m_bar_is: list of (m̄_i)_{i=1}^m in (3.2)
        # I_func: index set I_func in (3.1)
        # I_op: index set I_op in (3.1)
        # ... (omitted for brevity) ...

    @abstractmethod
    def get_ABCD(self, k: int) -> Tuple[np.ndarray, np.ndarray, np.ndarray, np.
        ndarray]:
        # Return the system matrices (A_k, B_k, C_k, D_k) for
        # iteration k in (3.2).
        pass
```

For a concrete example, see Section 5.1, which demonstrates the gradient method implemented as a concrete subclass.

We conclude this section by introducing a well-posedness assumption, which holds for all practical algorithms, and ensures that the subsequent theoretical results are properly formulated.

Assumption 3.1 (Well-posedness)
We assume that $((A_k, B_k, C_k, D_k))_{k=0}^K$ *is chosen such that there exists a sequence* $((\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{y}^k, \boldsymbol{F}^k))_{k=0}^K$ *satisfying* (3.2) *for each* $\boldsymbol{x}_0 \in \mathcal{H}^n$, *for each* $(f_i)_{i \in \mathcal{I}_{\text{func}}} \in \prod_{i \in \mathcal{I}_{\text{func}}} \mathcal{F}_i$, *and for each* $(G_i)_{i \in \mathcal{I}_{\text{op}}} \in \prod_{i \in \mathcal{I}_{\text{op}}} \mathcal{G}_i$.

4. The analysis tools

In this section, we introduce two classes of Lyapunov analyses. The first class comprises stationary, or iteration-independent, analyses, while the second involves nonstationary, or iteration-dependent, analyses. These analyses utilize quadratic ansatzes, facilitating verification through semidefinite programs, as detailed in Sections 6.3 and 6.4, respectively.

4.1 Iteration-independent Lyapunov analyses

In this case, we consider algorithms that continue to iterate indefinitely and have iteration-independent parameters. I.e., in (3.2), we assume that $K = \infty$ and that there exist fixed matrices

$$(A, B, C, D) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times \bar{m}} \times \mathbb{R}^{\bar{m}\times n} \times \mathbb{R}^{\bar{m}\times \bar{m}}$$

such that

$$(\forall k \in \mathbb{N}_0) \quad (A_k, B_k, C_k, D_k) = (A, B, C, D). \tag{4.1}$$

We are interested in Lyapunov analyses that may depend on a solution to the inclusion problem (3.1). In particular, without loss of generality and for computational efficiency (which will be clear later), we will consider the variables

$$(y^\star, \hat{\boldsymbol{u}}^\star, \boldsymbol{F}^\star) \in \left\{ \underbrace{(y, \hat{\boldsymbol{u}}, \boldsymbol{F})}_{\in \mathcal{H}\times\mathcal{H}^{m-1}\times\mathbb{R}^{m_{\text{func}}}} \;\middle|\; \begin{array}{l} (u_i)_{i\in\mathcal{I}_{\text{func}}} \in \prod_{i\in\mathcal{I}_{\text{func}}} \partial f_i(y), \\ (u_i)_{i\in\mathcal{I}_{\text{op}}} \in \prod_{i\in\mathcal{I}_{\text{op}}} G_i(y), \\ \sum_{i=1}^{m} u_i = 0, \\ \hat{\boldsymbol{u}} = (u_1, \ldots, u_{m-1}), \\ \boldsymbol{F} = (\boldsymbol{f}_i(y))_{i\in\mathcal{I}_{\text{func}}} \end{array} \right\}, \tag{4.2}$$

where $\hat{\boldsymbol{u}}^\star$ is void when $m = 1$. For example, it is clear that $y^\star$ in (4.2) is a solution to the inclusion problem (3.1).

Definition 4.1

Suppose that Assumption 3.1 and (4.1) hold, $\boldsymbol{x}_0 \in \mathcal{H}^n$ is an initial point, $((\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{y}^k, \boldsymbol{F}^k))_{k\in\mathbb{N}_0}$ is a sequence of iterates satisfying (3.2), $(y^\star, \hat{\boldsymbol{u}}^\star, \boldsymbol{F}^\star)$ is a point satisfying (4.2), $\rho \in [0, 1]$ is a contraction factor, $h \in \mathbb{N}_0$ is a history parameter, and $\alpha \in \mathbb{N}_0$ is an overlap parameter. Define

$$\begin{aligned}(\forall k \in \mathbb{N}_0) \quad \mathcal{V}(W, w, k) = {} & \mathcal{Q}(W, (\boldsymbol{x}^k, \boldsymbol{u}^k, \ldots, \boldsymbol{u}^{k+h}, \hat{\boldsymbol{u}}^\star, y^\star)) \\ & + w^\top(\boldsymbol{F}^k, \ldots, \boldsymbol{F}^{k+h}, \boldsymbol{F}^\star),\end{aligned} \tag{4.3}$$

for each $(W, w) \in \{(Q, q), (P, p)\}$, where $Q, P \in \mathbb{S}^{n+(h+1)\bar{m}+m}$ and $q, p \in \mathbb{R}^{(h+1)\bar{m}_{\text{func}}+m_{\text{func}}}$, and define

$$\begin{aligned}(\forall k \in \mathbb{N}_0) \quad \mathcal{R}(W, w, k) = {} & \mathcal{Q}(W, (\boldsymbol{x}^k, \boldsymbol{u}^k, \ldots, \boldsymbol{u}^{k+h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\ & + w^\top(\boldsymbol{F}^k, \ldots, \boldsymbol{F}^{k+h+\alpha+1}, \boldsymbol{F}^\star),\end{aligned} \tag{4.4}$$

for each $(W, w) \in \{(S, s), (T, t)\}$*, where* $S, T \in \mathbb{S}^{n+(h+\alpha+2)\bar{m}+m}$ *and* $s, t \in \mathbb{R}^{(h+\alpha+2)\bar{m}_{\text{func}}+m_{\text{func}}}$*. We say that* (Q, q, S, s) *satisfies a* $(P, p, T, t, \rho, h, \alpha)$-quadratic Lyapunov inequality *for algorithm* (3.2) *over the problem class defined by* $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ *and* $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ *if*

$$(\forall k \in \mathbb{N}_0) \quad \mathcal{V}(Q, q, k + \alpha + 1) \le \rho \mathcal{V}(Q, q, k) - \mathcal{R}(S, s, k), \tag{C1}$$
$$(\forall k \in \mathbb{N}_0) \quad \mathcal{V}(Q, q, k) \ge \mathcal{V}(P, p, k) \ge 0, \tag{C2}$$
$$(\forall k \in \mathbb{N}_0) \quad \mathcal{R}(S, s, k) \ge \mathcal{R}(T, t, k) \ge 0, \tag{C3}$$
$$(\forall k \in \mathbb{N}_0) \quad \mathcal{R}(S, s, k + 1) \le \mathcal{R}(S, s, k), \tag{C4}$$

hold for each initial point $\boldsymbol{x}_0$*, for each sequence of iterates* $((\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{y}^k, \boldsymbol{F}^k))_{k\in\mathbb{N}_0}$*, for each point* $(y^\star, \hat{\boldsymbol{u}}^\star, \boldsymbol{F}^\star)$*, for each* $(f_i)_{i\in\mathcal{I}_{\text{func}}} \in \prod_{i\in\mathcal{I}_{\text{func}}} \mathcal{F}_i$*, and for each* $(G_i)_{i\in\mathcal{I}_{\text{op}}} \in \prod_{i\in\mathcal{I}_{\text{op}}} \mathcal{G}_i$*, where* (C4) *is an optional requirement that may be removed.*

In the proposed methodology, the user specifies $(P, p, T, t, \rho, h, \alpha)$ and the methodology searches for (Q, q, S, s) complying with (C1)-(C3) (and optionally (C4)), if it exists. When such a (Q, q, S, s) exists, the choice of $(P, p, T, t, \rho, h, \alpha)$ decides which convergence properties Definition 4.1 implies.

- If $\rho \in [0, 1[$, then

$$0 \le \mathcal{V}(P, p, k) \le \mathcal{V}(Q, q, k) \le \rho^{\lfloor k/(\alpha+1)\rfloor} \max_{i\in[\![0,\alpha]\!]} \mathcal{V}(Q, q, i) \xrightarrow[k\to\infty]{} 0,$$

i.e., $(\mathcal{V}(P, p, k))_{k\in\mathbb{N}_0}$ converges $\sqrt[\alpha+1]{\rho}$-linearly to zero.

- If $\rho = 1$, then

$$(\forall k \in \mathbb{N}_0) \quad \sum_{i=0}^{k} \mathcal{R}(T, t, i) \le \sum_{i=0}^{k} \mathcal{R}(S, s, i) \le \sum_{j=0}^{\alpha} \mathcal{V}(Q, q, j),$$

using a telescoping summation argument. In particular, $(\mathcal{R}(T, t, k))_{k\in\mathbb{N}_0}$ is summable, converges to zero, and e.g., $\max_{i\in[\![0,k]\!]} \mathcal{R}(T, t, i) \in \mathcal{O}(1/k)$ as $k \to \infty$. If the optional requirement (C4) holds, we conclude the stronger last-iterate convergence result $\mathcal{R}(T, t, k) \in o(1/k)$ as $k \to \infty$.

The SDP formulation of Definition 4.1 is provided in Section 6.3. An interface for these types of analysis is provided in the class `IterationIndependent` in AutoLyap, shown below.

```
from typing import Type, Optional

class IterationIndependent:

    # Class attributes explained below
    LinearConvergence = LinearConvergence
    SublinearConvergence = SublinearConvergence

    @staticmethod
    def verify_iteration_independent_Lyapunov(
            prob: Type[InclusionProblem],
            algo: Type[Algorithm],
            P: np.ndarray,
            T: np.ndarray,
```

```
            p: Optional[np.ndarray] = None,
            t: Optional[np.ndarray] = None,
            rho: float = 1.0,
            h: int = 0,
            alpha: int = 0,
            Q_equals_P: bool = False,
            S_equals_T: bool = False,
            q_equals_p: bool = False,
            s_equals_t: bool = False,
            remove_C2: bool = False,
            remove_C3: bool = False,
            remove_C4: bool = True
        ) -> bool:
        # prob: An InclusionProblem instance.
        # algo: An Algorithm instance.
        # P: The user-specified matrix $P$ in Definition 4.1.
        # T: The user-specified matrix $T$ in Definition 4.1.
        # p: The user-specified vector $p$ in Definition 4.1.
        #    Required when $\mathcal{I}_{\text{func}} \neq \emptyset$.
        # t: The user-specified vector $t$ in Definition 4.1.
        #    Required when $\mathcal{I}_{\text{func}} \neq \emptyset$.
        # rho: Contraction factor $\rho$ in Definition 4.1.
        # h: History parameter $h$ in Definition 4.1.
        # alpha: Overlap parameter $\alpha$ in Definition 4.1.
        # Q_equals_P: If True, sets $Q$ equal to $P$ in
        # Definition 4.1.
        # S_equals_T: If True, sets $S$ equal to $T$ in
        # Definition 4.1.
        # q_equals_p: If True, sets $q$ equal to $p$ in
        # Definition 4.1.
        # s_equals_t: If True, sets $s$ equal to $t$ in
        # Definition 4.1.
        # remove_C2: Flag to remove (C2) in Definition 4.1.
        # remove_C3: Flag to remove (C3) in Definition 4.1.
        # remove_C4: Flag to remove (C4) in Definition 4.1.

        # Returns True if the SDP is solved successfully
        # (which implies that a Lyapunov inequality in
        # the sense of Definition 4.1 exists), False otherwise.
        # ... (omitted for brevity) ...
```

The class attributes `LinearConvergence` and `SublinearConvergence` are static classes that contain helper functions for constructing appropriate parameters (P, p, T, t), with additional details given in Sections 6.3.1 and 6.3.2, respectively. Moreover, the class `LinearConvergence` has a helper function `bisection_search_rho` that performs a bisection search to find the smallest $\rho \in [0, 1]$ in Definition 4.1 via `IterationIndependent.verify_iteration_independent_Lyapunov` with tolerance `tol`, as shown below.

```
class LinearConvergence:
    @staticmethod
    def bisection_search_rho(
            prob: Type[InclusionProblem],
            algo: Type[Algorithm],
            P: np.ndarray,
            T: np.ndarray,
            p: Optional[np.ndarray] = None,
            t: Optional[np.ndarray] = None,
            h: int = 0,
            alpha: int = 0,
            Q_equals_P: bool = False,
            S_equals_T: bool = False,
```

```
            q_equals_p: bool = False,
            s_equals_t: bool = False,
            remove_C2: bool = False,
            remove_C3: bool = False,
            remove_C4: bool = True,
            lower_bound: float = 0.0,
            upper_bound: float = 1.0,
            tol: float = 1e-12
        ) -> Optional[float]:
        # ...  (omitted for brevity) ...
```

4.2 Iteration-dependent Lyapunov analyses

In this section, we focus on finite-horizon analyses, which apply to algorithms that run indefinitely as well as algorithms with finite iteration budgets matching or exceeding the horizon. For notational simplicity, we assume that $K \in \mathbb{N}$ in (3.2), representing the horizon. Following [37], we adopt an ansatz consisting of a sequence of iteration-dependent and quadratic Lyapunov functions, also often referred to as *potential functions* [3].

Definition 4.2

Suppose that Assumption 3.1 holds, $\boldsymbol{x}_0 \in \mathcal{H}^n$ is an initial point, $((\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{y}^k, \boldsymbol{F}^k))_{k=0}^K$ is a sequence of iterates satisfying (3.2), $(y^\star, \hat{\boldsymbol{u}}^\star, \boldsymbol{F}^\star)$ is a point satisfying (4.2), and $c \in \mathbb{R}_+$. Define

$$(\forall k \in [\![0, K]\!]) \quad \begin{cases} \mathcal{V}(k) = \mathcal{Q}(Q_k, (\boldsymbol{x}^k, \boldsymbol{u}^k, \hat{\boldsymbol{u}}^\star, y^\star)) + q_k^\top(\boldsymbol{F}^k, \boldsymbol{F}^\star), \\ Q_k \in \mathbb{S}^{n+\bar{m}+m}, \\ q_k \in \mathbb{R}^{\bar{m}_{\text{func}}+m_{\text{func}}}. \end{cases} \tag{4.5}$$

We say that $((Q_k, q_k))_{k=0}^K$ and c satisfy a length K sequence of chained Lyapunov inequalities *for algorithm (3.2) over the problem class defined by $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ and $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ if*

$$\mathcal{V}(K) \leq \mathcal{V}(K-1) \leq \ldots \leq \mathcal{V}(1) \leq c\mathcal{V}(0) \tag{4.6}$$

holds for each initial point $\boldsymbol{x}_0$, for each sequence of iterates $((\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{y}^k, \boldsymbol{F}^k))_{k=0}^K$, for each point $(y^\star, \hat{\boldsymbol{u}}^\star, \boldsymbol{F}^\star)$, for each $(f_i)_{i\in\mathcal{I}_{\text{func}}} \in \prod_{i\in\mathcal{I}_{\text{func}}} \mathcal{F}_i$, and for each $(G_i)_{i\in\mathcal{I}_{\text{op}}} \in \prod_{i\in\mathcal{I}_{\text{op}}} \mathcal{G}_i$.

In the proposed methodology, the user specifies (Q_0, q_0, Q_K, q_K) and the methodology searches for $((Q_k, q_k))_{k=1}^{K-1}$ and a minimal c complying with (4.6), if they exist. In particular, the user specifies the initial Lyapunov function $\mathcal{V}(0)$, final Lyapunov function $\mathcal{V}(K)$, and if (4.6) holds, we can conclude that

$$\mathcal{V}(K) \leq c\mathcal{V}(0).$$

The SDP formulation of Definition 4.2 is provided in Section 6.4. However, we would like to highlight a computational aspect in this case. Each inequality in (4.6) is verified via a so-called one-step analysis, leading to a system of positive semidefinite constraints of constant dimension. In particular, the number of variables and constraints grows linearly with K. Moreover, similar to Definition 4.1, it is possible to introduce history and overlap parameters in Definition 4.2. However, we opt not to include this extension in this presentation in favor of readability.

An interface for the analysis in Definition 4.2 is provided in the class `IterationDependent` in AutoLyap, which is shown below.

```python
class IterationDependent:
    @staticmethod
    def verify_iteration_dependent_Lyapunov(
            prob: Type[InclusionProblem],
            algo: Type[Algorithm],
            K: int,
            Q_0: np.ndarray,
            Q_K: np.ndarray,
            q_0: Optional[np.ndarray] = None,
            q_K: Optional[np.ndarray] = None
        ) -> Tuple[bool, Optional[float]]:
        # prob: An InclusionProblem instance.
        # algo: An Algorithm instance.
        # K: The user-specified horizon K in Definition 4.2.
        # Q_0: The user-specified matrix Q_0 in Definition 4.2.
        # Q_K: The user-specified matrix Q_K in Definition 4.2.
        # q_0: The user-specified matrix q_0 in Definition 4.2.
        #      Required when I_func ≠ ∅.
        # q_K: The user-specified matrix q_K in Definition 4.2.
        #      Required when I_func ≠ ∅.

        # Returns a tuple (True, c), where c is c in
        # Definition 4.2, if the SDP is solved successfully, or
        # (False, None) otherwise.
        # ... (omitted for brevity) ...
```

The class `IterationDependent` also contains helper functions for selecting appropriate parameters (Q_0, q_0, Q_K, q_K), with details given in Section 6.4.1.

5. Additional examples and numerical results

This section presents examples demonstrating the use of AutoLyap for convergence analysis across various methods and settings. The first example explicitly illustrates how to translate a first-order method written in standard form into the algorithm representation given by equation (3.2), as well as how this translation is implemented in AutoLyap. For brevity, this explicit translation is omitted in most subsequent examples, as it can easily be done analogously (see [42, Section 2.5] for numerous examples detailing the transition from standard form to the algorithm representation (3.2)).

5.1 Gradient method on a nonconvex problem

Consider the optimization problem

$$\underset{y\in\mathcal{H}}{\text{minimize}} \ f(y) \tag{5.1}$$

where $f : \mathcal{H} \to \mathbb{R}$ is μ_{gd}-gradient dominated and L-smooth for some $\mu_{\text{gd}}, L \in \mathbb{R}_{++}$ such that $\mu_{\text{gd}} \leq L$ (see Definition 6.1). In particular, (5.1) is not necessarily a convex optimization problem. Note that $0 = \nabla f(y)$, or equivalently, $0 \in \partial f(y) = \{\nabla f(y)\}$, is an optimality condition for (5.1), and fits (3.1).

For the gradient method, i.e.,

$$(\forall k \in \mathbb{N}_0) \quad x^{k+1} = x^k - \gamma \nabla f(x^k) \tag{5.2}$$

for an initial point $x^0 \in \mathcal{H}$ and step size $\gamma \in (0, 2/L)$, it is known that $(f(x^k) - f(y^\star))_{k\in\mathbb{N}_0}$ converges to zero linearly for each $y^\star \in \operatorname{Argmin}_{y\in\mathcal{H}} f(y)$. Let us find the rate using AutoLyap.

The first step is to write (5.2) in the algorithm representation (3.2). Direct inspection gives

$$(\forall k \in \mathbb{N}_0) \qquad \begin{cases} \boldsymbol{x}^{k+1} = \big([1] \otimes \mathrm{Id}\big)\boldsymbol{x}^k + \big([-\gamma] \otimes \mathrm{Id}\big)\boldsymbol{u}^k, \\ \boldsymbol{y}^k = \big([1] \otimes \mathrm{Id}\big)\boldsymbol{x}^k + \big([0] \otimes \mathrm{Id}\big)\boldsymbol{u}^k, \\ \boldsymbol{u}^k \in \boldsymbol{\partial f}(\boldsymbol{y}^k), \end{cases} \tag{5.3}$$

where $\boldsymbol{x}^k = x^k$, $\boldsymbol{y}^k = x^k$, $\boldsymbol{u}^k = \nabla f(x^k)$, and $\boldsymbol{\partial f}(\boldsymbol{y}) = \{\nabla f(\boldsymbol{y})\}$ for each $\boldsymbol{y} \in \mathcal{H}$. The implementation of the gradient method (5.3) in AutoLyap, also found in the `autolyap.algorithms` submodule, is shown below.

```
from autolyap.algorithms import Algorithm

class GradientMethod(Algorithm):
    def __init__(self, gamma):
        super().__init__(n=1, m=1, m_bar_is=[1], I_func=[1], I_op=[])
        # n: dimension n of x^k in (3.2)
        # m: number of components m in (3.1)
        # m_bar_is: list of (m̄_i)_{i=1}^m in (3.2)
        # I_func: index set I_func in (3.1)
        # I_op: index set I_op in (3.1)
        self.gamma = gamma

    def get_ABCD(self, k: int):
        A = np.array([[1]])
        B = np.array([[-self.gamma]])
        C = np.array([[1]])
        D = np.array([[0]])
        return (A, B, C, D)
```

The code below, using the values $(\mu_{\mathrm{gd}}, L, \gamma) = (0.5, 1, 1)$, performs a bisection search to find the smallest possible $\rho \in [0, 1]$ such that

$$f(x^k) - f(y^\star) \in \mathcal{O}(\rho^k) \quad \text{as} \quad k \to \infty \tag{5.4}$$

provable via the Lyapunov analysis in Section 4.1 under default settings, where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$.

```
components_list = [
    [pc.GradientDominated(mu_gd=0.5), pc.Smooth(L=1)], # f in (5.1)
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.GradientMethod(gamma=1)
(P, p, T, t) = IterationIndependent.LinearConvergence.
    get_parameters_function_value_suboptimality(algorithm)
rho = IterationIndependent.LinearConvergence.bisection_search_rho(problem, algorithm,
    P, T, p=p, t=t)
```

Repeating this for multiple values of $\gamma \in (0, 2)$ gives Figure 2. Numerical evidence suggests that AutoLyap improves the rate in [1, Theorem 3] for γ in the interval $(1.74, 2)$. Indeed, a careful inspection shows that the proof of [1, Theorem 3] does not use interpolation conditions with respect to solutions $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$ when going from (7) to (10) in [1], which is one source of a priori conservatism.

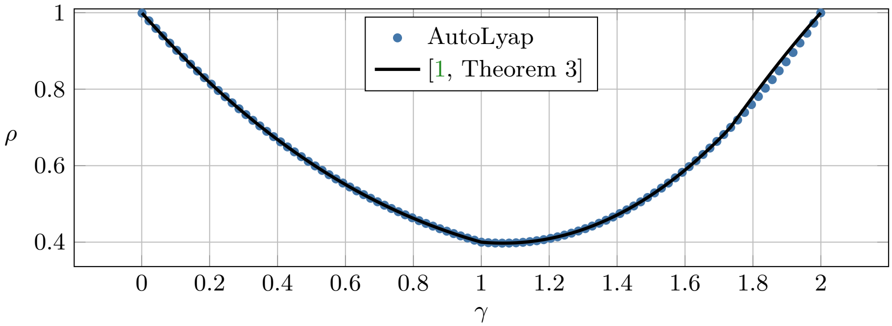


Figure 2. The linear convergence rate ρ in (5.4) for the gradient method (5.2), with step size γ and applied to optimization problem (5.1) with a 0.5-gradient dominated and 1-smooth objective function, as computed by AutoLyap and compared against known theoretical rates.

5.2 Heavy-ball method

Let $f \in \mathcal{F}_{0,L}(\mathcal{H})$ for some $L \in \mathbb{R}_{++}$ and consider the heavy-ball method, i.e.,

$$(\forall k \in \mathbb{N}_0) \quad x^{k+1} = x^k - \gamma \nabla f(x^k) + \delta(x^k - x^{k-1}) \tag{5.5}$$

for some initial point $x^0 \in \mathcal{H}$, step size $\gamma \in \mathbb{R}_{++}$, and momentum parameter $\delta \in \mathbb{R}$. The code below, using the values $(L, \gamma, \delta) = (1, 1, 0.5)$, checks whether

$$f(x^k) - f(y^\star) \in o(1/k) \quad \text{as} \quad k \to \infty$$

where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$, via the Lyapunov analysis in Section 4.1 by utilizing (C4).

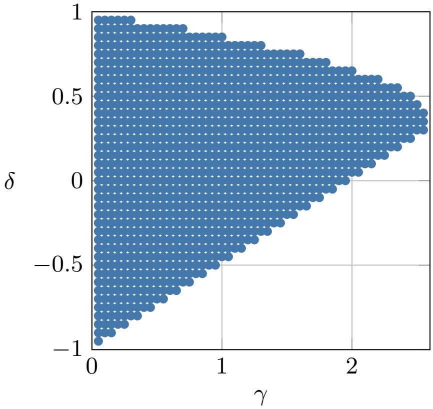
```
components_list = [
    pc.SmoothConvex(L=1) # f
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.HeavyBallMethod(gamma=1, delta=0.5)
(P, p, T, t) = IterationIndependent.SublinearConvergence.
    get_parameters_function_value_suboptimality(algorithm)
successful = IterationIndependent.verify_iteration_independent_Lyapunov(problem,
    algorithm, P, T, p=p, t=t, rho=1, remove_C4=False)
```

In this case, the verification is successful, i.e., `successful` is `True`. Repeating this for values in $\gamma \in (0, 3)$ and $\delta \in (-1, 1)$ gives Figure 3a.

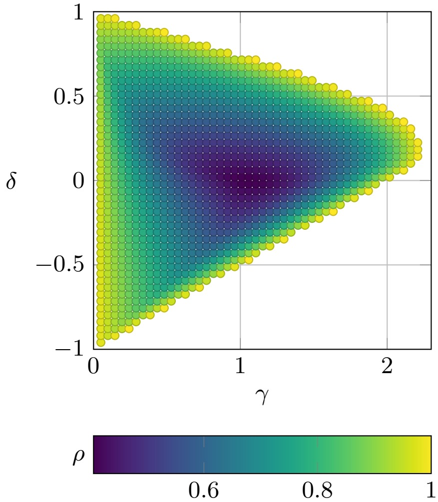
Suppose instead that the heavy-ball method (5.5) is applied to a function $f : \mathcal{H} \to \mathbb{R}$ that is μ_{gd}-gradient dominated and L-smooth for some $\mu_{\mathrm{gd}}, L \in \mathbb{R}_{++}$ such that $\mu_{\mathrm{gd}} \leq L$. The code below, using the values $(\mu_{\mathrm{gd}}, L, \gamma, \delta) = (0.5, 1, 1, 0.5)$, performs a bisection search to find the smallest possible $\rho \in [0, 1]$ such that

$$f(x^k) - f(y^\star) \in \mathcal{O}(\rho^k) \quad \text{as} \quad k \to \infty \tag{5.6}$$

provable via the Lyapunov analysis in Section 4.1 under default settings, where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$.



(a) Verification of $f(x^k) - f(y^\star) \in o(1/k)$ as $k \to \infty$ where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$ when $f \in \mathcal{F}_{0,1}(\mathcal{H})$.



(b) The linear convergence rate ρ in (5.6) when f is 0.5-gradient dominated and 1-smooth.

Figure 3. Convergence regions and rates for the heavy-ball method (5.5) with step size γ and momentum parameter δ using AutoLyap.

```
components_list = [
    [pc.GradientDominated(mu_gd=0.5), pc.Smooth(L=1)], # f
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.HeavyBallMethod(gamma=1, delta=0.5)
(P, p, T, t) = IterationIndependent.LinearConvergence.
     get_parameters_function_value_suboptimality(algorithm)
rho = IterationIndependent.LinearConvergence.bisection_search_rho(problem, algorithm,
      P, T, p=p, t=t)
```

Repeating this for multiple values of $\gamma \in (0, 3)$ and $\delta \in (-1, 1)$ gives Figure 3b.

5.3 Gradient method with constant Nesterov momentum

Let $f \in \mathcal{F}_{0,L}(\mathcal{H})$ for some $L \in \mathbb{R}_{++}$ and consider the gradient method with constant Nesterov momentum, i.e.,

$$(\forall k \in \mathbb{N}_0) \quad \left[\begin{aligned} y^k &= x^k + \delta(x^k - x^{k-1}), \\ x^{k+1} &= y^k - \gamma \nabla f(y^k) \end{aligned} \right. \tag{5.7}$$

for some initial points $x^{-1}, x^0 \in \mathcal{H}$, step size $\gamma \in \mathbb{R}_{++}$, and momentum parameter $\delta \in \mathbb{R}$. The code below, using the values $(L, \gamma, \delta) = (1, 1, 0.5)$, checks whether

$$f(x^k) - f(y^\star) \in o(1/k) \quad \text{as} \quad k \to \infty$$

where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$, via the Lyapunov analysis in Section 4.1 by utilizing (C4).

```
components_list = [
    pc.SmoothConvex(L=1) # f
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.GradientNesterovMomentum(gamma=1, delta=0.5)
(P, p, T, t) = IterationIndependent.SublinearConvergence.
      get_parameters_function_value_suboptimality(algorithm)
successful = IterationIndependent.verify_iteration_independent_Lyapunov(problem,
      algorithm, P, T, p=p, t=t, rho=1, remove_C4=False)
```

In this case, the verification is successful, i.e., successful is True. Repeating this for values in $\gamma \in (0,4)$ and $\delta \in (-1,1)$ gives Figure 4a.

Suppose instead that the gradient method with Nesterov momentum (5.7) is applied to a function $f : \mathcal{H} \to \mathbb{R}$ that is μ_{gd}-gradient dominated and L-smooth for some $\mu_{\mathrm{gd}}, L \in \mathbb{R}_{++}$ such that $\mu_{\mathrm{gd}} \leq L$. The code below, using the values $(\mu_{\mathrm{gd}}, L, \gamma, \delta) = (0.5, 1, 1, 0.5)$, performs a bisection search to find the smallest possible $\rho \in [0,1]$ such that

$$f(x^k) - f(y^\star) \in \mathcal{O}(\rho^k) \quad \text{as} \quad k \to \infty \tag{5.8}$$

provable via the Lyapunov analysis in Section 4.1 under default settings, where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$.

```
components_list = [
    [pc.GradientDominated(mu_gd=0.5), pc.Smooth(L=1)], # f
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.GradientNesterovMomentum(gamma=1, delta=0.5)
(P, p, T, t) = IterationIndependent.LinearConvergence.
      get_parameters_function_value_suboptimality(algorithm)
rho = IterationIndependent.LinearConvergence.bisection_search_rho(problem, algorithm,
       P, T, p=p, t=t)
```

Repeating this for multiple values of $\gamma \in (0,4)$ and $\delta \in (-1,1)$ gives Figure 4b.

5.4 Chambolle–Pock

Suppose that $f_1, f_2 \in \mathcal{F}_{0,\infty}(\mathcal{H})$. The Chambolle–Pock method [8] (in the special case when the linear operator is the identity mapping) solves the minimization problem

$$\underset{y \in \mathcal{H}}{\text{minimize}} \;\; f_1(y) + f_2(y)$$

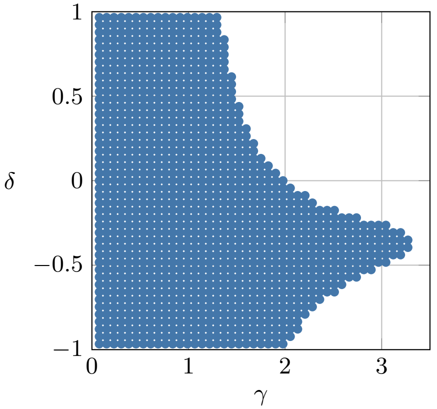
by solving the inclusion problem

$$\text{find } y \in \mathcal{H} \;\text{ such that }\; 0 \in \partial f_1(y) + \partial f_2(y), \tag{5.9}$$

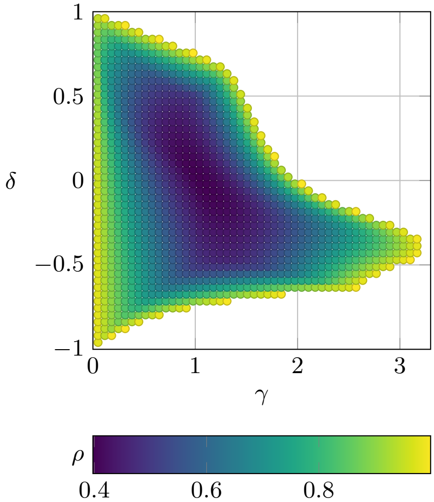
and is given by

$$(\forall k \in \mathbb{N}_0) \quad \left[\begin{aligned} x^{k+1} &= \operatorname{prox}_{\tau f_1}(x^k - \tau y^k), \\ y^{k+1} &= \operatorname{prox}_{\sigma f_2^*}(y^k + \sigma(x^{k+1} + \theta(x^{k+1} - x^k))), \end{aligned}\right. \tag{5.10}$$

where $\tau, \sigma \in \mathbb{R}_{++}$ are primal and dual step sizes, respectively, $\theta \in \mathbb{R}$ is a relaxation parameter, and f_2^* is the conjugate of f_2. The update (5.10) can be written in the



(a) Verification of $f(x^k) - f(y^\star) \in o(1/k)$ as $k \to \infty$ where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$ when $f \in \mathcal{F}_{0,1}(\mathcal{H})$.



(b) The linear convergence rate ρ in (5.8) when f is 0.5-gradient dominated and 1-smooth.

Figure 4. Convergence regions and rates for the gradient method with Nesterov momentum (5.7) with step size γ and momentum parameter δ using AutoLyap.

algorithm representation (3.2) (see [42, Section 2.5.5]) as

$$(\forall k \in \mathbb{N}_0)\quad \left[\begin{aligned} \boldsymbol{x}^{k+1} &= \left(\begin{bmatrix} 1 & -\tau \\ 0 & 0 \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}^k + \left(\begin{bmatrix} -\tau & 0 \\ 0 & 1 \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{u}^k, \\ \boldsymbol{y}^k &= \left(\begin{bmatrix} 1 & -\tau \\ 1 & \frac{1}{\sigma} - \tau(1+\theta) \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}^k + \left(\begin{bmatrix} -\tau & 0 \\ -\tau(1+\theta) & -\frac{1}{\sigma} \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{u}^k, \\ \boldsymbol{u}^k &\in \boldsymbol{\partial f}(\boldsymbol{y}^k), \end{aligned}\right. \tag{5.11}$$

where, in particular, $\boldsymbol{x}^k = (x^k, y^k)$.

For this example, we use the squared fixed-point residual $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2$ as a performance measure. Note that if $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2$ is zero, then x^k is a solution to (5.9). The code below checks whether $(\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2)_{k \in \mathbb{N}_0}$ is summable via the Lyapunov analysis in Section 4.1 by utilizing the history parameter h and overlap parameter α, using the values $(\tau, \sigma, \theta, h, \alpha) = (1, 1, 1, 1, 1)$.

```
components_list = [
    pc.Convex(), # f1
    pc.Convex()  # f2
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.ChambollePock(tau=1, sigma=1, theta=1)
(P, p, T, t) = IterationIndependent.SublinearConvergence.
    get_parameters_fixed_point_residual(algorithm, h=1, alpha=1)
```

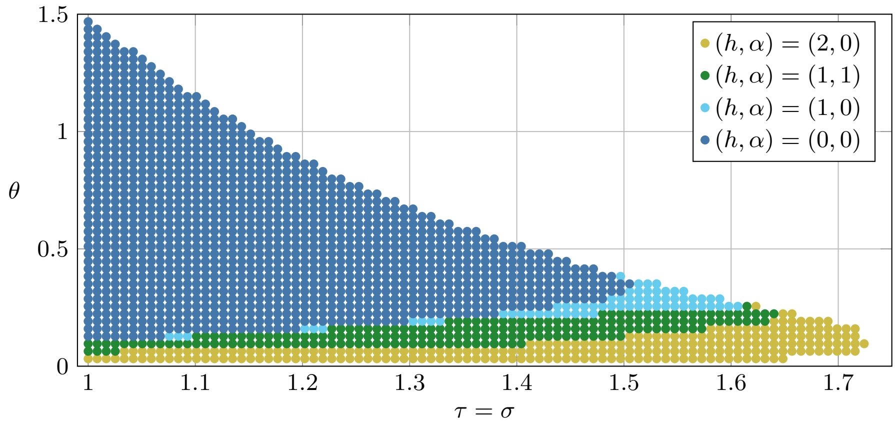


Figure 5. Verification of summability of the squared fixed-point residual $(\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2)_{k\in\mathbb{N}_0}$ for the Chambolle–Pock method (5.11) with primal step size τ and dual step size σ, using AutoLyap and the Lyapunov analysis in Section 4.1 with history parameter h and overlap parameter α.

```
successful = IterationIndependent.verify_iteration_independent_Lyapunov(problem,
    algorithm, P, T, p=p, t=t, rho=1.0, h=1, alpha=1)
```

In this case, the verification is successful, i.e., `successful` is `True`. Repeating this for multiple values of $\tau = \sigma \in (1, 2)$, $\theta \in (0, 3/2)$, and a few different values of (h, α) gives Figure 5. We observe that the region increases in size as h and α increase. However, in our numerical experiments, we did not observe any further increase beyond $(h, \alpha) = (2, 0)$, except for minor artefacts that may be attributed to the SDP solver. Furthermore, the case $(h, \alpha) = (0, 0)$ corresponds to the smallest region, which matches the result in [42, Figure 4a], even though a different performance measure is used there.

5.5 Nesterov's fast gradient method

Suppose that $f \in \mathcal{F}_{0,L}(\mathcal{H})$ for some $L \in \mathbb{R}_{++}$, $\gamma \in \mathbb{R}_{++}$, $\lambda_0 = 1$, $x^{-1}, x^0 \in \mathcal{H}$, and let

$$(\forall k \in \mathbb{N}_0) \quad \begin{bmatrix} y^k = x^k + \delta_k(x^k - x^{k-1}), \\ x^{k+1} = y^k - \gamma \nabla f(y^k), \end{bmatrix} \tag{5.12}$$

where

$$(\forall k \in \mathbb{N}_0) \quad \begin{bmatrix} \delta_k = \dfrac{\lambda_k - 1}{\lambda_{k+1}}, \\ \lambda_{k+1} = \dfrac{1 + \sqrt{1 + 4\lambda_k^2}}{2}, \end{bmatrix}$$

which is a particular instance of Nesterov's fast gradient method [31]. If $\gamma = 1/L$, the Lyapunov analysis in [31] gives the bounds

$$\begin{pmatrix} \forall k \in \mathbb{N} \\ \forall y^\star \in \operatorname{Argmin}_{y\in\mathcal{H}} f(y) \end{pmatrix} \quad \begin{aligned} f(x^k) - f(y^\star) &\leq \frac{L\|x^0 - y^\star\|^2}{2\lambda_k^2} \\ &\leq \frac{2L\|x^0 - y^\star\|^2}{(k+2)^2}. \end{aligned} \tag{5.13}$$

In this example, we use AutoLyap to find the smallest possible constant $c \in \mathbb{R}_+$ such that

$$f(x^k) - f(y^\star) \leq c\|x^0 - y^\star\|^2 \tag{5.14}$$

where $k \in \mathbb{N}$ is fixed and $y^\star \in \operatorname{Argmin}_{y\in\mathcal{H}} f(y)$, provable via the Lyapunov analysis in Section 4.2. Note that the update in (5.12) only evaluates f (or more precisely ∇f) at y^k, and not x^k, which is required for (5.14). Thus, to cover this type of Lyapunov analysis, we need to consider an extended version of the update (5.12) in the algorithm representation (3.2), which explicitly evaluates f at x^k. This can be done via[2]

$$(\forall k \in \mathbb{N}_0) \quad \left[\begin{aligned} \boldsymbol{x}^{k+1} &= \left(\begin{bmatrix} 1+\delta_k & -\delta_k \\ 1 & 0 \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}^k + \left(\begin{bmatrix} -\gamma & 0 \\ 0 & 0 \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{u}^k, \\ \boldsymbol{y}^k &= \left(\begin{bmatrix} 1+\delta_k & -\delta_k \\ 1 & 0 \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{x}^k + \left(\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \otimes \mathrm{Id}\right)\boldsymbol{u}^k, \\ \boldsymbol{u}^k &\in \boldsymbol{\partial f}(\boldsymbol{y}^k), \end{aligned}\right. \tag{5.15}$$

where $\boldsymbol{x}^k = (x^k, x^{k-1})$, $\boldsymbol{u}^k = (\nabla f(y^k), \nabla f(x^k))$, $\boldsymbol{y}^k = (y^k, x^k)$, and $\boldsymbol{\partial f}(\boldsymbol{y}) = \{\nabla f(y)\} \times \{\nabla f(x)\}$ for each $\boldsymbol{y} = (y, x) \in \mathcal{H}^2$.

The code below finds c in (5.14), using the values $(L, \gamma, k) = (1, 1, 10)$.

```
from autolyap import IterationDependent

components_list = [
    pc.SmoothConvex(L=1), # f
]
problem = pc.InclusionProblem(components_list)
algorithm = NesterovFastGradientMethod(gamma=1)
(Q_0, q_0) = IterationDependent.get_parameters_distance_to_solution(algorithm, k=0, i
    =1, j=2)
(Q_k, q_k) = IterationDependent.get_parameters_function_value_suboptimality(algorithm
    , k=10, j=2)
(successful, c) = IterationDependent.verify_iteration_dependent_Lyapunov(problem,
    algorithm, 10, Q_0, Q_k, q_0, q_k)
```

In this case, we get `(successful, c) = (True, 0.0110)`. Repeating this for $k \in [\![1, 100]\!]$ gives Figure 6.

[2] The class `NesterovFastGradientMethod` in the `autolyap.algorithms` submodule implements this version.

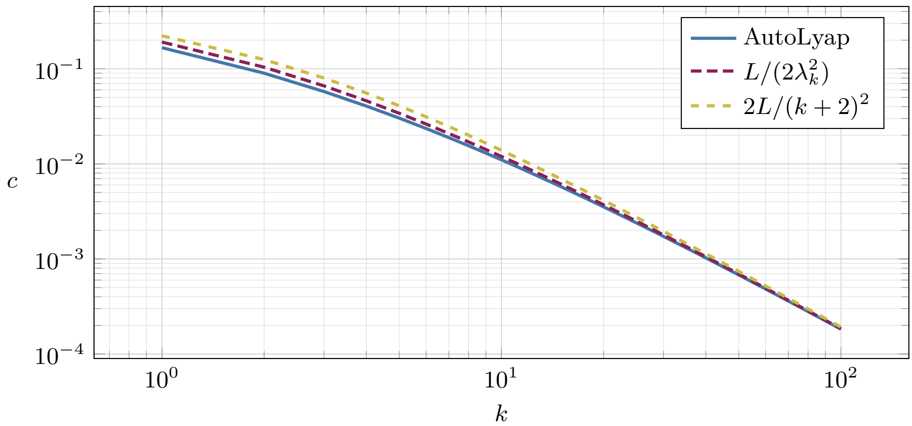


Figure 6. Constants c in (5.14) for Nesterov's fast gradient method (5.12) with step size $\gamma = 1$ applied to a function $f \in \mathcal{F}_{0,1}(\mathcal{H})$, obtained with AutoLyap and compared with the classical rates in [31].

5.6 Optimized gradient method

Let $f \in \mathcal{F}_{0,L}(\mathcal{H})$ for some $L \in \mathbb{R}_{++}$. The optimized gradient method, first considered in [14] and formally obtained in [21], is given by

$$
\begin{aligned}
&(\forall k \in [\![0, K-1]\!]) \\
&\left\lfloor \begin{aligned}
y^{k+1} &= x^k - \frac{1}{L}\nabla f(x^k), \\
x^{k+1} &= y^{k+1} + \frac{\theta_k - 1}{\theta_{k+1}}(y^{k+1} - y^k) + \frac{\theta_k}{\theta_{k+1}}(y^{k+1} - x^k),
\end{aligned}\right.
\end{aligned}
\tag{5.16}
$$

where $x^0, y^0 \in \mathcal{H}$ are initial points, $K \in \mathbb{N}$ is the iteration budget, and

$$
\theta_k = \begin{cases} 1 & \text{if } k = 0, \\ \frac{1+\sqrt{1+4\theta_{k-1}^2}}{2} & \text{if } k \in [\![1, K-1]\!], \\ \frac{1+\sqrt{1+8\theta_{k-1}^2}}{2} & \text{if } k = K. \end{cases}
$$

The bound

$$
(\forall y^\star \in \operatorname*{Argmin}_{y \in \mathcal{H}} f(y)) \quad f(x^K) - f(y^\star) \leq \frac{L\|x^0 - y^\star\|^2}{2\theta_K^2}, \tag{5.17}
$$

is proven in [21, Theorem 2], and [21, Section 8], using [40, Section 4.2], established tightness in the sense that there exists a function f that achieves equality in (5.17). Later, [13] proved that the bound (5.17) is the tightest possible among all first-order methods for smooth and convex minimization. A Lyapunov analysis matching (5.17) is provided in [10, Section 4.3.1].

The code below uses AutoLyap to find the smallest possible constant $c \in \mathbb{R}_+$ provable via the Lyapunov analysis in Section 4.2 such that

$$
f(x^K) - f(y^\star) \leq c\|x^0 - y^\star\|^2 \tag{5.18}
$$

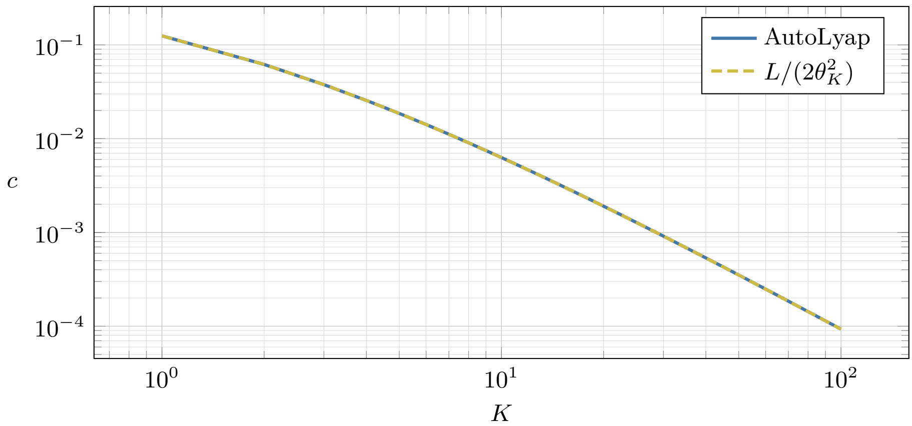


Figure 7. Constants c in (5.18) for the optimized gradient method (5.16) applied to a function $f \in \mathcal{F}_{0,1}(\mathcal{H})$, obtained with AutoLyap and compared with the rate in (5.17).

where $y^\star \in \operatorname{Argmin}_{y \in \mathcal{H}} f(y)$, using the values $(L, K) = (1, 10)$.

```
components_list = [
    pc.SmoothConvex(L=1), # f
]
problem = pc.InclusionProblem(components_list)
algorithm = algs.OptimizedGradientMethod(L=1, K=10)
(Q_0, q_0) = IterationDependent.get_parameters_distance_to_solution(algorithm, k=0)
(Q_K, q_K) = IterationDependent.get_parameters_function_value_suboptimality(algorithm
    , k=10)
(successful, c) = IterationDependent.verify_iteration_dependent_Lyapunov(problem,
    algorithm, 10, Q_0, Q_K, q_0, q_K)
```

In this case, we get `(successful, c) = (True, 0.0063)`. Repeating this for $k \in [\![1, 100]\!]$ gives Figure 7.

6. Mathematical background

Section 6.1 introduces the notation and mathematical preliminaries used in this paper. Section 6.2 introduces the main theoretical tool, which enables recasting the verification of the existence of a Lyapunov analysis as solving an SDP. This is then used in Section 6.3 for the iteration-independent Lyapunov analyses presented in Section 4.1, and in Section 6.4 for the iteration-dependent Lyapunov analyses presented in Section 4.2.

6.1 Notation and preliminaries

Let $\mathbb{N}_0$ denote the set of nonnegative integers, $\mathbb{N}$ the set of positive integers, $\mathbb{Z}$ the set of integers, $[\![n, m]\!] = \{l \in \mathbb{Z} \mid n \leq l \leq m\}$ the set of integers between $n, m \in \mathbb{Z} \cup \{\pm\infty\}$, $\mathbb{R}$ the set of real numbers, $\mathbb{R}_+$ the set of nonnegative real numbers, $\mathbb{R}_{++}$ the set of positive real numbers, $\mathbb{R}^n$ the set of all n-tuples of elements of $\mathbb{R}$, $\mathbb{R}^{m \times n}$ the set of real-valued matrices of size $m \times n$, if $M \in \mathbb{R}^{m \times n}$ then $[M]_{i,j}$ the i, j-th element of M, $\mathbb{S}^n$ the set of symmetric real-valued matrices of size $n \times n$, $\mathbb{S}^n_+ \subseteq \mathbb{S}^n$ the set of positive semidefinite real-valued matrices of size $n \times n$, $0_{n\times m} \in \mathbb{R}^{n\times m}$ the matrix of all zeros of size $n \times m$, $I_n \in \mathbb{R}^{n\times n}$ the identity matrix of size $n \times n$, $e_i^n \in \mathbb{R}^n$ the ith standard basis vector in $\mathbb{R}^n$, and $\mathbf{1}_n \in \mathbb{R}^n$ the vector of all ones in $\mathbb{R}^n$. All vectors in $\mathbb{R}^n$ are column vectors by convention.

Throughout this paper, $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ will denote a real Hilbert space. All norms $\|\cdot\|$ are canonical norms where the inner product will be clear from the context. We denote the identity mapping $x \mapsto x$ on $\mathcal{H}$ by Id.

Definition 6.1
Let $f\colon \mathcal{H} \to \mathbb{R} \cup \{\pm\infty\}$, $L \in \mathbb{R}_+$ and $\mu, \widetilde{\mu}, \mu_{\mathrm{gd}} \in \mathbb{R}_{++}$. The function f is said to be

(i) proper *if $-\infty \notin f(\mathcal{H})$ and* $\operatorname{dom} f \neq \emptyset$, *where the set* $\operatorname{dom} f = \{x \in \mathcal{H} \mid f(x) < +\infty\}$ *is called the* effective domain *of f,*

(ii) lower semicontinuous *if $\liminf_{y\to x} f(y) \geq f(x)$ for each $x \in \mathcal{H}$,*

(iii) convex *if $f((1-\lambda)x + \lambda y) \leq (1-\lambda)f(x) + \lambda f(y)$ for each $x, y \in \mathcal{H}$ and $0 \leq \lambda \leq 1$,*

(iv) μ-strongly convex *if f is proper and $f - (\mu/2)\|\cdot\|^2$ is convex,*

(v) $\widetilde{\mu}$-weakly convex *if $f + (\widetilde{\mu}/2)\|\cdot\|^2$ is convex,*

(vi) L-smooth *if f is Fréchet differentiable and the gradient $\nabla f : \mathcal{H} \to \mathcal{H}$ is L-Lipschitz continuous, i.e., $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ for each $x, y \in \mathcal{H}$, and*

(vii) μ_{gd}-gradient dominated *if f is Fréchet differentiable and*

$$f(x) - \inf_{y\in\mathcal{H}} f(y) \leq \frac{1}{2\mu_{\mathrm{gd}}}\|\nabla f(x)\|^2 \tag{6.1}$$

for each $x \in \mathcal{H}$. Inequality (6.1) is sometimes called the Polyak–Łojasiewicz *inequality or simply the* Łojasiewicz *inequality.*

Definition 6.2
Let $-\infty < \mu \leq L \leq +\infty$ such that $L \geq 0$. We let $\mathcal{F}_{\mu,L}(\mathcal{H})$ denote the class of all proper and lower semicontinuous functions $f : \mathcal{H} \to \mathbb{R} \cup \{\pm\infty\}$ such that

(i) $(L/2)\|\cdot\|^2 - f$ *is convex and f is Fréchet differentiable if $L < +\infty$, and*

(ii) $f - (\mu/2)\|\cdot\|^2$ *is convex.*

For example, $\mathcal{F}_{-L,L}(\mathcal{H})$ is equal to the class of L-smooth functions with domain $\mathcal{H}$.

The *Fréchet subdifferential* of a function $f\colon \mathcal{H} \to \mathbb{R} \cup \{\pm\infty\}$ is the set-valued operator $\partial f : \mathcal{H} \rightrightarrows \mathcal{H}$ given by

$$\partial f(x) = \begin{cases} \left\{ u \in \mathcal{H} \;\middle|\; \liminf\limits_{y\to x} \dfrac{f(y) - f(x) - \langle u, y - x\rangle}{\|y - x\|} \geq 0 \right\} & \text{if } |f(x)| < +\infty, \\ \emptyset & \text{otherwise} \end{cases}$$

for each $x \in \mathcal{H}$.

(i) If f is Fréchet differentiable at a point $x \in \mathcal{H}$, then $\partial f(x) = \{\nabla f(x)\}$ [29, Proposition 1.87].

(ii) If f is proper and convex, the Fréchet subdifferential becomes the *convex subdifferential*, i.e., $\partial f(x) = \{u \in \mathcal{H} \mid \forall y \in \mathcal{H},\ f(y) \geq f(x) + \langle u, y - x\rangle\}$ for each $x \in \mathcal{H}$.

(iii) If f is proper and $\widetilde{\mu}$-weakly convex for some $\widetilde{\mu} \in \mathbb{R}_{++}$, then $\partial f(x) = \partial(f + (\widetilde{\mu}/2)\|\cdot\|^2)(x) - \widetilde{\mu}x$ for each $x \in \mathcal{H}$ [29, Proposition 1.107 (i)], where $\partial(f + (\widetilde{\mu}/2)\|\cdot\|^2)$ reduces to the convex subdifferential.

Definition 6.3
Suppose that $f \in \mathcal{F}_{0,\infty}(\mathcal{H})$ and $\gamma \in \mathbb{R}_{++}$. Then the proximal operator *of f with* step size *γ, denoted $\operatorname{prox}_{\gamma f} : \mathcal{H} \to \mathcal{H}$, is defined as the single-valued operator given by*

$$\operatorname{prox}_{\gamma f}(x) = \underset{z \in \mathcal{H}}{\operatorname{argmin}} \left(f(z) + \frac{1}{2\gamma}\|x - z\|^2 \right)$$

for each $x \in \mathcal{H}$.

Suppose that $f \in \mathcal{F}_{0,\infty}(\mathcal{H})$ and $\gamma \in \mathbb{R}_{++}$. If $x, p \in \mathcal{H}$, then $p = \operatorname{prox}_{\gamma f}(x) \Leftrightarrow \gamma^{-1}(x - p) \in \partial f(p)$. Moreover, the *conjugate* of f, denoted $f^* : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, is the proper, lower semicontinuous and convex function given by $f^*(u) = \sup_{x \in \mathcal{H}} (\langle u, x\rangle - f(x))$ for each $u \in \mathcal{H}$. If $x, u \in \mathcal{H}$, then $u \in \partial f(x) \Leftrightarrow x \in \partial f^*(u)$.

Let $G : \mathcal{H} \rightrightarrows \mathcal{H}$ be a set-valued operator. The set of *zeros* of G is denoted by $\operatorname{zer} G = \{x \in \mathcal{H} \mid 0 \in G(x)\}$ and the *graph* of G is denoted by $\operatorname{gra} G = \{(x, y) \in \mathcal{H} \times \mathcal{H} \mid y \in G(x)\}$.

Definition 6.4
Let $G : \mathcal{H} \rightrightarrows \mathcal{H}$ and $\mu \in \mathbb{R}_{++}$. The operator G is said to be

(i) monotone *if $\langle u - v, x - y\rangle \geq 0$ for each $(x, u), (y, v) \in \operatorname{gra} G$,*

(ii) maximally monotone *if G is monotone and there does not exist a monotone operator $H : \mathcal{H} \rightrightarrows \mathcal{H}$ such that $\operatorname{gra} G \subsetneq \operatorname{gra} H$, and*

(iii) μ-strongly monotone *if $\langle u - v, x - y\rangle \geq \mu\|x - y\|^2$ for each $(x, u), (y, v) \in \operatorname{gra} G$.*

The *inverse* of a set-valued operator $G : \mathcal{H} \rightrightarrows \mathcal{H}$, denoted by $G^{-1} : \mathcal{H} \rightrightarrows \mathcal{H}$, is defined through its graph $\operatorname{gra} G^{-1} = \{(y, x) \in \mathcal{H} \times \mathcal{H} \mid (x, y) \in \operatorname{gra} G\}$.

Definition 6.5
Suppose that $G : \mathcal{H} \rightrightarrows \mathcal{H}$ is maximally monotone and $\gamma \in \mathbb{R}_{++}$. The resolvent *of G with* step size *γ, denoted $J_{\gamma G} : \mathcal{H} \to \mathcal{H}$, is defined by*

$$(\operatorname{Id} + \gamma G)^{-1}(x) = \{J_{\gamma G}(x)\}$$

for each $x \in \mathcal{H}$, since $(\operatorname{Id} + \gamma G)^{-1}$ is singleton-valued in this case.

Definition 6.6
Let $G : \mathcal{H} \to \mathcal{H}$, $L \in \mathbb{R}_+$, and $\beta \in \mathbb{R}_{++}$. The operator G is said to be

(i) L-Lipschitz continuous *if $\|G(x) - G(y)\| \leq L\|x - y\|$ for each $x, y \in \mathcal{H}$, and*

(ii) β-cocoercive *if $\langle G(x) - G(y), x - y\rangle \geq \beta\|G(x) - G(y)\|^2$ for each $x, y \in \mathcal{H}$.*

We introduce the following conventions that enable us to treat single-valued and singleton-valued operators interchangeably.

(i) For notational convenience (at the expense of a slight abuse of notation), we will sometimes identify the operator $G : \mathcal{H} \to \mathcal{H}$ with the set-valued mapping $\mathcal{H} \ni x \mapsto \{G(x)\} \subseteq \mathcal{H}$, which will be clear from context. For example, if $x, y \in \mathcal{H}$, the inclusion $y \in G(x)$ should be interpreted as the equality $y = G(x)$.

(ii) Similarly, if $G : \mathcal{H} \rightrightarrows \mathcal{H}$ and $T : \mathcal{H} \to \mathcal{H}$ satisfy $G(x) = \{T(x)\}$ for each $x \in \mathcal{H}$, i.e., G is a singleton-valued operator, we will sometimes identify G with the corresponding single-valued operator T.

Given any positive integer n, we let the inner product $\langle \cdot, \cdot \rangle$ on $\mathcal{H}^n$ be given by

$$\langle \boldsymbol{z}^1, \boldsymbol{z}^2 \rangle = \sum_{j=1}^{n} \langle z_j^1, z_j^2 \rangle$$

for each $\boldsymbol{z}^i = (z_1^i, \ldots, z_n^i) \in \mathcal{H}^n$ and $i \in [\![1, 2]\!]$. If $M \in \mathbb{R}^{m \times n}$, we define the tensor product $M \otimes \mathrm{Id}$ to be the mapping $(M \otimes \mathrm{Id}) : \mathcal{H}^n \to \mathcal{H}^m$ such that

$$(M \otimes \mathrm{Id})\boldsymbol{z} = \left(\sum_{j=1}^{n} [M]_{1,j} z_j, \ldots, \sum_{j=1}^{n} [M]_{m,j} z_j \right)$$

for each $\boldsymbol{z} = (z_1, \ldots, z_n) \in \mathcal{H}^n$. The adjoint satisfies $(M \otimes \mathrm{Id})^* = M^\top \otimes \mathrm{Id}$. If $N \in \mathbb{R}^{n \times l}$, the composition rule $(M \otimes \mathrm{Id}) \circ (N \otimes \mathrm{Id}) = (MN) \otimes \mathrm{Id}$ holds.

If we let $M_1 \in \mathbb{R}^{m \times n_1}$ and $M_2 \in \mathbb{R}^{m \times n_2}$, the relations above imply that $\langle (M_1 \otimes \mathrm{Id})\boldsymbol{z}^1, (M_2 \otimes \mathrm{Id})\boldsymbol{z}^2 \rangle = \langle \boldsymbol{z}^1, ((M_1^\top M_2) \otimes \mathrm{Id})\boldsymbol{z}^2 \rangle$ for each $\boldsymbol{z}^1 \in \mathcal{H}^{n_1}$ and $\boldsymbol{z}^2 \in \mathcal{H}^{n_2}$. We define the mapping $\mathcal{Q} : \mathbb{S}^n \times \mathcal{H}^n \to \mathbb{R}$ by $\mathcal{Q}(M, \boldsymbol{z}) = \langle \boldsymbol{z}, (M \otimes \mathrm{Id})\boldsymbol{z} \rangle$ for each $M \in \mathbb{S}^n$ and $\boldsymbol{z} \in \mathcal{H}^n$. Note that, if $M \in \mathbb{S}^n$, $N \in \mathbb{R}^{n \times m}$ and $\boldsymbol{z} \in \mathcal{H}^m$, then $\mathcal{Q}(M, (N \otimes \mathrm{Id})\boldsymbol{z}) = \mathcal{Q}(N^\top M N, \boldsymbol{z})$. We define the *Gramian function* $\mathtt{G} : \mathcal{H}^n \to \mathbb{S}_+^n$ such that $[\mathtt{G}(\boldsymbol{z})]_{i,j} = \langle z_i, z_j \rangle$ for each $\boldsymbol{z} = (z_1, \ldots, z_n) \in \mathcal{H}^n$. If $M \in \mathbb{S}^n$ and $\boldsymbol{z} \in \mathcal{H}^n$, it holds that $\mathcal{Q}(M, \boldsymbol{z}) = \mathrm{trace}(M\mathtt{G}(\boldsymbol{z}))$.

6.2 Main tool

This section presents a generalization of [42, Lemma 1] for the algorithm representation in (3.2). First, we introduce some necessary matrices. If $m \geq 2$, let

$$N = \begin{bmatrix} I_{m-1} \\ -\mathbf{1}_{m-1}^\top \end{bmatrix} \in \mathbb{R}^{m \times (m-1)}. \tag{6.2}$$

For each $\underline{k}, \bar{k} \in [\![0, K]\!]$ such that $\underline{k} \leq \bar{k}$, we define the matrices $X_k^{\underline{k},\bar{k}} \in \mathbb{R}^{n \times (n + (\bar{k} - \underline{k} + 1)\bar{m} + m)}$ as

$$X_k^{\underline{k},\bar{k}} = \begin{cases} \begin{bmatrix} I_n & 0_{n \times ((\bar{k} - \underline{k} + 1)\bar{m} + m)} \end{bmatrix} & \text{if } k = \underline{k}, \\ \begin{bmatrix} A_{\underline{k}} & B_{\underline{k}} & 0_{n \times ((\bar{k} - \underline{k})\bar{m} + m)} \end{bmatrix} & \text{if } k = \underline{k} + 1, \\ \begin{bmatrix} (A_{k-1} \cdots A_{\underline{k}})^\top \\ (A_{k-1} \cdots A_{\underline{k}+1} B_{\underline{k}})^\top \\ (A_{k-1} \cdots A_{\underline{k}+2} B_{\underline{k}+1})^\top \\ \vdots \\ (A_{k-1} A_{k-2} B_{k-3})^\top \\ (A_{k-1} B_{k-2})^\top \\ B_{k-1}^\top \\ 0_{n \times ((\bar{k} + 1 - k)\bar{m} + m)}^\top \end{bmatrix}^\top & \begin{array}{l} \text{if } k \in [\![\underline{k} + 2, \bar{k} + 1]\!] \\ \text{and } \underline{k} + 1 \leq \bar{k}, \end{array} \end{cases} \tag{6.3}$$

the matrices $Y_k^{\underline{k},\bar{k}} \in \mathbb{R}^{\bar{m}\times(n+(\bar{k}-\underline{k}+1)\bar{m}+m)}$ as

$$Y_k^{\underline{k},\bar{k}} = \begin{cases} \begin{bmatrix} C_{\underline{k}} & D_{\underline{k}} & 0_{\bar{m}\times((\bar{k}-\underline{k})\bar{m}+m)} \end{bmatrix} & \text{if } k = \underline{k}, \\ \begin{bmatrix} (C_{\underline{k}+1}A_{\underline{k}})^\top \\ (C_{\underline{k}+1}B_{\underline{k}})^\top \\ D_{\underline{k}+1}^\top \\ 0^\top_{\bar{m}\times((\bar{k}-\underline{k}-1)\bar{m}+m)} \end{bmatrix}^\top & \begin{array}{l}\text{if } k = \underline{k}+1 \\ \text{and } \underline{k}+1 \leq \bar{k},\end{array} \\ \begin{bmatrix} (C_k A_{k-1}\cdots A_{\underline{k}})^\top \\ (C_k A_{k-1}\cdots A_{\underline{k}+1}B_{\underline{k}})^\top \\ (C_k A_{k-1}\cdots A_{\underline{k}+2}B_{\underline{k}+1})^\top \\ \vdots \\ (C_k A_{k-1}B_{k-2})^\top \\ (C_k B_{k-1})^\top \\ D_k^\top \\ 0^\top_{\bar{m}\times((\bar{k}-k)\bar{m}+m)} \end{bmatrix}^\top & \begin{array}{l}\text{if } k \in [\![\underline{k}+2,\bar{k}]\!] \\ \text{and } \underline{k}+2 \leq \bar{k},\end{array} \end{cases} \tag{6.4}$$

the matrix

$$Y_\star^{\underline{k},\bar{k}} = \underbrace{\begin{bmatrix} 0_{m\times(n+(\bar{k}-\underline{k}+1)\bar{m}+m-1)} & \mathbf{1}_m \end{bmatrix}}_{\in\mathbb{R}^{m\times(n+(\bar{k}-\underline{k}+1)\bar{m}+m)}}, \tag{6.5}$$

the matrices $U_k^{\underline{k},\bar{k}} \in \mathbb{R}^{\bar{m}\times(n+(\bar{k}-\underline{k}+1)\bar{m}+m)}$

$$(\forall k \in [\![\underline{k},\bar{k}]\!]) \quad U_k^{\underline{k},\bar{k}} = \begin{bmatrix} 0_{\bar{m}\times(n+(k-\underline{k})\bar{m})} & I_{\bar{m}} & 0_{\bar{m}\times((\bar{k}-k)\bar{m}+m)} \end{bmatrix}, \tag{6.6}$$

the matrix

$$U_\star^{\underline{k},\bar{k}} = \underbrace{\begin{bmatrix} 0_{m\times(n+(\bar{k}-\underline{k}+1)\bar{m})} & N & 0_{m\times 1} \end{bmatrix}}_{\in\mathbb{R}^{m\times(n+(\bar{k}-\underline{k}+1)\bar{m}+m)}} \tag{6.7}$$

with the interpretation that the block column containing N is removed from $U_\star^{\underline{k},\bar{k}}$ when $m = 1$, the matrices

$$\begin{aligned} \begin{pmatrix} \forall i \in [\![1,m]\!] \\ \forall j \in [\![1,\bar{m}_i]\!] \end{pmatrix} \quad & P_{(i,j)} = \underbrace{\begin{bmatrix} 0_{1\times\sum_{r=1}^{i-1}\bar{m}_r} & (e_j^{\bar{m}_i})^\top & 0_{1\times\sum_{r=i+1}^{m}\bar{m}_r} \end{bmatrix}}_{\in\mathbb{R}^{1\times\bar{m}}}, \\ (\forall i \in [\![1,m]\!]) \quad & P_{(i,\star)} = (e_i^m)^\top \in \mathbb{R}^{1\times m}, \end{aligned} \tag{6.8}$$

and the matrices $F_{(i,j,k)}^{\underline{k},\bar{k}}, F_{(i,\star,\star)}^{\underline{k},\bar{k}} \in \mathbb{R}^{1\times((\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}})}$

$$\begin{aligned} \begin{pmatrix} \forall i \in [\![1,m]\!] \\ \forall j \in [\![1,\bar{m}_i]\!] \\ \forall k \in [\![\underline{k},\bar{k}]\!] \end{pmatrix} \quad & F_{(i,j,k)}^{\underline{k},\bar{k}} = \begin{bmatrix} 0^\top_{1\times((k-\underline{k})\bar{m}_{\text{func}}+\sum_{r=1}^{\kappa(i)-1}\bar{m}_{\kappa^{-1}(r)})} \\ e_j^{\bar{m}_i} \\ 0^\top_{1\times(((\bar{k}-k))\bar{m}_{\text{func}}+m_{\text{func}}+\sum_{r=\kappa(i)+1}^{m_{\text{func}}}\bar{m}_{\kappa^{-1}(r)})} \end{bmatrix}^\top, \\ (\forall i \in [\![1,m]\!]) \quad & F_{(i,\star,\star)}^{\underline{k},\bar{k}} = \begin{bmatrix} 0_{1\times((\bar{k}-\underline{k}+1)\bar{m}_{\text{func}})} & (e_{\kappa(i)}^{m_{\text{func}}})^\top \end{bmatrix}, \end{aligned} \tag{6.9}$$

where $\kappa : \mathcal{I}_{\text{func}} \to [\![1, m_{\text{func}}]\!]$ is a bijective and increasing function (and therefore uniquely specified).

Next, we present the main object of interest. Let $\underline{k}, \bar{k} \in [\![0, K]\!]$ such that $\underline{k} \le \bar{k}$, $W \in \mathbb{S}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}$, $w \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}$ and consider the optimization problem

$$
\begin{array}{ll}
\text{maximize} & \mathcal{Q}(W, (\boldsymbol{x}^{\underline{k}}, \boldsymbol{u}^{\underline{k}}, \ldots, \boldsymbol{u}^{\bar{k}}, \hat{\boldsymbol{u}}^\star, y^\star)) + w^\top (\boldsymbol{F}^{\underline{k}}, \ldots, \boldsymbol{F}^{\bar{k}}, \boldsymbol{F}^\star) \\
\text{subject to} & \textbf{for each } k \in [\![\underline{k}, \bar{k}]\!] \\
& \quad \boldsymbol{x}^{k+1} = (A_k \otimes \mathrm{Id})\boldsymbol{x}^k + (B_k \otimes \mathrm{Id})\boldsymbol{u}^k, \\
& \quad \boldsymbol{y}^k = (C_k \otimes \mathrm{Id})\boldsymbol{x}^k + (D_k \otimes \mathrm{Id})\boldsymbol{u}^k, \\
& \quad (\boldsymbol{u}_i^k)_{i\in\mathcal{I}_{\text{func}}} \in \displaystyle\prod_{i\in\mathcal{I}_{\text{func}}} \boldsymbol{\partial f}_i(\boldsymbol{y}_i^k), \\
& \quad (\boldsymbol{u}_i^k)_{i\in\mathcal{I}_{\text{op}}} \in \displaystyle\prod_{i\in\mathcal{I}_{\text{op}}} \boldsymbol{G}_i(\boldsymbol{y}_i^k), \\
& \quad \boldsymbol{F}^k = (\boldsymbol{f}_i(\boldsymbol{y}_i^k))_{i\in\mathcal{I}_{\text{func}}} \in \mathbb{R}^{\bar{m}_{\text{func}}}, \\
& \quad \boldsymbol{u}^k = (\boldsymbol{u}_1^k, \ldots, \boldsymbol{u}_m^k) \in \displaystyle\prod_{i=1}^m \mathcal{H}^{\bar{m}_i}, \\
& \quad \boldsymbol{y}^k = (\boldsymbol{y}_1^k, \ldots, \boldsymbol{y}_m^k) \in \displaystyle\prod_{i=1}^m \mathcal{H}^{\bar{m}_i}, \\
& \textbf{end} \\
& \boldsymbol{u}^\star = (u_1^\star, \ldots, u_m^\star) \in \mathcal{H}^m, \\
& y^\star \in \mathcal{H}, \\
& (u_i^\star)_{i\in\mathcal{I}_{\text{func}}} \in \displaystyle\prod_{i\in\mathcal{I}_{\text{func}}} \partial f_i(y^\star), \\
& (u_i^\star)_{i\in\mathcal{I}_{\text{op}}} \in \displaystyle\prod_{i\in\mathcal{I}_{\text{op}}} G_i(y^\star), \\
& \displaystyle\sum_{i=1}^m u_i^\star = 0, \\
& \hat{\boldsymbol{u}}^\star = (u_1^\star, \ldots, u_{m-1}^\star), \\
& \boldsymbol{F}^\star = (\boldsymbol{f}_i(y^\star))_{i\in\mathcal{I}_{\text{func}}} \in \mathbb{R}^{m_{\text{func}}}, \\
& (f_i)_{i\in\mathcal{I}_{\text{func}}} \in \displaystyle\prod_{i\in\mathcal{I}_{\text{func}}} \mathcal{F}_i, \\
& (G_i)_{i\in\mathcal{I}_{\text{op}}} \in \displaystyle\prod_{i\in\mathcal{I}_{\text{op}}} \mathcal{G}_i,
\end{array}
\tag{PEP}
$$

where everything except W, w, $\underline{k}$, $\bar{k}$, $((A_k, B_k, C_k, D_k))_{k=\underline{k}}^{\bar{k}}$, $\mathcal{H}$, $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$, and $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ are optimization variables. Note that the last two constraints in (PEP) are infinite-dimensional. We use the following assumption to reduce these infinite-dimensional constraints to a finite set of quadratic constraints.

Assumption 6.7
Consider (3.2) and for notational convenience let

$$\begin{pmatrix}\forall i \in \mathcal{I}_{\text{func}} \cup \mathcal{I}_{\text{op}} \\ \forall k \in [\![0, K]\!]\end{pmatrix} \quad \boldsymbol{u}_i^k = (u_{i,1}^k, \ldots, u_{i,\bar{m}_i}^k) \in \mathcal{H}^{\bar{m}_i},$$

$$\begin{pmatrix}\forall i \in \mathcal{I}_{\text{func}} \cup \mathcal{I}_{\text{op}} \\ \forall k \in [\![0, K]\!]\end{pmatrix} \quad \boldsymbol{y}_i^k = (y_{i,1}^k, \ldots, y_{i,\bar{m}_i}^k) \in \mathcal{H}^{\bar{m}_i},$$

$$\begin{pmatrix}\forall i \in \mathcal{I}_{\text{func}} \\ \forall k \in [\![0, K]\!]\end{pmatrix} \quad \boldsymbol{f}_i(\boldsymbol{y}_i^k) = (F_{i,1}^k, \ldots, F_{i,\bar{m}_i}^k) \in \mathbb{R}^{\bar{m}_i}$$

and

$$\begin{aligned}
\boldsymbol{u}^\star &= (u_1^\star, \ldots, u_m^\star) = (u_{1,\star}^\star, \ldots, u_{m,\star}^\star) \in \mathcal{H}^m \quad \textit{such that} \quad \sum_{i=1}^m u_i^\star = 0, \\
y^\star &= y_{1,\star}^\star = \cdots = y_{m,\star}^\star \in \mathcal{H}, \\
\boldsymbol{F}^\star &= (F_{i,\star}^\star)_{i \in \mathcal{I}_{\text{func}}} \in \mathbb{R}^{m_{\text{func}}}.
\end{aligned}$$

(a) For each $i \in \mathcal{I}_{\text{func}}$, suppose that there exist finite and disjoint sets $\mathcal{O}_i^{\text{func-ineq}}$ and $\mathcal{O}_i^{\text{func-eq}}$, vectors and matrices

$$\begin{aligned}
(\forall o \in \mathcal{O}_i^{\text{func-ineq}}) \quad (a_{(i,o)}^{\text{func-ineq}}, M_{(i,o)}^{\text{func-ineq}}) \in \times \mathbb{R}^{n_{i,o}} \times \mathbb{S}^{2n_{i,o}}, \\
(\forall o \in \mathcal{O}_i^{\text{func-eq}}) \quad (a_{(i,o)}^{\text{func-eq}}, M_{(i,o)}^{\text{func-eq}}) \in \times \mathbb{R}^{n_{i,o}} \times \mathbb{S}^{2n_{i,o}},
\end{aligned}$$

and, depending on $\underline{k}, \bar{k} \in [\![0, K]\!]$ such that $\underline{k} \leq \bar{k}$, index sets

$$\begin{gathered}
(\forall o \in \mathcal{O}_i^{\text{func-ineq}} \cup \mathcal{O}_i^{\text{func-eq}}) \\
\mathcal{J}_{i,o}^{\underline{k},\bar{k}} \subseteq \left(([\![1, \bar{m}_i]\!] \times [\![\underline{k}, \bar{k}]\!]) \cup \{(\star, \star)\} \right)^{n_{i,o}},
\end{gathered}$$

such that (i) implies (ii) below:

(i) There exists a function $f_i \in \mathcal{F}_i$ such that

$$(\forall (j,k) \in ([\![1, \bar{m}_i]\!] \times [\![\underline{k}, \bar{k}]\!]) \cup \{(\star, \star)\}) \quad \left[\begin{aligned} f_i(y_{i,j}^l) &= F_{i,j}^k, \\ u_{i,j}^k &\in \partial f_i(y_{i,j}^k). \end{aligned}\right.$$

(ii) It holds that

$$\begin{aligned}
&(\forall o \in \mathcal{O}_i^{\text{func-ineq}})(\forall ((j_1, k_1), \ldots, (j_{n_{i,o}}, k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}) \\
&(a_{(i,o)}^{\text{func-ineq}})^\top \begin{bmatrix} F_{i,j_1}^{k_1} \\ \vdots \\ F_{i,j_{n_{i,o}}}^{k_{n_{i,o}}} \end{bmatrix} \\
&+ \mathcal{Q}(M_{(i,o)}^{\text{func-ineq}}, (y_{i,j_1}^{k_1}, \ldots, y_{i,j_{n_{i,o}}}^{k_{n_{i,o}}}, u_{i,j_1}^{k_1}, \ldots, u_{i,j_{n_{i,o}}}^{k_{n_{i,o}}})) \leq 0,
\end{aligned}$$

and

$$(\forall o \in \mathcal{O}_i^{\text{func-eq}})(\forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}})$$
$$(a_{(i,o)}^{\text{func-eq}})^\top \begin{bmatrix} F_{i,j_1}^{k_1} \\ \vdots \\ F_{i,j_{n_{i,o}}}^{k_{n_{i,o}}} \end{bmatrix}$$
$$+ \mathcal{Q}(M_{(i,o)}^{\text{func-eq}}, (y_{i,j_1}^{k_1},\ldots,y_{i,j_{n_{i,o}}}^{k_{n_{i,o}}}, u_{i,j_1}^{k_1},\ldots,u_{i,j_{n_{i,o}}}^{k_{n_{i,o}}})) = 0,$$

Moreover, if the converse holds, i.e., (ii) implies (i), then we say that the function class $\mathcal{F}_i$ has a tight interpolation condition*.*

(b) Similarly, for each $i \in \mathcal{I}_{\text{op}}$, suppose that there exists a finite set $\mathcal{O}_i^{\text{op}}$, matrices

$$(\forall o \in \mathcal{O}_i^{\text{op}}) \quad M_{(i,o)}^{\text{op}} \in \mathbb{S}^{2n_{i,o}},$$

and, depending on $\underline{k},\bar{k} \in [\![0,K]\!]$ such that $\underline{k} \le \bar{k}$, index sets

$$(\forall o \in \mathcal{O}_i^{\text{op}}) \quad \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \subseteq \left(([\![1,\bar{m}_i]\!] \times [\![\underline{k},\bar{k}]\!]) \cup \{(\star,\star)\}\right)^{n_{i,o}}$$

such that (i) implies (ii) below:

(i) There exists an operator $G_i \in \mathcal{G}_i$ such that

$$(\forall (j,k) \in ([\![1,\bar{m}_i]\!] \times [\![\underline{k},\bar{k}]\!]) \cup \{(\star,\star)\}) \quad u_{i,j}^k \in G_i(y_{i,j}^k).$$

(ii) It holds that

$$(\forall o \in \mathcal{O}_i^{\text{op}})(\forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}})$$
$$\mathcal{Q}(M_{(i,o)}^{\text{op}}, (y_{i,j_1}^{k_1},\ldots,y_{i,j_{n_{i,o}}}^{k_{n_{i,o}}}, u_{i,j_1}^{k_1},\ldots,u_{i,j_{n_{i,o}}}^{k_{n_{i,o}}})) \le 0.$$

Moreover, if the converse holds, i.e., (ii) implies (i), then we say that the operator class $\mathcal{G}_i$ has a tight interpolation condition*.*

Examples of interpolation conditions for some function and operator classes are provided in Appendix A.

We are now ready to state the main theoretical result.

Theorem 6.8
Suppose that Assumption 6.7 holds, let (PEP)* be the optimal value of (PEP), and consider the matrices defined in (6.4) to (6.9). A sufficient condition for (PEP)$^\star \le 0$ is that the following system

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-ineq}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \end{pmatrix} \quad \lambda_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{\text{func-ineq}} \ge 0,$$

$$
\begin{aligned}
&\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-eq}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \end{pmatrix} \quad \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \in \mathbb{R}, \\
&\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{op}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \end{pmatrix} \quad \lambda^{\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \\
&-W + \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\
&\quad + \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \qquad \text{(D-PEP)} \\
&\quad + \sum_{\substack{i\in\mathcal{I}_{\text{op}} \\ o\in\mathcal{O}_i^{\text{op}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \succeq 0, \\
&-w + \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\
&\quad + \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = 0,
\end{aligned}
$$

is feasible for the scalars

$$
\begin{aligned}
&\lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}, \\
&\nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}, \\
&\lambda^{\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)},
\end{aligned}
$$

where

$$
W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = \underbrace{\left(E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}\right)^{\top} M^{\text{func-ineq}}_{(i,o)} E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}}_{\in\mathbb{S}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}}, \tag{6.10}
$$

$$W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = \underbrace{\left(E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}\right)^{\top} M^{\text{func-eq}}_{(i,o)} E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}}_{\in\mathbb{S}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}}, \tag{6.11}$$

$$W^{\underline{k},\,\bar{k},\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = \underbrace{\left(E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}\right)^{\top} M^{\text{op}}_{(i,o)} E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}}_{\in\mathbb{S}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}}, \tag{6.12}$$

$$F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = \underbrace{\begin{bmatrix}\left(F^{\underline{k},\bar{k}}_{(i,j_1,k_1)}\right)^{\top} & \cdots & \left(F^{\underline{k},\bar{k}}_{(i,j_{n_{i,o}},k_{n_{i,o}})}\right)^{\top}\end{bmatrix} a^{\text{func-ineq}}_{(i,o)}}_{\in\mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}}, \tag{6.13}$$

$$F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = \underbrace{\begin{bmatrix}\left(F^{\underline{k},\bar{k}}_{(i,j_1,k_1)}\right)^{\top} & \cdots & \left(F^{\underline{k},\bar{k}}_{(i,j_{n_{i,o}},k_{n_{i,o}})}\right)^{\top}\end{bmatrix} a^{\text{func-eq}}_{(i,o)}}_{\in\mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}}, \tag{6.14}$$

$$E^{\underline{k},\bar{k}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})} = \underbrace{\begin{bmatrix} P_{(i,j_1)}Y^{\underline{k},\bar{k}}_{k_1} \\ \vdots \\ P_{(i,j_{n_{i,o}})}Y^{\underline{k},\bar{k}}_{k_{n_{i,o}}} \\ P_{(i,j_1)}U^{\underline{k},\bar{k}}_{k_1} \\ \vdots \\ P_{(i,j_{n_{i,o}})}U^{\underline{k},\bar{k}}_{k_{n_{i,o}}} \end{bmatrix}}_{\in\mathbb{R}^{2n_{i,o}\times(n+(\bar{k}-\underline{k}+1)\bar{m}+m)}}. \tag{6.15}$$

Furthermore, if the interpolation conditions for $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ *and* $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ *are tight,* $\dim\mathcal{H} \geq n + (\bar{k}-\underline{k}+1)\bar{m} + m$*, and there exists* $G \in \mathbb{S}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}_{++}$ *and* $\boldsymbol{\chi} \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}$ *such that*

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}^{\text{func-ineq}}_i \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}^{\underline{k},\bar{k}}_{i,o} \end{pmatrix} \quad \begin{aligned} &\boldsymbol{\chi}^{\top} F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &+ \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} G) \leq 0, \end{aligned}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}^{\text{func-eq}}_i \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}^{\underline{k},\bar{k}}_{i,o} \end{pmatrix} \quad \begin{aligned} &\boldsymbol{\chi}^{\top} F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &+ \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} G) = 0, \end{aligned}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}^{\text{op}}_i \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}^{\underline{k},\bar{k}}_{i,o} \end{pmatrix} \quad \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} G) \leq 0,$$

then the feasibility of (D-PEP) *is a necessary condition for* $(\text{PEP})^{\star} \leq 0$*.*

Proof. We prove Theorem 6.8 in a sequence of steps:

Formulating a primal semidefinite program. Note that

$$\boldsymbol{u}^{\star} = \begin{cases} 0 & \text{if } m = 1, \\ (N \otimes \mathrm{Id})\hat{\boldsymbol{u}}^{\star} & \text{if } m \geq 2, \end{cases}$$

for N given in (6.2). Moreover, back substitution gives

$$\begin{aligned} \boldsymbol{x}^{\underline{k}+1} &= (A_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{x}^{\underline{k}} + (B_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{u}^{\underline{k}}, \\ (\forall k \in [\![\underline{k}+2, \bar{k}+1]\!]) \quad \boldsymbol{x}^{k} &= (A_{k-1}\cdots A_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{x}^{\underline{k}} \\ &\quad + \sum_{i=\underline{k}}^{k-2} (A_{k-1}\cdots A_{i+1}B_i \otimes \mathrm{Id})\boldsymbol{u}^{i} \\ &\quad + (B_{k-1} \otimes \mathrm{Id})\boldsymbol{u}^{k-1}. \end{aligned}$$

Thus, the constraints of (PEP) can equivalently be written as

$$\begin{aligned}
&\boldsymbol{x}^{\underline{k}} \in \mathcal{H}^{n}, \\
&\boldsymbol{y}^{\underline{k}} = (C_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{x}^{\underline{k}} + (D_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{u}^{\underline{k}}, \\
&\boldsymbol{y}^{\underline{k}+1} = (C_{\underline{k}+1}A_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{x}^{\underline{k}} + (C_{\underline{k}+1}B_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{u}^{\underline{k}} + (D_{\underline{k}+1} \otimes \mathrm{Id})\boldsymbol{u}^{\underline{k}+1}, \\
&\textbf{for each } k \in [\![\underline{k}+2, \bar{k}]\!] \\
&\quad \boldsymbol{y}^{k} = (C_k A_{k-1}\cdots A_{\underline{k}} \otimes \mathrm{Id})\boldsymbol{x}^{\underline{k}} + \sum_{i=\underline{k}}^{k-2} (C_k A_{k-1}\cdots A_{i+1}B_i \otimes \mathrm{Id})\boldsymbol{u}^{i} \\
&\qquad\quad + (C_k B_{k-1} \otimes \mathrm{Id})\boldsymbol{u}^{k-1} + (D_k \otimes \mathrm{Id})\boldsymbol{u}^{k}, \\
&\textbf{end} \\
&\textbf{for each } k \in [\![\underline{k}, \bar{k}]\!] \\
&\quad \boldsymbol{u}^{k} = (\boldsymbol{u}_1^{k}, \ldots, \boldsymbol{u}_m^{k}) \in \prod_{i=1}^{m} \mathcal{H}^{\bar{m}_i}, \\
&\quad \boldsymbol{y}^{k} = (\boldsymbol{y}_1^{k}, \ldots, \boldsymbol{y}_m^{k}) \in \prod_{i=1}^{m} \mathcal{H}^{\bar{m}_i}, \\
&\quad (\boldsymbol{u}_i^{k})_{i\in\mathcal{I}_{\mathrm{func}}} \in \prod_{i\in\mathcal{I}_{\mathrm{func}}} \boldsymbol{\partial f}_i(\boldsymbol{y}_i^{k}), \\
&\quad (\boldsymbol{u}_i^{k})_{i\in\mathcal{I}_{\mathrm{op}}} \in \prod_{i\in\mathcal{I}_{\mathrm{op}}} \boldsymbol{G}_i(\boldsymbol{y}_i^{k}), \\
&\quad \boldsymbol{F}^{k} = (\boldsymbol{f}_i(\boldsymbol{y}_i^{k}))_{i\in\mathcal{I}_{\mathrm{func}}} \in \mathbb{R}^{\bar{m}_{\mathrm{func}}}, \\
&\textbf{end} \\
&\boldsymbol{u}^{\star} = (u_1^{\star}, \ldots, u_m^{\star}) = \begin{cases} 0 & \text{if } m = 1, \\ (N \otimes \mathrm{Id})\hat{\boldsymbol{u}}^{\star} & \text{if } m \geq 2, \text{ where } \hat{\boldsymbol{u}}^{\star} \in \mathcal{H}^{m-1}, \end{cases} \\
&y^{\star} \in \mathcal{H}, \\
&(u_i^{\star})_{i\in\mathcal{I}_{\mathrm{func}}} \in \prod_{i\in\mathcal{I}_{\mathrm{func}}} \partial f_i(y^{\star}), \\
&(u_i^{\star})_{i\in\mathcal{I}_{\mathrm{op}}} \in \prod_{i\in\mathcal{I}_{\mathrm{op}}} G_i(y^{\star}),
\end{aligned}$$

$$\boldsymbol{F}^\star = (\boldsymbol{f}_i(y^\star))_{i\in\mathcal{I}_{\text{func}}} \in \mathbb{R}^{m_{\text{func}}},$$
$$(f_i)_{i\in\mathcal{I}_{\text{func}}} \in \prod_{i\in\mathcal{I}_{\text{func}}} \mathcal{F}_i,$$
$$(G_i)_{i\in\mathcal{I}_{\text{op}}} \in \prod_{i\in\mathcal{I}_{\text{op}}} \mathcal{G}_i,$$

or equivalently

$$\begin{aligned}
&\textbf{for each } i \in \mathcal{I}_{\text{func}}\\
&\quad \textbf{for each } j \in [\![1, \bar{m}_i]\!]\\
&\quad\quad \textbf{for each } k \in [\![\underline{k}, \bar{k}]\!]\\
&\quad\quad\quad (P_{(i,j)} U_k^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta} \in \partial f_i((P_{(i,j)} Y_k^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}),\\
&\quad\quad\quad F_{(i,j,k)}^{\underline{k},\bar{k}} \boldsymbol{\chi} = f_i((P_{(i,j)} Y_k^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}),\\
&\quad\quad \textbf{end}\\
&\quad \textbf{end}\\
&\quad (P_{(i,\star)} U_\star^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta} \in \partial f_i((P_{(i,\star)} Y_\star^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}),\\
&\quad F_{(i,\star,\star)}^{\underline{k},\bar{k}} \boldsymbol{\chi} = f_i((P_{(i,\star)} Y_\star^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}),\\
&\quad f_i \in \mathcal{F}_i,\\
&\textbf{end}\\
&\textbf{for each } i \in \mathcal{I}_{\text{op}}\\
&\quad \textbf{for each } j \in [\![1, \bar{m}_i]\!]\\
&\quad\quad \textbf{for each } k \in [\![\underline{k}, \bar{k}]\!]\\
&\quad\quad\quad (P_{(i,j)} U_k^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta} \in G_i((P_{(i,j)} Y_k^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}),\\
&\quad\quad \textbf{end}\\
&\quad \textbf{end}\\
&\quad (P_{(i,\star)} U_\star^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta} \in G_i((P_{(i,\star)} Y_\star^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}),\\
&\quad G_i \in \mathcal{G}_i,\\
&\textbf{end}\\
&\boldsymbol{\zeta} = (\boldsymbol{x}^{\underline{k}}, \boldsymbol{u}^{\underline{k}}, \ldots, \boldsymbol{u}^{\bar{k}}, \hat{\boldsymbol{u}}^\star, y^\star) \in \mathcal{H}^n \times (\prod_{i=\underline{k}}^{\bar{k}} \mathcal{H}^{\bar{m}}) \times \mathcal{H}^{m-1} \times \mathcal{H},\\
&\boldsymbol{\chi} = (\boldsymbol{F}^{\underline{k}}, \ldots, \boldsymbol{F}^{\bar{k}}, \boldsymbol{F}^\star) \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}},
\end{aligned} \tag{6.16}$$

where we have used (6.4), (6.5), (6.6), (6.7), (6.8), and (6.9). Using Assumption 6.7, we get the following relaxation (equivalent representation if the interpolation conditions for $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ and $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ are tight) of (6.16):

$$\begin{aligned}
&\textbf{for each } i \in \mathcal{I}_{\text{func}}\\
&\quad \textbf{for each } o \in \mathcal{O}_i^{\text{func-ineq}}
\end{aligned}$$

$$
\begin{aligned}
&\quad\quad\textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}\\
&\quad\quad\quad (F_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{\underline{k},\,\bar{k},\,\text{func-ineq}})^\top \boldsymbol{\chi}\\
&\quad\quad\quad + \mathcal{Q}(M_{(i,o)}^{\text{func-ineq}}, (E_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}) \leq 0,\\
&\quad\quad\textbf{end}\\
&\quad\textbf{end}\\
&\quad\textbf{for each } o \in \mathcal{O}_i^{\text{func-eq}}\\
&\quad\quad\textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}\\
&\quad\quad\quad (F_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{\underline{k},\,\bar{k},\,\text{func-eq}})^\top \boldsymbol{\chi}\\
&\quad\quad\quad + \mathcal{Q}(M_{(i,o)}^{\text{func-eq}}, (E_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}) = 0,\\
&\quad\quad\textbf{end}\\
&\quad\textbf{end}\\
&\textbf{end}\\
&\textbf{for each } i \in \mathcal{I}_{\text{op}}\\
&\quad\textbf{for each } o \in \mathcal{O}_i^{\text{op}}\\
&\quad\quad\textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}\\
&\quad\quad\quad \mathcal{Q}(M_{(i,o)}^{\text{op}}, (E_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}})}^{\underline{k},\bar{k}} \otimes \mathrm{Id})\boldsymbol{\zeta}) \leq 0,\\
&\quad\quad\textbf{end}\\
&\quad\textbf{end}\\
&\textbf{end}\\
&\boldsymbol{\zeta} \in \mathcal{H}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m},\\
&\boldsymbol{\chi} \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}},
\end{aligned}
\tag{6.17}
$$

where we have used (6.13), (6.14), and (6.15). If we use (6.10), (6.11), and (6.12), the constraints in (6.17) can equivalently be written as

$$
\begin{aligned}
&\textbf{for each } i \in \mathcal{I}_{\text{func}}\\
&\quad\textbf{for each } o \in \mathcal{O}_i^{\text{func-ineq}}\\
&\quad\quad\textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}\\
&\quad\quad\quad (F_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{\underline{k},\,\bar{k},\,\text{func-ineq}})^\top \boldsymbol{\chi}\\
&\quad\quad\quad + \operatorname{trace}(W_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{\underline{k},\,\bar{k},\,\text{func-ineq}} \mathtt{G}(\boldsymbol{\zeta})) \leq 0,\\
&\quad\quad\textbf{end}\\
&\quad\textbf{end}\\
&\quad\textbf{for each } o \in \mathcal{O}_i^{\text{func-eq}}
\end{aligned}
$$

$$
\begin{aligned}
&\quad\quad \textbf{for each } ((j_1, k_1), \ldots, (j_{n_{i,o}}, k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
&\quad\quad\quad (F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)})^\top \boldsymbol{\chi} \\
&\quad\quad\quad + \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \mathtt{G}(\boldsymbol{\zeta})) = 0, \qquad (6.18) \\
&\quad\quad \textbf{end} \\
&\quad \textbf{end} \\
&\textbf{end} \\
&\textbf{for each } i \in \mathcal{I}_{\text{op}} \\
&\quad \textbf{for each } o \in \mathcal{O}_i^{\text{op}} \\
&\quad\quad \textbf{for each } ((j_1, k_1), \ldots, (j_{n_{i,o}}, k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
&\quad\quad\quad \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \mathtt{G}(\boldsymbol{\zeta})) \leq 0, \\
&\quad\quad \textbf{end} \\
&\quad \textbf{end} \\
&\textbf{end} \\
&\boldsymbol{\zeta} \in \mathcal{H}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}, \\
&\boldsymbol{\chi} \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}.
\end{aligned}
$$

Next, we consider the objective function of (PEP). It can be written as

$$
\begin{aligned}
&\mathcal{Q}(W, (\boldsymbol{x}^{\underline{k}}, \boldsymbol{u}^{\underline{k}}, \ldots, \boldsymbol{u}^{\bar{k}}, \hat{\boldsymbol{u}}^\star, y^\star)) + w^\top (\boldsymbol{F}^{\underline{k}}, \ldots, \boldsymbol{F}^{\bar{k}}, \boldsymbol{F}^\star) \\
&= \operatorname{trace}(W \mathtt{G}(\boldsymbol{\zeta})) + w^\top \boldsymbol{\chi}.
\end{aligned}
$$

If we combine this observation about the objective function of (PEP) with the (possibly relaxed) constraints in (6.18), we conclude that a (possibly relaxed) version of (PEP) can be written as

$$
\begin{aligned}
\text{maximize} \quad & \operatorname{trace}(W \mathtt{G}(\boldsymbol{\zeta})) + w^\top \boldsymbol{\chi} \\
\text{subject to} \quad & \textbf{for each } i \in \mathcal{I}_{\text{func}} \\
& \quad \textbf{for each } o \in \mathcal{O}_i^{\text{func-ineq}} \\
& \quad\quad \textbf{for each } ((j_1, k_1), \ldots, (j_{n_{i,o}}, k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
& \quad\quad\quad (F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)})^\top \boldsymbol{\chi} \\
& \quad\quad\quad + \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \mathtt{G}(\boldsymbol{\zeta})) \leq 0, \\
& \quad\quad \textbf{end} \\
& \quad \textbf{end} \\
& \quad \textbf{for each } o \in \mathcal{O}_i^{\text{func-eq}} \\
& \quad\quad \textbf{for each } ((j_1, k_1), \ldots, (j_{n_{i,o}}, k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
& \quad\quad\quad (F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)})^\top \boldsymbol{\chi}
\end{aligned}
$$

$$+ \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}\mathtt{G}(\boldsymbol{\zeta})) = 0, \tag{6.19}$$

end

end

end

for each $i \in \mathcal{I}_{\text{op}}$

for each $o \in \mathcal{O}_i^{\text{op}}$

for each $((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}$

$$\operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}\mathtt{G}(\boldsymbol{\zeta})) \leq 0,$$

end

end

end

$\boldsymbol{\zeta} \in \mathcal{H}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}$,

$\boldsymbol{\chi} \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}$.

The problem

maximize $\operatorname{trace}(WG) + w^\top \boldsymbol{\chi}$

subject to **for each** $i \in \mathcal{I}_{\text{func}}$

for each $o \in \mathcal{O}_i^{\text{func-ineq}}$

for each $((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}$

$$\left(F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}\right)^\top \boldsymbol{\chi}$$

$$+ \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}G) \leq 0, \tag{6.20a}$$

end

end

for each $o \in \mathcal{O}_i^{\text{func-eq}}$

for each $((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}$

$$\left(F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}\right)^\top \boldsymbol{\chi}$$

$$+ \operatorname{trace}(W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}G) = 0, \tag{6.20b}$$

end

end

end

for each $i \in \mathcal{I}_{\text{op}}$

for each $o \in \mathcal{O}_i^{\text{op}}$

for each $((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}$

$$\operatorname{trace}(W^{\underline{k},\,\bar{k},\,\mathrm{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}G) \leq 0, \tag{6.20c}$$

end

end

end

$$G \in \mathbb{S}_+^{n+(\bar{k}-\underline{k}+1)\bar{m}+m},$$

$$\chi \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}}}.$$

is a relaxation of (6.19), and therefore, has optimal value greater or equal to (PEP)*.

We will make use of the following fact: If $\dim \mathcal{H} \geq k$, then $G \in \mathbb{S}_+^k$ if and only if there exists $\boldsymbol{z} \in \mathcal{H}^k$ such that $G = \mathtt{G}(\boldsymbol{z})$. [34, Lemma 3.1] shows the result for the case $k = 4$ and is based on the Cholesky decomposition of positive semidefinite matrices. The general case is a straightforward extension. This fact implies that if $\dim \mathcal{H} \geq n + (\bar{k} - \underline{k} + 1)\bar{m} + m$, then (6.20) has optimal value equal to (6.19). Note that (6.20) is a convex semidefinite program.

Dual problem. For (6.20a), (6.20b), and (6.20c), we introduce corresponding dual variables

$$\lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0,$$
$$\nu^{\text{func-in}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \in \mathbb{R},$$
$$\lambda^{\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0,$$

respectively. With this, the objective function of the Lagrange dual problem of (6.20) becomes

$$\sup_{G \in \mathbb{S}_+^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}} \operatorname{trace}\Bigg(\Bigg(W - \sum_{\substack{i \in \mathcal{I}_{\mathrm{func}} \\ o \in \mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}$$
$$- \sum_{\substack{i \in \mathcal{I}_{\mathrm{func}} \\ o \in \mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}$$
$$- \sum_{\substack{i \in \mathcal{I}_{\mathrm{op}} \\ o \in \mathcal{O}_i^{\mathrm{op}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\mathrm{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\mathrm{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}\Bigg) G\Bigg)$$

$$
\begin{aligned}
&+ \sup_{\chi \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}} \Bigg(w \\
&\quad - \sum_{\substack{i \in \mathcal{I}_{\text{func}} \\ o \in \mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\
&\quad - \sum_{\substack{i \in \mathcal{I}_{\text{func}} \\ o \in \mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \Bigg)^{\top} \chi.
\end{aligned}
$$

Since the dual problem is a minimization problem over the dual variables, we conclude that it can be written as

$$
\begin{array}{ll}
\text{minimize} & 0 \\
\text{subject to} & \textbf{for each } i \in \mathcal{I}_{\text{func}} \\
& \quad \textbf{for each } o \in \mathcal{O}_i^{\text{func-ineq}} \\
& \quad\quad \textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
& \quad\quad\quad \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \\
& \quad\quad \textbf{end} \\
& \quad \textbf{end} \\
& \quad \textbf{for each } o \in \mathcal{O}_i^{\text{func-eq}} \\
& \quad\quad \textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
& \quad\quad\quad \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \in \mathbb{R}, \\
& \quad\quad \textbf{end} \\
& \quad \textbf{end} \\
& \textbf{end} \\
& \textbf{for each } i \in \mathcal{I}_{\text{op}} \\
& \quad \textbf{for each } o \in \mathcal{O}_i^{\text{op}} \\
& \quad\quad \textbf{for each } ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{\underline{k},\bar{k}} \\
& \quad\quad\quad \lambda^{\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \\
& \quad\quad \textbf{end} \\
& \quad \textbf{end} \\
& \textbf{end}
\end{array}
\tag{6.21}
$$

$$
\begin{aligned}
&- W + \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\
&+ \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\
&+ \sum_{\substack{i\in\mathcal{I}_{\text{op}} \\ o\in\mathcal{O}_i^{\text{op}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{\underline{k},\,\bar{k},\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \succeq 0, \\
&- w + \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \lambda^{\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{\underline{k},\,\bar{k},\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\
&+ \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{\underline{k},\bar{k}}}} \nu^{\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{\underline{k},\,\bar{k},\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = 0. \quad \square
\end{aligned}
$$

which is a feasibility problem. Since (6.21) is the dual of (6.20), we conclude that if (6.21) is feasible, then $(\text{PEP})^\star \leq 0$.

Next, suppose that the primal problem (6.20) has a Slater point, i.e., there exists $G \in \mathbb{S}_{++}^{n+(\bar{k}-\underline{k}+1)\bar{m}+m}$ and $\chi \in \mathbb{R}^{(\bar{k}-\underline{k}+1)\bar{m}_{\text{func}}+m_{\text{func}}}$ such that (6.20a), (6.20b), and (6.20c) hold. Then there is no duality gap, i.e., strong duality holds, between the primal problem (6.20) and the dual problem (6.21).

This concludes the proof.

6.3 Automated iteration-independent analysis

This section provides a necessary and sufficient condition, in terms of the feasibility of a semidefinite program, for the existence of a quadratic Lyapunov inequality in the sense of Definition 4.1.

Theorem 6.9

Suppose that Assumption 3.1, Assumption 6.7, and (4.1) hold, $\rho \in [0,1]$ is a contraction factor, $h \in \mathbb{N}_0$ is a history parameter, and $\alpha \in \mathbb{N}_0$ is an overlap parameter. Let $P \in \mathbb{S}^{n+(h+1)\bar{m}+m}$ and $p \in \mathbb{R}^{(h+1)\bar{m}_{\text{func}}+m_{\text{func}}}$ such that

$$(\forall k \in \mathbb{N}_0) \qquad \mathcal{V}(P,p,k) \geq 0,$$

where $\mathcal{V}$ is defined in (4.3), and $T \in \mathbb{S}^{n+(h+\alpha+2)\bar{m}+m}$ and $t \in \mathbb{R}^{(h+\alpha+2)\bar{m}_{\text{func}}+m_{\text{func}}}$ such that

$$(\forall k \in \mathbb{N}_0) \qquad \mathcal{R}(T,t,k) \geq 0,$$

where $\mathcal{R}$ is defined in (4.4). Then a sufficient condition for there to exists (Q, q, S, s) that satisfies a $(P, p, T, t, \rho, h, \alpha)$-quadratic Lyapunov inequality for algorithm (3.2) over the problem class defined by $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ and $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ (see Definition 4.1 and recall that (C4) is optional and may be omitted) is that the following system of constraints

for each cond $\in \{\text{C1}, \text{C2}, \text{C3}, \text{C4}\}$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-ineq}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}} \end{pmatrix} \qquad \lambda^{\text{func-ineq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \tag{6.22a}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-eq}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}} \end{pmatrix} \qquad \nu^{\text{func-eq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \in \mathbb{R}, \tag{6.22b}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{op}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}} \end{pmatrix} \qquad \lambda^{\text{op, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \tag{6.22c}$$

$$\begin{aligned} -W^{\text{cond}} &+ \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}}}} \lambda^{\text{func-ineq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{0,\bar{k}_{\text{cond}},\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &+ \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}}}} \nu^{\text{func-eq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{0,\bar{k}_{\text{cond}},\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &+ \sum_{\substack{i\in\mathcal{I}_{\text{op}} \\ o\in\mathcal{O}_i^{\text{op}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}}}} \lambda^{\text{op, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{0,\bar{k}_{\text{cond}},\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \succeq 0, \end{aligned} \tag{6.22d}$$

$$\begin{aligned} -w^{\text{cond}} &+ \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}}}} \lambda^{\text{func-ineq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{0,\bar{k}_{\text{cond}},\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &+ \sum_{\substack{i\in\mathcal{I}_{\text{func}} \\ o\in\mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}}))\in\mathcal{J}_{i,o}^{0,\bar{k}_{\text{cond}}}}} \nu^{\text{func-eq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{0,\bar{k}_{\text{cond}},\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = 0, \end{aligned} \tag{6.22e}$$

end

$$Q \in \mathbb{S}^{n+(h+1)\bar{m}+m}, \tag{6.22f}$$

$$q \in \mathbb{R}^{(h+1)\bar{m}_{\text{func}}+m_{\text{func}}}, \tag{6.22g}$$

$$S \in \mathbb{S}^{n+(h+\alpha+2)\bar{m}+m}, \tag{6.22h}$$

$$s \in \mathbb{R}^{(h+\alpha+2)\bar{m}_{\text{func}}+m_{\text{func}}} \tag{6.22i}$$

is feasible for the scalars

$$\lambda^{\text{func-ineq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)},$$
$$\nu^{\text{func-eq, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)},$$
$$\lambda^{\text{op, cond}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)},$$

the matrices Q and S, and the vectors q and s, where

$$W^{\text{C1}} = (\Theta_1^{\text{C1}})^\top Q \Theta_1^{\text{C1}} - \rho (\Theta_0^{\text{C1}})^\top Q \Theta_0^{\text{C1}} + S, \tag{6.23}$$

$$w^{\text{C1}} = (\theta_1^{\text{C1}} - \rho \theta_0^{\text{C1}})^\top q + s, \tag{6.24}$$

$$W^{\text{C2}} = P - Q, \tag{6.25}$$

$$w^{\text{C2}} = p - q, \tag{6.26}$$

$$W^{\text{C3}} = T - S, \tag{6.27}$$

$$w^{\text{C3}} = t - s, \tag{6.28}$$

$$W^{\text{C4}} = (\Theta_1^{\text{C4}})^\top S \Theta_1^{\text{C4}} - (\Theta_0^{\text{C4}})^\top S \Theta_0^{\text{C4}}, \tag{6.29}$$

$$w^{\text{C4}} = (\theta_1^{\text{C4}} - \theta_0^{\text{C4}})^\top s, \tag{6.30}$$

$$\bar{k}_{\text{cond}} = \begin{cases} h+\alpha+1 & \text{if cond} \in \{\text{C1}, \text{C3}\}, \\ h & \text{if cond} \in \{\text{C2}\}, \\ h+\alpha+2 & \text{if cond} \in \{\text{C4}\}, \end{cases}$$

and

$$\Theta_0^{\text{C1}} = \underbrace{\begin{bmatrix} I_{n+(h+1)\bar{m}} & 0_{(n+(h+1)\bar{m})\times(\alpha+1)\bar{m}} & 0_{(n+(h+1)\bar{m})\times m} \\ 0_{m\times(n+(h+1)\bar{m})} & 0_{m\times(\alpha+1)\bar{m}} & I_m \end{bmatrix}}_{\in\mathbb{R}^{(n+(h+1)\bar{m}+m)\times(n+(h+\alpha+2)\bar{m}+m)}}, \tag{6.31}$$

$$\theta_0^{\text{C1}} = \underbrace{\begin{bmatrix} I_{(h+1)\bar{m}_{\text{func}}} & 0_{(h+1)\bar{m}_{\text{func}}\times(\alpha+1)\bar{m}_{\text{func}}} & 0_{(h+1)\bar{m}_{\text{func}}\times m_{\text{func}}} \\ 0_{m_{\text{func}}\times(h+1)\bar{m}_{\text{func}}} & 0_{m_{\text{func}}\times(\alpha+1)\bar{m}_{\text{func}}} & I_{m_{\text{func}}} \end{bmatrix}}_{\in\mathbb{R}^{((h+1)\bar{m}_{\text{func}}+m_{\text{func}})\times((h+\alpha+2)\bar{m}_{\text{func}}+m_{\text{func}})}}, \tag{6.32}$$

$$\Theta_1^{\text{C1}} = \underbrace{\left[\begin{array}{c|c} \multicolumn{2}{c}{X^{0,h+\alpha+1}_{\alpha+1}} \\ \hline 0_{((h+1)\bar{m}+m)\times(n+(\alpha+1)\bar{m})} & I_{(h+1)\bar{m}+m} \end{array}\right]}_{\in\mathbb{R}^{(n+(h+1)\bar{m}+m)\times(n+(h+\alpha+2)\bar{m}+m)}}, \tag{6.33}$$

$$\theta_1^{\text{C1}} = \underbrace{\begin{bmatrix} 0_{((h+1)\bar{m}_{\text{func}}+m_{\text{func}})\times(\alpha+1)\bar{m}_{\text{func}}} & I_{(h+1)\bar{m}_{\text{func}}+m_{\text{func}}} \end{bmatrix}}_{\in\mathbb{R}^{((h+1)\bar{m}_{\text{func}}+m_{\text{func}})\times((h+\alpha+2)\bar{m}_{\text{func}}+m_{\text{func}})}}, \tag{6.34}$$

$$\Theta_0^{\text{C4}} = \underbrace{\begin{bmatrix} I_{n+(h+\alpha+2)\bar{m}} & 0_{(n+(h+\alpha+2)\bar{m})\times\bar{m}} & 0_{(n+(h+\alpha+2)\bar{m})\times m} \\ 0_{m\times(n+(h+\alpha+2)\bar{m})} & 0_{m\times\bar{m}} & I_m \end{bmatrix}}_{\in\mathbb{R}^{(n+(h+\alpha+2)\bar{m}+m)\times(n+(h+\alpha+3)\bar{m}+m)}}, \tag{6.35}$$

$$\theta_0^{\mathrm{C4}} = \underbrace{\begin{bmatrix} I_{(h+\alpha+2)\bar{m}_{\mathrm{func}}} & 0_{(h+\alpha+2)\bar{m}_{\mathrm{func}}\times\bar{m}_{\mathrm{func}}} & 0_{(h+\alpha+2)\bar{m}_{\mathrm{func}}\times m_{\mathrm{func}}} \\ 0_{m_{\mathrm{func}}\times(h+\alpha+2)\bar{m}_{\mathrm{func}}} & 0_{m_{\mathrm{func}}\times\bar{m}_{\mathrm{func}}} & I_{m_{\mathrm{func}}} \end{bmatrix}}_{\in\mathbb{R}^{((h+\alpha+2)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}})\times((h+\alpha+3)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}})}}, \tag{6.36}$$

$$\Theta_1^{\mathrm{C4}} = \underbrace{\left[\begin{array}{c|c} \multicolumn{2}{c}{X_1^{0,h+\alpha+2}} \\ \hline 0_{((h+\alpha+2)\bar{m}+m)\times(n+\bar{m})} & I_{(h+\alpha+2)\bar{m}+m} \end{array}\right]}_{\in\mathbb{R}^{(n+(h+\alpha+2)\bar{m}+m)\times(n+(h+\alpha+3)\bar{m}+m)}}, \tag{6.37}$$

$$\theta_1^{\mathrm{C4}} = \underbrace{\begin{bmatrix} 0_{((h+\alpha+2)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}})\times\bar{m}_{\mathrm{func}}} & I_{(h+\alpha+2)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}}} \end{bmatrix}}_{\in\mathbb{R}^{((h+\alpha+2)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}})\times((h+\alpha+3)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}})}}. \tag{6.38}$$

Furthermore, if the interpolation conditions for $(\mathcal{F}_i)_{i\in\mathcal{I}_{\mathrm{func}}}$ and $(\mathcal{G}_i)_{i\in\mathcal{I}_{\mathrm{op}}}$ are tight,

for each cond $\in$ {C1, C2, C3, C4}

$\quad \dim\mathcal{H} \geq n + (\bar{k}_{\mathrm{cond}}+1)\bar{m} + m,$

end

and there exists

for each cond $\in$ {C1, C2, C3, C4}

$\quad G_{\mathrm{cond}} \in \mathbb{S}_{++}^{n+(\bar{k}_{\mathrm{cond}}+1)\bar{m}+m},$

$\quad \boldsymbol{\chi}_{\mathrm{cond}} \in \mathbb{R}^{(\bar{k}_{\mathrm{cond}}+1)\bar{m}_{\mathrm{func}}+m_{\mathrm{func}}},$

end

such that

for each cond $\in$ {C1, C2, C3, C4}

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\mathrm{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-ineq}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{0,\bar{k}_{\mathrm{cond}}} \end{pmatrix} \quad \begin{aligned} &\boldsymbol{\chi}_{\mathrm{cond}}^\top F_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{0,\bar{k}_{\mathrm{cond}},\text{func-ineq}} \\ &+ \mathrm{trace}(W_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{0,\bar{k}_{\mathrm{cond}},\text{func-ineq}} G_{\mathrm{cond}}) \leq 0, \end{aligned}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\mathrm{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-eq}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{0,\bar{k}_{\mathrm{cond}}} \end{pmatrix} \quad \begin{aligned} &\boldsymbol{\chi}_{\mathrm{cond}}^\top F_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{0,\bar{k}_{\mathrm{cond}},\text{func-eq}} \\ &+ \mathrm{trace}(W_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{0,\bar{k}_{\mathrm{cond}},\text{func-eq}} G_{\mathrm{cond}}) = 0, \end{aligned}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\mathrm{func}} \\ \forall o \in \mathcal{O}_i^{\mathrm{op}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{0,\bar{k}_{\mathrm{cond}}} \end{pmatrix} \quad \mathrm{trace}(W_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)}^{0,\bar{k}_{\mathrm{cond}},\mathrm{op}} G_{\mathrm{cond}}) \leq 0,$$

end

then (6.22) *is also a necessary condition.*

Proof. By induction, we only need to consider the case $k=0$ in (C1) to (C4). Additionally, note that

$$(\boldsymbol{x}^0,\boldsymbol{u}^0,\ldots,\boldsymbol{u}^h,\hat{\boldsymbol{u}}^\star,y^\star) = (\Theta_0^{\mathrm{C1}}\otimes\mathrm{Id})(\boldsymbol{x}^0,\boldsymbol{u}^0,\ldots,\boldsymbol{u}^{h+\alpha+1},\hat{\boldsymbol{u}}^\star,y^\star),$$
$$(\boldsymbol{F}^0,\ldots,\boldsymbol{F}^h,\boldsymbol{F}^\star) = \theta_0^{\mathrm{C1}}(\boldsymbol{F}^0,\ldots,\boldsymbol{F}^{h+\alpha+1},\boldsymbol{F}^\star),$$

$$
\begin{aligned}
(\boldsymbol{x}^{\alpha+1}, \boldsymbol{u}^{\alpha+1}, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star) &= (\Theta_1^{\mathrm{C1}} \otimes \mathrm{Id})(\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star), \\
(\boldsymbol{F}^{\alpha+1}, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) &= \theta_1^{\mathrm{C1}}(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star), \\
(\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star) &= (\Theta_0^{\mathrm{C4}} \otimes \mathrm{Id})(\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+2}, \hat{\boldsymbol{u}}^\star, y^\star), \\
(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) &= \theta_0^{\mathrm{C4}}(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+2}, \boldsymbol{F}^\star), \\
(\boldsymbol{x}^1, \boldsymbol{u}^1, \ldots, \boldsymbol{u}^{h+\alpha+2}, \hat{\boldsymbol{u}}^\star, y^\star) &= (\Theta_1^{\mathrm{C4}} \otimes \mathrm{Id})(\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+2}, \hat{\boldsymbol{u}}^\star, y^\star), \\
(\boldsymbol{F}^1, \ldots, \boldsymbol{F}^{h+\alpha+2}, \boldsymbol{F}^\star) &= \theta_1^{\mathrm{C4}}(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+2}, \boldsymbol{F}^\star),
\end{aligned}
$$

where we have used (6.31), (6.32), (6.33), (6.34), (6.35), (6.36), (6.37), and (6.38), respectively.

First, suppose that the parameters (Q, q, S, s) are fixed. Note that

$$
\begin{aligned}
&\mathcal{V}(Q, q, \alpha+1) - \rho\mathcal{V}(Q, q, 0) + \mathcal{R}(S, s, 0) \\
&= \mathcal{Q}(Q, (\boldsymbol{x}^{\alpha+1}, \boldsymbol{u}^{\alpha+1}, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + q^\top(\boldsymbol{F}^{\alpha+1}, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) \\
&\quad - \rho\mathcal{Q}(Q, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^h, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad - \rho q^\top(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^h, \boldsymbol{F}^\star) \\
&\quad + \mathcal{Q}(S, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + s^\top(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) \\
&= \mathcal{Q}((\Theta_1^{\mathrm{C1}})^\top Q \Theta_1^{\mathrm{C1}} - \rho(\Theta_0^{\mathrm{C1}})^\top Q \Theta_0^{\mathrm{C1}} + S, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (q^\top(\theta_1^{\mathrm{C1}} - \rho\theta_0^{\mathrm{C1}}) + s^\top)(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) \\
&= \mathcal{Q}(W^{\mathrm{C1}}, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (w^{\mathrm{C1}})^\top(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star)
\end{aligned} \tag{6.39}
$$

where (6.23) and (6.24) are used in the last equality. Therefore, using (6.39) as the objective function in (PEP), Theorem 6.8 gives that (6.22), with cond = C1, is a sufficient condition for (C1). Note that

$$
\begin{aligned}
&\mathcal{V}(P, p, 0) - \mathcal{V}(Q, q, 0) \\
&= \mathcal{Q}(P - Q, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^h, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (p - q)^\top(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^h, \boldsymbol{F}^\star) \\
&= \mathcal{Q}(W^{\mathrm{C2}}, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^h, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (w^{\mathrm{C2}})^\top(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^h, \boldsymbol{F}^\star)
\end{aligned} \tag{6.40}
$$

where (6.25) and (6.26) is used in the last equality. Therefore, using the (6.40) as the objective function in (PEP), Theorem 6.8 gives that (6.22), with cond = C2, is a sufficient condition for (C2). Note that

$$\begin{aligned}
&\mathcal{R}(T,t,0) - \mathcal{R}(S,s,0) \\
&= \mathcal{Q}(T-S, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (t-s)^\top (\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) \\
&= \mathcal{Q}(W^{\mathrm{C3}}, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (w^{\mathrm{C3}})^\top (\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star)
\end{aligned} \tag{6.41}$$

where (6.27) and (6.28) is used in the last equality. Therefore, using the (6.41) as the objective function in (PEP), Theorem 6.8 gives that (6.22), with cond = C3, is a sufficient condition for (C3). Note that

$$\begin{aligned}
&\mathcal{R}(S,s,1) - \mathcal{R}(S,s,0) \\
&= \mathcal{Q}(S, (\boldsymbol{x}^1, \boldsymbol{u}^1, \ldots, \boldsymbol{u}^{h+\alpha+2}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + s^\top (\boldsymbol{F}^1, \ldots, \boldsymbol{F}^{h+\alpha+2}, \boldsymbol{F}^\star) \\
&\quad - \mathcal{Q}(S, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad - s^\top (\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+1}, \boldsymbol{F}^\star) \\
&= \mathcal{Q}((\Theta_1^{\mathrm{C4}})^\top S \Theta_1^{\mathrm{C4}} - (\Theta_0^{\mathrm{C4}})^\top S \Theta_0^{\mathrm{C4}}, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+2}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (s^\top (\theta_1^{\mathrm{C4}} - \theta_0^{\mathrm{C4}}))(\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+2}, \boldsymbol{F}^\star) \\
&= \mathcal{Q}(W^{\mathrm{C4}}, (\boldsymbol{x}^0, \boldsymbol{u}^0, \ldots, \boldsymbol{u}^{h+\alpha+2}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (w^{\mathrm{C4}})^\top (\boldsymbol{F}^0, \ldots, \boldsymbol{F}^{h+\alpha+2}, \boldsymbol{F}^\star)
\end{aligned} \tag{6.42}$$

where (6.29) and (6.30) is used in the last equality. Therefore, using the (6.42) as the objective function in (PEP), Theorem 6.8 gives that (6.22), with cond = C4, is a sufficient condition for (C4).

Second, note that the proof is complete if we let the parameters (Q, q, S, s) free, as in (6.22f) to (6.22i). □

6.3.1 Linear convergence

The class `LinearConvergence`, accessible through `IterationIndependent` as a class attribute, contains helper functions for creating parameters (P, p, T, t) in Definition 4.1 in Section 4.1. In particular, in the case of linear convergence, recall that the lower bound $(\mathcal{V}(P,p,k))_{k\in\mathbb{N}_0}$ is the relevant quality, and we can simply set the other lower bound $(\mathcal{R}(T,t,k))_{k\in\mathbb{N}_0}$ to zero by choosing $T = 0$ and $t = 0$. Setting this second lower bound to zero provides the greatest flexibility when analyzing linear convergence.

We provide examples of these helper functions below, and use the notation given in (6.2) to (6.9).

Distance to solution. The method below returns

$$\begin{cases}
P = (P_{(i,j)} Y_\tau^{0,h} - P_{(i,\star)} Y_\star^{0,h})^\top (P_{(i,j)} Y_\tau^{0,h} - P_{(i,\star)} Y_\star^{0,h}), \\
p = 0, \\
T = 0, \\
t = 0,
\end{cases} \tag{6.43}$$

where $\tau \in [\![0, h]\!]$, $j \in [\![1, \bar{m}_i]\!]$, and $i \in \mathcal{I}_{\text{func}} \cup \mathcal{I}_{\text{op}}$. This gives

$$\mathcal{V}(P, p, k) = \|y_{i,j}^{k+\tau} - y^\star\|^2,$$

which measures the squared distance to the solution.

```
class LinearConvergence:
    @staticmethod
    def get_parameters_distance_to_solution(
            algo: Type[Algorithm],
            h: int = 0,
            alpha: int = 0,
            i: int = 1,
            j: int = 1,
            tau: int = 0
        ) -> Union[
            Tuple[np.ndarray, np.ndarray],
            Tuple[np.ndarray, np.ndarray, np.ndarray, np.ndarray]
            ]:
        # Returns (P,T) in (6.43) when I_func = ∅, otherwise (P,p,T,t).
        # ... (omitted for brevity) ...
```

Function-value suboptimality. The method below requires $m = m_{\text{func}} = 1$ and returns

$$\begin{cases} P = 0, \\ p = \left(F_{(1,j,\tau)}^{0,h} - F_{(1,\star,\star)}^{0,h}\right)^\top, \\ T = 0, \\ t = 0, \end{cases} \tag{6.44}$$

where $\tau \in [\![0, h]\!]$ and $j \in [\![1, \bar{m}_1]\!]$. This gives

$$\mathcal{V}(P, p, k) = f_1(y_{1,j}^{k+\tau}) - f_1(y^\star),$$

which is a measure of function-value suboptimality.

```
class LinearConvergence:
    @staticmethod
    def get_parameters_function_value_suboptimality(
            algo: Type[Algorithm],
            h: int = 0,
            alpha: int = 0,
            j: int = 1,
            tau: int = 0
        ) -> Tuple[np.ndarray, np.ndarray, np.ndarray, np.ndarray]:
        # Returns (P,p,T,t) in (6.44).
        # ... (omitted for brevity) ...
```

6.3.2 Sublinear convergence

The class `SublinearConvergence`, accessible through `IterationIndependent` as a class attribute, contains helper functions for creating parameters (P, p, T, t) in Definition 4.1 in Section 4.1. In particular, in the case of sublinear convergence, recall that the lower bound $(\mathcal{R}(T, t, k))_{k \in \mathbb{N}_0}$ is the relevant quality, and we can simply set the other lower bound

$(\mathcal{V}(P,p,k))_{k\in\mathbb{N}_0}$ to zero by choosing $P=0$ and $p=0$. Setting this second lower bound to zero provides the greatest flexibility when analyzing sublinear convergence.

We provide examples of these helper functions below, and use the notation given in (6.2) to (6.9).

An optimality measure. The method below returns

$$\begin{bmatrix} P = 0, \\ p = 0, \\ T = \begin{cases} \left(P_{(1,1)}U_\tau^{0,h+\alpha+1}\right)^\top P_{(1,1)}U_\tau^{0,h+\alpha+1} & \text{if } m=1, \\ \left(\sum_{i=1}^m P_{(i,1)}U_\tau^{0,h+\alpha+1}\right)^\top \sum_{i=1}^m P_{(i,1)}U_\tau^{0,h+\alpha+1} \\ \quad + \sum_{i=2}^m \left((P_{(1,1)}-P_{(i,1)})Y_\tau^{0,h+\alpha+1}\right)^\top \left((P_{(1,1)}-P_{(i,1)})Y_\tau^{0,h+\alpha+1}\right) & \text{if } m>1, \end{cases} \\ t = 0, \end{bmatrix} \tag{6.45}$$

where $\tau \in [\![0, h+\alpha+1]\!]$. This gives

$$\mathcal{R}(T,t,k) = \begin{cases} \|u_{1,1}^{k+\tau}\|^2 & \text{if } m=1, \\ \|\sum_{i=1}^m u_{i,1}^{k+\tau}\|^2 + \sum_{i=2}^m \|y_{1,1}^{k+\tau} - y_{i,1}^{k+\tau}\|^2 & \text{if } m>1, \end{cases}$$

which is an optimality measure for the inclusion problem (3.1).

```
class SublinearConvergence:
    @staticmethod
    def get_parameters_optimality_measure(
            algo: Type[Algorithm],
            h: int = 0,
            alpha: int = 0,
            tau: int = 0
        ) -> Union[
            Tuple[np.ndarray, np.ndarray],
            Tuple[np.ndarray, np.ndarray, np.ndarray, np.ndarray]
            ]:
        # Returns (P,T) in (6.45) when I_func = ∅, otherwise (P,p,T,t).
        # ... (omitted for brevity) ...
```

Fixed-point residual. The method below returns

$$\begin{bmatrix} P = 0, \\ p = 0, \\ T = \left(X_{\tau+1}^{0,h+\alpha+1} - X_\tau^{0,h+\alpha+1}\right)^\top \left(X_{\tau+1}^{0,h+\alpha+1} - X_\tau^{0,h+\alpha+1}\right), \\ t = 0, \end{bmatrix} \tag{6.46}$$

where $\tau \in [\![0, h+\alpha+1]\!]$. This gives

$$\mathcal{R}(T,t,k) = \|\boldsymbol{x}^{k+\tau+1} - \boldsymbol{x}^{k+\tau}\|^2,$$

i.e., the squared norm of the fixed-point residual of (3.2).

```
class SublinearConvergence:
    @staticmethod
    def get_parameters_fixed_point_residual(
            algo: Type[Algorithm],
            h: int = 0,
            alpha: int = 0,
            tau: int = 0
        ) -> Union[
            Tuple[np.ndarray, np.ndarray],
            Tuple[np.ndarray, np.ndarray, np.ndarray, np.ndarray]
            ]:
        # Returns (P,T) in (6.46) when I_func = ∅, otherwise (P,p,T,t).
        # ... (omitted for brevity) ...
```

Function-value suboptimality. The method below requires $m = m_{\text{func}} = 1$ and returns

$$\begin{cases} P = 0, \\ p = 0, \\ T = 0, \\ t = \left(F^{0,h+\alpha+1}_{(1,j,\tau)} - F^{0,h+\alpha+1}_{(1,\star,\star)}\right)^\top, \end{cases} \tag{6.47}$$

where $\tau \in [\![0, h+\alpha+1]\!]$ and $j \in [\![1, \bar{m}_1]\!]$. This gives

$$\mathcal{R}(T,t,k) = f_1(y^{k+\tau}_{1,j}) - f_1(y^\star),$$

which is a measure of function-value suboptimality.

```
class SublinearConvergence:
    @staticmethod
    def get_parameters_function_value_suboptimality(
            algo: Type[Algorithm],
            h: int = 0,
            alpha: int = 0,
            j: int = 1,
            tau: int = 0
        ) -> Tuple[np.ndarray, np.ndarray, np.ndarray, np.ndarray]:
        # Returns (P,p,T,t) in (6.47).
        # ... (omitted for brevity) ...
```

6.4 Automated iteration-dependent analysis

This section provides a necessary and sufficient condition, in terms of the feasibility of a semidefinite program, for the existence of chained quadratic Lyapunov inequalities in the sense of Definition 4.2 in Section 4.2.

Theorem 6.10

Suppose that Assumptions 3.1 and 6.7 hold with $K \in \mathbb{N}$, $Q_0, Q_K \in \mathbb{S}^{n+\bar{m}+m}$, and $q_0, q_k \in \mathbb{R}^{\bar{m}_{\text{func}}+m_{\text{func}}}$. Then a sufficient condition for there to exists $((Q_k, q_k))_{k=1}^{K-1}$ and c such that $((Q_k, q_k))_{k=0}^{K}$ and c satisfies a length K sequence of chained Lyapunov inequalities for algorithm (3.2) over the problem class defined by $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ and $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ (see Definition 4.2) is that the following system of constraints

for each $k \in [\![0, K-1]\!]$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-ineq}} \\ \forall((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1} \end{pmatrix} \quad \lambda^{\text{func-ineq},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \tag{6.48a}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-eq}} \\ \forall((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1} \end{pmatrix} \quad \nu^{\text{func-eq},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \in \mathbb{R}, \tag{6.48b}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{op}} \\ \forall((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1} \end{pmatrix} \quad \lambda^{\text{op},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \geq 0, \tag{6.48c}$$

$$\begin{aligned} &-W_k + \sum_{\substack{i \in \mathcal{I}_{\text{func}} \\ o \in \mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1}}} \lambda^{\text{func-ineq},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{k,\,k+1,\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &\quad + \sum_{\substack{i \in \mathcal{I}_{\text{func}} \\ o \in \mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1}}} \nu^{\text{func-eq},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{k,\,k+1,\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &\quad + \sum_{\substack{i \in \mathcal{I}_{\text{op}} \\ o \in \mathcal{O}_i^{\text{op}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1}}} \lambda^{\text{op},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} W^{k,\,k+1,\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \succeq 0, \end{aligned} \tag{6.48d}$$

$$\begin{aligned} &-w_k + \sum_{\substack{i \in \mathcal{I}_{\text{func}} \\ o \in \mathcal{O}_i^{\text{func-ineq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1}}} \lambda^{\text{func-ineq},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{k,\,k+1,\,\text{func-ineq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &\quad + \sum_{\substack{i \in \mathcal{I}_{\text{func}} \\ o \in \mathcal{O}_i^{\text{func-eq}} \\ ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1}}} \nu^{\text{func-eq},\,k}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} F^{k,\,k+1,\,\text{func-eq}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} = 0, \end{aligned} \tag{6.48e}$$

end

$$(\forall k \in [\![1, K-1]\!]) \quad Q_k \in \mathbb{S}^{n+\bar{m}+m}, \tag{6.48f}$$

$$(\forall k \in [\![1, K-1]\!]) \quad q_k \in \mathbb{R}^{\bar{m}_{\text{func}}+m_{\text{func}}}, \tag{6.48g}$$

$$c \in \mathbb{R}_+, \tag{6.48h}$$

is feasible for the scalars

$$\lambda^{\text{func-ineq},\,k}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)},$$
$$\nu^{\text{func-eq},\,k}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)},$$
$$\lambda^{\text{op},\,k}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)},$$

the matrices and vectors $((Q_k, q_k))_{k=1}^{K-1}$*, and the scalar* c*, where we have introduced*

$$W_0 = (\Theta_1^{(0)})^\top Q_1 \Theta_1^{(0)} - c\Theta_0^\top Q_0 \Theta_0, \tag{6.49}$$

$$w_0 = \theta_1^\top q_1 - c\theta_0^\top q_0, \tag{6.50}$$

$$(\forall k \in [\![1, K-1]\!]) \quad W_k = (\Theta_1^{(k)})^\top Q_{k+1} \Theta_1^{(k)} - \Theta_0^\top Q_k \Theta_0, \tag{6.51}$$

$$(\forall k \in [\![1, K-1]\!]) \quad w_k = \theta_1^\top q_{k+1} - \theta_0^\top q_k, \tag{6.52}$$

and

$$\Theta_0 = \underbrace{\begin{bmatrix} I_{n+\bar{m}} & 0_{(n+\bar{m})\times\bar{m}} & 0_{(n+\bar{m})\times m} \\ 0_{m\times(n+\bar{m})} & 0_{m\times\bar{m}} & I_m \end{bmatrix}}_{\in\mathbb{R}^{(n+\bar{m}+m)\times(n+2\bar{m}+m)}}, \tag{6.53}$$

$$\theta_0 = \underbrace{\begin{bmatrix} I_{\bar{m}_{\text{func}}} & 0_{\bar{m}_{\text{func}}\times\bar{m}_{\text{func}}} & 0_{\bar{m}_{\text{func}}\times m_{\text{func}}} \\ 0_{m_{\text{func}}\times\bar{m}_{\text{func}}} & 0_{m_{\text{func}}\times\bar{m}_{\text{func}}} & I_{m_{\text{func}}} \end{bmatrix}}_{\in\mathbb{R}^{(\bar{m}_{\text{func}}+m_{\text{func}})\times(2\bar{m}_{\text{func}}+m_{\text{func}})}}, \tag{6.54}$$

$$(\forall k \in [\![0, K-1]\!]) \quad \Theta_1^{(k)} = \underbrace{\left[\begin{array}{c|c} \multicolumn{2}{c}{X_{k+1}^{k,k+1}} \\ \hline 0_{(\bar{m}+m)\times(n+\bar{m})} & I_{\bar{m}+m} \end{array}\right]}_{\in\mathbb{R}^{(n+\bar{m}+m)\times(n+2\bar{m}+m)}}, \tag{6.55}$$

$$\theta_1 = \underbrace{\begin{bmatrix} 0_{(\bar{m}_{\text{func}}+m_{\text{func}})\times\bar{m}_{\text{func}}} & I_{\bar{m}_{\text{func}}+m_{\text{func}}} \end{bmatrix}}_{\in\mathbb{R}^{(\bar{m}_{\text{func}}+m_{\text{func}})\times(2\bar{m}_{\text{func}}+m_{\text{func}})}}. \tag{6.56}$$

Furthermore, if the interpolation conditions for $(\mathcal{F}_i)_{i\in\mathcal{I}_{\text{func}}}$ *and* $(\mathcal{G}_i)_{i\in\mathcal{I}_{\text{op}}}$ *are tight,* $\dim\mathcal{H} \geq n + 2\bar{m} + m$*, and there exists*

$$(\forall k \in [\![0, K-1]\!]) \quad \begin{bmatrix} G_k \in \mathbb{S}_{++}^{n+2\bar{m}+m}, \\ \chi_k \in \mathbb{R}^{2\bar{m}_{\text{func}}+m_{\text{func}}}, \end{bmatrix}$$

such that

for each $k \in [\![0, K-1]\!]$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-ineq}} \\ \forall ((j_1,k_1),\dots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1} \end{pmatrix} \quad \begin{aligned} &\chi_k^\top F^{k,\,k+1,\,\text{func-ineq}}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &\quad + \operatorname{trace}(W^{k,\,k+1,\,\text{func-ineq}}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)} G_k) \leq 0, \end{aligned}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{func-eq}} \\ \forall ((j_1,k_1),\dots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1} \end{pmatrix} \quad \begin{aligned} &\chi_k^\top F^{k,\,k+1,\,\text{func-eq}}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)} \\ &\quad + \operatorname{trace}(W^{k,\,k+1,\,\text{func-eq}}_{(i,j_1,k_1,\dots,j_{n_{i,o}},k_{n_{i,o}},o)} G_k) = 0, \end{aligned}$$

$$\begin{pmatrix} \forall i \in \mathcal{I}_{\text{func}} \\ \forall o \in \mathcal{O}_i^{\text{op}} \\ \forall ((j_1,k_1),\ldots,(j_{n_{i,o}},k_{n_{i,o}})) \in \mathcal{J}_{i,o}^{k,k+1} \end{pmatrix} \quad \operatorname{trace}(W^{k,\,k+1,\,\text{op}}_{(i,j_1,k_1,\ldots,j_{n_{i,o}},k_{n_{i,o}},o)} G_k) \leq 0,$$

end

then (6.48) *is also a necessary condition.*

Proof. Note that

$$\begin{aligned}
(\boldsymbol{x}^k, \boldsymbol{u}^k, \hat{\boldsymbol{u}}^\star, y^\star) &= (\Theta_0 \otimes \mathrm{Id})(\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{u}^{k+1}, \hat{\boldsymbol{u}}^\star, y^\star), \\
(\boldsymbol{F}^k, \boldsymbol{F}^\star) &= \theta_0(\boldsymbol{F}^k, \boldsymbol{F}^{k+1}, \boldsymbol{F}^\star), \\
(\boldsymbol{x}^{k+1}, \boldsymbol{u}^{k+1}, \hat{\boldsymbol{u}}^\star, y^\star) &= (\Theta_1^{(k)} \otimes \mathrm{Id})(\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{u}^{k+1}, \hat{\boldsymbol{u}}^\star, y^\star), \\
(\boldsymbol{F}^{k+1}, \boldsymbol{F}^\star) &= \theta_1(\boldsymbol{F}^k, \boldsymbol{F}^{k+1}, \boldsymbol{F}^\star),
\end{aligned}$$

where we have used (6.53), (6.54), (6.55), and (6.56), respectively.

First, suppose that the parameters $((Q_k, q_k))_{k=1}^{K-1}$ and c are fixed and consider $(\mathcal{V}(k))_{k=0}^{K}$ in (4.5). Note that

$$\begin{aligned}
&\mathcal{V}(1) - c\mathcal{V}(0) \\
&= \mathcal{Q}((\Theta_1^{(0)})^\top Q_1 \Theta_1^{(0)} - c\Theta_0^\top Q_0 \Theta_0, (\boldsymbol{x}^0, \boldsymbol{u}^0, \boldsymbol{u}^1, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (q_1^\top \theta_1 - c q_0^\top \theta_0)(\boldsymbol{F}^0, \boldsymbol{F}^1, \boldsymbol{F}^\star) \\
&= \mathcal{Q}(W_0, (\boldsymbol{x}^0, \boldsymbol{u}^0, \boldsymbol{u}^1, \hat{\boldsymbol{u}}^\star, y^\star)) + w_0^\top (\boldsymbol{F}^0, \boldsymbol{F}^1, \boldsymbol{F}^\star)
\end{aligned} \tag{6.57}$$

where (6.49) and (6.50) is used in the last equality. Therefore, using the (6.57) as the objective function in (PEP), Theorem 6.8 gives that (6.48), with $k = 0$, is a sufficient condition for the inequality $\mathcal{V}(1) \leq c\mathcal{V}(0)$.

For $k \in [\![1, K-1]\!]$, note that

$$\begin{aligned}
&\mathcal{V}(k+1) - \mathcal{V}(k) \\
&= \mathcal{Q}(Q_{k+1}, (\boldsymbol{x}^{k+1}, \boldsymbol{u}^{k+1}, \hat{\boldsymbol{u}}^\star, y^\star)) + q_{k+1}^\top (\boldsymbol{F}^{k+1}, \boldsymbol{F}^\star) \\
&\quad - \mathcal{Q}(Q_k, (\boldsymbol{x}^k, \boldsymbol{u}^k, \hat{\boldsymbol{u}}^\star, y^\star)) - q_k^\top (\boldsymbol{F}^k, \boldsymbol{F}^\star) \\
&= \mathcal{Q}((\Theta_1^{(k)})^\top Q_{k+1} \Theta_1^{(k)} - \Theta_0^\top Q_k \Theta_0, (\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{u}^{k+1}, \hat{\boldsymbol{u}}^\star, y^\star)) \\
&\quad + (q_{k+1}^\top \theta_1 - q_k^\top \theta_0)(\boldsymbol{F}^k, \boldsymbol{F}^{k+1}, \boldsymbol{F}^\star) \\
&= \mathcal{Q}(W_k, (\boldsymbol{x}^k, \boldsymbol{u}^k, \boldsymbol{u}^{k+1}, \hat{\boldsymbol{u}}^\star, y^\star)) + w_k^\top (\boldsymbol{F}^k, \boldsymbol{F}^{k+1}, \boldsymbol{F}^\star)
\end{aligned} \tag{6.58}$$

where (6.51) and (6.52) is used in the last equality. Therefore, using the (6.58) as the objective function in (PEP), Theorem 6.8 gives that (6.48), constrained to this particular k, is a sufficient condition for the inequality $\mathcal{V}(k+1) \leq \mathcal{V}(k)$.

Second, note that the proof is complete if we let the parameters $((Q_k, q_k))_{k=1}^{K-1}$ and c free, as in (6.48f), (6.48g), and (6.48h). □

6.4.1 Selecting initial and final Lyapunov functions

The class `IterationDependent` contains helper functions for creating matrices $((Q_k, q_k))_{k=0}^{K}$ that parameterize $(\mathcal{V}(k))_{k=0}^{K}$ in Definition 4.2 in Section 4.2. We provide examples of these helper functions below, and use the notation given in (6.2) to (6.9).

Function-value suboptimality. The method below requires $m = m_{\text{func}} = 1$ and returns

$$\begin{bmatrix} Q_k = 0, \\ q_k = \left(F^{k,k}_{(1,j,k)} - F^{k,k}_{(1,\star,\star)}\right)^\top, \end{bmatrix} \tag{6.59}$$

where $k \in [\![0, K]\!]$ and $j \in [\![1, \bar{m}_i]\!]$. This gives

$$\mathcal{V}(k) = f_1(y^k_{1,j}) - f_1(y^\star),$$

which is a measure of function-value suboptimality.

```
class IterationDependent:
    @staticmethod
    def get_parameters_function_value_suboptimality(
            algo: Type[Algorithm],
            k: int,
            j: int = 1
        ) -> Tuple[np.ndarray, np.ndarray]:
        # Returns (Q_k, q_k) in (6.59).
        # ... (omitted for brevity) ...
```

Distance to solution. The method below returns

$$\begin{bmatrix} Q_k = \left(P_{(i,j)}Y^{k,k}_k - P_{(i,\star)}Y^{k,k}_\star\right)^\top \left(P_{(i,j)}Y^{k,k}_k - P_{(i,\star)}Y^{k,k}_\star\right), \\ q_k = 0, \end{bmatrix} \tag{6.60}$$

where $k \in [\![0, K]\!]$, $j \in [\![1, \bar{m}_i]\!]$, and $i \in \mathcal{I}_{\text{func}} \cup \mathcal{I}_{\text{op}}$. This gives

$$\mathcal{V}(k) = \|y^k_{i,j} - y^\star\|^2,$$

which measures the squared distance to the solution.

```
class IterationDependent:
    @staticmethod
    def get_parameters_distance_to_solution(
            algo: Type[Algorithm],
            k: int,
            i: int = 1,
            j: int = 1
        ) -> Union[np.ndarray, Tuple[np.ndarray, np.ndarray]]:
        # Returns Q_k in (6.60) when I_func = ∅, otherwise (Q_k, q_k).
        # ... (omitted for brevity) ...
```

Fixed-point residual. The method below returns

$$\begin{bmatrix} Q_k = \left(X^{k,k}_{k+1} - X^{k,k}_k\right)^\top \left(X^{k,k}_{k+1} - X^{k,k}_k\right), \\ q_k = 0, \end{bmatrix} \tag{6.61}$$

where $k \in [\![0, K]\!]$. This gives

$$\mathcal{V}(k) = \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2,$$

i.e., the squared norm of the fixed-point residual of (3.2).

```python
class IterationDependent:
    @staticmethod
    def get_parameters_fixed_point_residual(
            algo: Type[Algorithm],
            k: int
        ) -> Union[np.ndarray, Tuple[np.ndarray, np.ndarray]]:
        # Returns Q_k in (6.61) when I_func = ∅, otherwise (Q_k, q_k).
        # ... (omitted for brevity) ...
```

An optimality measure. The method below returns

$$
\left[\begin{aligned}
&Q_k = \begin{cases} \left(P_{(1,1)}U_k^{k,k}\right)^\top P_{(1,1)}U_k^{k,k} & \text{if } m = 1, \\ \left(\sum_{i=1}^m P_{(i,1)}U_k^{k,k}\right)^\top \sum_{i=1}^m P_{(i,1)}U_k^{k,k} & \\ \quad + \sum_{i=2}^m \left((P_{(1,1)} - P_{(i,1)})Y_k^{k,k}\right)^\top \left((P_{(1,1)} - P_{(i,1)})Y_k^{k,k}\right) & \text{if } m > 1, \end{cases} \\
&q_k = 0,
\end{aligned}\right. \tag{6.62}
$$

where $k \in [\![0, K]\!]$. This gives

$$
\mathcal{V}(k) = \begin{cases} \|u_{1,1}^k\|^2 & \text{if } m = 1, \\ \|\sum_{i=1}^m u_{i,1}^k\|^2 + \sum_{i=2}^m \|y_{1,1}^k - y_{i,1}^k\|^2 & \text{if } m > 1, \end{cases}
$$

which is an optimality measure for the inclusion problem (3.1).

```python
class IterationDependent:
    @staticmethod
    def get_parameters_optimality_measure(
            algo: Type[Algorithm],
            k: int
        ) -> Union[np.ndarray, Tuple[np.ndarray, np.ndarray]]:
        # Returns Q_k in (6.62) when I_func = ∅, otherwise (Q_k, q_k).
        # ... (omitted for brevity) ...
```

7. Conclusion

This paper introduced AutoLyap, a Python package designed to automate Lyapunov analyses for first-order methods via semidefinite programming. It simplifies the derivation and verification of convergence guarantees for structured optimization and inclusion problems.

AutoLyap is structured to facilitate community involvement, allowing researchers to extend the framework and collaboratively improve its capabilities. We mention two relevant directions for future development.

(i) The allowable classes of interpolation conditions in Assumption 6.7 are quite general and allow for integrating new and improved interpolation conditions developed over time (see, e.g., [27, 33] for recent efforts in this direction).

(ii) The algorithm representation (3.2) can be extended to allow for more types of oracles (including, e.g., Frank–Wolfe-type oracles [39], Bregman-type oracles [12], or approximate oracles with explicit error bounds [4]).

Appendix A: Interpolation conditions

This section contains some examples of interpolation conditions in the sense of Assumption 6.7 in Section 6.2 that are used in AutoLyap. Below, we give the generic versions of the interpolation conditions, and the precise translation to fit Assumption 6.7 is done in the `InclusionProblem` class and the function and operator classes found in the `autolyap.problemclass` module of AutoLyap, as discussed in Section 3.1.

A.1 Smooth and (strongly/weakly) convex functions

Proposition A.1 ([40, Theorem 4], [32, Theorem 3.1])
Let $-\infty < \mu < L \leq +\infty$, $L > 0$, and $\{(y_i, F_i, u_i)\}_{i \in \mathcal{J}}$ be a finite family of triplets in $\mathcal{H} \times \mathbb{R} \times \mathcal{H}$ indexed by $\mathcal{J}$. Then the following are equivalent:

(i) There exists $f \in \mathcal{F}_{\mu,L}(\mathcal{H})$ such that

$$(\forall i \in \mathcal{J}) \quad \begin{bmatrix} f(y_i) = F_i, \\ u_i \in \partial f(y_i). \end{bmatrix}$$

(ii) It holds that

$$\begin{aligned}(\forall i, j \in \mathcal{J}) \quad F_i \geq F_j + \langle u_j, y_i - y_j \rangle &+ \frac{\mu}{2}\|y_i - y_j\|^2 \\ &+ \frac{1}{2(L-\mu)}\|u_i - u_j - \mu(y_i - y_j)\|^2,\end{aligned}$$

where $\frac{1}{2(L-\mu)}$ is interpreted as 0 in the case $L = +\infty$.

Note that condition (ii) in Proposition A.1 can be written as

$$(\forall i, j \in \mathcal{J}) \quad a^\top \begin{bmatrix} F_i \\ F_j \end{bmatrix} + \mathcal{Q}(M, (y_i, y_j, u_i, u_j)) \leq 0,$$

where $a^\top = \begin{bmatrix} -1 & 1 \end{bmatrix}$ and

$$M = \begin{cases} \dfrac{1}{2(L-\mu)} \begin{bmatrix} \mu L & -\mu L & -\mu & L \\ -\mu L & \mu L & \mu & -L \\ -\mu & \mu & 1 & -1 \\ L & -L & -1 & 1 \end{bmatrix} & \text{if } L < \infty, \\ \dfrac{1}{2} \begin{bmatrix} \mu & -\mu & 0 & 1 \\ -\mu & \mu & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} & \text{if } L = +\infty. \end{cases}$$

In particular, the function class $\mathcal{F}_{\mu,L}(\mathcal{H})$ has a tight interpolation condition that fits Assumption 6.7.

A.2 Gradient-dominated functions (Łojasiewicz)

Let $\mu_{\text{gd}} \in \mathbb{R}_{++}$ and $\{(y_i, F_i, u_i)\}_{i \in \mathcal{J}}$ be a finite family of triplets in $\mathcal{H} \times \mathbb{R} \times \mathcal{H}$ indexed by $\mathcal{J}$ such that $\star \in \mathcal{J}$ and $u_\star = 0$.

Suppose that there exists a μ_{gd}-gradient dominated $f : \mathcal{H} \to \mathbb{R}$ such that

$$(\forall i \in \mathcal{J}) \quad \begin{bmatrix} f(y_i) = F_i, \\ u_i \in \partial f(y_i). \end{bmatrix}$$

Then it is clear that

$$(\forall i \in \mathcal{J}) \quad 0 \leq F_i - F_\star \leq \frac{1}{2\mu_{gd}} \|u_i\|^2. \tag{A.1}$$

Note that (A.1) can be written as

$$\begin{pmatrix} \forall i \in \mathcal{J} \\ \forall o \in \{1, 2\} \end{pmatrix} \quad a_{(o)}^\top \begin{bmatrix} F_i \\ F_\star \end{bmatrix} + \mathcal{Q}(M_{(o)}, (y_i, y_\star, u_i, u_\star)) \leq 0,$$

where

$$a_{(1)} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad \text{and} \quad M_{(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

and

$$a_{(2)} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad \text{and} \quad M_{(2)} = \frac{1}{2\mu_{gd}} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

In particular, this function class has an interpolation condition that fits Assumption 6.7. This is the version presently implemented in `GradientDominated`; it may be extended or revised in future updates.

A.3 Maximally monotone operators

Proposition A.2 ([5, Theorem 20.21])
Let $\{(y_i, u_i)\}_{i \in \mathcal{J}}$ be a family of couples in $\mathcal{H} \times \mathcal{H}$ indexed by $\mathcal{J}$. Then the following are equivalent:

(i) There exists a maximally monotone operator $G : \mathcal{H} \rightrightarrows \mathcal{H}$ such that

$$(\forall i \in \mathcal{J}) \quad u_i \in G(y_i).$$

(ii) It holds that

$$(\forall i, j \in \mathcal{J}) \quad \langle u_i - u_j, y_i - y_j \rangle \geq 0.$$

Note that condition (ii) in Proposition A.2 can be written as

$$(\forall i, j \in \mathcal{J}) \quad \mathcal{Q}(M, (y_i, y_j, u_i, u_j)) \leq 0$$

where

$$M = \frac{1}{2}\begin{bmatrix} 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \\ -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix}.$$

In particular, this operator class has a tight interpolation condition that fits Assumption 6.7.

A.4 Strongly monotone operators

Proposition A.3 ([34, Proposition 1])
Let $\mu \in \mathbb{R}_{++}$ and $\{(y_i, u_i)\}_{i\in\mathcal{J}}$ be a family of couples in $\mathcal{H} \times \mathcal{H}$ indexed by $\mathcal{J}$. Then the following are equivalent:

(i) There exists a μ-strongly and maximally monotone operator $G : \mathcal{H} \rightrightarrows \mathcal{H}$ such that

$$(\forall i \in \mathcal{J}) \quad u_i \in G(y_i).$$

(ii) It holds that

$$(\forall i, j \in \mathcal{J}) \quad \langle u_i - u_j, y_i - y_j \rangle \geq \mu \|y_i - y_j\|^2.$$

Note that condition (ii) in Proposition A.3 can be written as

$$(\forall i, j \in \mathcal{J}) \quad \mathcal{Q}(M, (y_i, y_j, u_i, u_j)) \leq 0,$$

where

$$M = \frac{1}{2}\begin{bmatrix} 2\mu & -2\mu & -1 & 1 \\ -2\mu & 2\mu & 1 & -1 \\ -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix}.$$

In particular, this operator class has a tight interpolation condition that fits Assumption 6.7.

A.5 Lipschitz continuous operators

Proposition A.4 (Kirszbraun–Valentine Theorem [22, 43, 44])
Let $L \in \mathbb{R}_{++}$ and $\{(y_i, u_i)\}_{i\in\mathcal{J}}$ be a family of couples in $\mathcal{H} \times \mathcal{H}$ indexed by $\mathcal{J}$. Then the following are equivalent:

(i) There exists a L-Lipschitz continuous operator $G : \mathcal{H} \to \mathcal{H}$ such that

$$(\forall i \in \mathcal{J}) \quad u_i \in G(y_i).$$

(ii) It holds that

$$(\forall i, j \in \mathcal{J}) \quad \|u_i - u_j\| \leq L\|y_i - y_j\|.$$

Note that condition (ii) in Proposition A.4 can be written as

$$(\forall i, j \in \mathcal{J}) \quad \mathcal{Q}(M, (y_i, y_j, u_i, u_j)) \leq 0,$$

where

$$M = \begin{bmatrix} -L^2 & L^2 & 0 & 0 \\ L^2 & -L^2 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}.$$

In particular, this operator class has a tight interpolation condition that fits Assumption 6.7.

A.6 Cocoercive operators

Proposition A.5 ([34, Proposition 2])
Let $\beta \in \mathbb{R}_{++}$ and $\{(y_i, u_i)\}_{i \in \mathcal{J}}$ be a family of couples in $\mathcal{H} \times \mathcal{H}$ indexed by $\mathcal{J}$. Then the following are equivalent:

(i) There exists a β-cocoercive operator $G : \mathcal{H} \to \mathcal{H}$ such that

$$(\forall i \in \mathcal{J}) \quad u_i \in G(y_i).$$

(ii) It holds that

$$(\forall i, j \in \mathcal{J}) \quad \langle u_i - u_j, y_i - y_j \rangle \geq \beta \|u_i - u_j\|^2.$$

Note that condition (ii) in Proposition A.5 can be written as

$$(\forall i, j \in \mathcal{J}) \quad \mathcal{Q}(M, (y_i, y_j, u_i, u_j)) \leq 0,$$

where

$$M = \frac{1}{2} \begin{bmatrix} 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \\ -1 & 1 & 2\beta & -2\beta \\ 1 & -1 & -2\beta & 2\beta \end{bmatrix}.$$

In particular, this operator class has a tight interpolation condition that fits Assumption 6.7.

Acknowledgments

M. Upadhyaya and P. Giselsson acknowledge support from the ELLIIT Strategic Research Area and the Wallenberg AI, Autonomous Systems, and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation. Additionally, P. Giselsson acknowledges support from the Swedish Research Council. A. B. Taylor is supported by the European Union (ERC grant CASPER 101162889). The French government also partly funded this work under the management of Agence Nationale de la Recherche as part of the "France 2030" program, reference ANR-23-IACL-0008 "PR[AI]RIE-PSAI".

References

[1] H. Abbaszadehpeivasti, E. de Klerk, and M. Zamani. "Conditions for linear convergence of the gradient method for non-convex optimization". *Optimization Letters* **17**:5 (2023), pp. 1105–1125. DOI: 10.1007/s11590-023-01981-2.

[2] M. ApS. *The MOSEK Python fusion API manual. Version 11.0.* 2025. URL: https://docs.mosek.com/latest/pythonfusion/index.html.

[3] N. Bansal and A. Gupta. "Potential-function proofs for gradient methods". *Theory of Computing* **15**:1 (2019), pp. 1–32. DOI: 10.4086/toc.2019.v015a004.

[4] M. Barré, A. B. Taylor, and F. Bach. "Principled analyses and design of first-order methods with inexact proximal operators". *Mathematical Programming* **201**:1–2 (2022), pp. 185–230. DOI: 10.1007/s10107-022-01903-7.

[5] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer International Publishing, 2017. ISBN: 9783319483115. DOI: 10.1007/978-3-319-48311-5.

[6] L. M. Briceño-Arias and P. L. Combettes. "Monotone operator methods for Nash equilibria in non-potential games". In: *Computational and Analytical Mathematics*. Springer New York, 2013, pp. 143–159. ISBN: 9781461476214. DOI: 10.1007/978-1-4614-7621-4_9.

[7] M. N. Bùi and P. L. Combettes. "Analysis and numerical solution of a modular convex Nash equilibrium problem" (2021). arXiv: 2111.01781v1 [math.OC].

[8] A. Chambolle and T. Pock. "A first-order primal-dual algorithm for convex problems with applications to imaging". *Journal of Mathematical Imaging and Vision* **40**:1 (2011), pp. 120–145. DOI: 10.1007/s10851-010-0251-1.

[9] P. L. Combettes and S. A. Hirstoaga. "Equilibrium programming in Hilbert spaces". *Journal of Nonlinear and Convex Analysis* **6** (2005), pp. 117–136.

[10] A. d'Aspremont, D. Scieur, and A. Taylor. "Acceleration methods". *Foundations and Trends in Optimization* **5**:1–2 (2021), pp. 1–245. DOI: 10.1561/2400000036.

[11] J. Douglas and H. H. Rachford. "On the numerical solution of heat conduction problems in two and three space variables". *Transactions of the American Mathematical Society* **82**:2 (1956), pp. 421–439. DOI: 10.1090/s0002-9947-1956-0084194-4.

[12] R.-A. Dragomir, A. B. Taylor, A. d'Aspremont, and J. Bolte. "Optimal complexity and certification of Bregman first-order methods". *Mathematical Programming* **194**:1–2 (2021), pp. 41–83. DOI: 10.1007/s10107-021-01618-1.

[13] Y. Drori. "The exact information-based complexity of smooth convex minimization". *Journal of Complexity* **39** (2017), pp. 1–16. DOI: 10.1016/j.jco.2016.11.001.

[14] Y. Drori and M. Teboulle. "Performance of first-order methods for smooth convex minimization: A novel approach". *Mathematical Programming* **145**:1–2 (2014), pp. 451–482. DOI: 10.1007/s10107-013-0653-0.

[15] J. Eckstein and D. P. Bertsekas. "On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators". *Mathematical Programming* **55**:1–3 (1992), pp. 293–318. DOI: 10.1007/bf01581204.

[16] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Vol. I. Springer New York, 2004. ISBN: 9780387955803. DOI: 10.1007/b97543.

[17] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Vol. II. Springer New York, 2004. ISBN: 9780387955810. DOI: 10.1007/b97544.

[18] M. Fazlyab, A. Ribeiro, M. Morari, and V. M. Preciado. "Analysis of optimization algorithms via integral quadratic constraints: Nonstrongly convex problems". *SIAM Journal on Optimization* **28**:3 (2018), pp. 2654–2689. DOI: 10.1137/17m1136845.

[19] P. Giselsson. "Tight global linear convergence rate bounds for Douglas–Rachford splitting". *Journal of Fixed Point Theory and Applications* **19**:4 (2017), pp. 2241–2270. DOI: 10.1007/s11784-017-0417-1.

[20] B. Goujaud, C. Moucer, F. Glineur, J. M. Hendrickx, A. B. Taylor, and A. Dieuleveut. "PEPit: Computer-assisted worst-case analyses of first-order optimization methods in Python". *Mathematical Programming Computation* **16**:3 (2024), pp. 337–367. DOI: 10.1007/s12532-024-00259-7.

[21] D. Kim and J. A. Fessler. "Optimized first-order methods for smooth convex minimization". *Mathematical Programming* **159**:1–2 (2016), pp. 81–107. DOI: 10.1007/s10107-015-0949-3.

[22] M. Kirszbraun. "Über die zusammenziehende und Lipschitzsche transformationen". German. *Fundamenta Mathematicae* **22**:1 (1934), pp. 77–108.

[23] G. M. Korpelevich. "The extragradient method for finding saddle points and other problems". *Matecon* **12** (1976), pp. 747–756.

[24] L. Lessard. "The analysis of optimization algorithms: A dissipativity approach". *IEEE Control Systems* **42**:3 (2022), pp. 58–72. DOI: 10.1109/mcs.2022.3157115.

[25] L. Lessard, B. Recht, and A. Packard. "Analysis and design of optimization algorithms via integral quadratic constraints". *SIAM Journal on Optimization* **26**:1 (2016), pp. 57–95. DOI: 10.1137/15m1009597.

[26] P. L. Lions and B. Mercier. "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6 (1979), pp. 964–979. DOI: 10.1137/0716071.

[27] A. Luner and B. Grimmer. "Performance estimation for smooth and strongly convex sets" (2024). arXiv: 2410.14811v2 [math.OC].

[28] A. Megretski and A. Rantzer. "System analysis via integral quadratic constraints". *IEEE Transactions on Automatic Control* **42**:6 (1997), pp. 819–830. DOI: 10.1109/9.587335.

[29] B. S. Mordukhovich. *Variational Analysis and Generalized Differentiation I*. Springer Berlin Heidelberg, 2006. ISBN: 9783540312475. DOI: 10.1007/3-540-31247-1.

[30] C. Moucer, A. Taylor, and F. Bach. "A systematic approach to Lyapunov analyses of continuous-time models in convex optimization". *SIAM Journal on Optimization* **33**:3 (2023), pp. 1558–1586. DOI: 10.1137/22m1498486.

[31] Y. Nesterov. "A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$". *Dokl. Akad. Nauk SSSR* **269**:3 (1983), pp. 543–547.

[32] T. Rotaru, F. Glineur, and P. Patrinos. "Tight convergence rates of the gradient method on smooth hypoconvex functions" (2022). arXiv: 2203.00775v2 [math.OC].

[33] A. Rubbens, J. M. Hendrickx, and A. Taylor. "A constructive approach to strengthen algebraic descriptions of function and operator classes" (2025). arXiv: 2504.14377v3 [math.OC].

[34] E. K. Ryu, A. B. Taylor, C. Bergeling, and P. Giselsson. "Operator splitting performance estimation: Tight contraction factors and optimal parameter selection". *SIAM Journal on Optimization* **30**:3 (2020), pp. 2251–2271. DOI: 10.1137/19m1304854.

[35] E. K. Ryu and W. Yin. *Large-Scale Convex Optimization. Algorithms & Analyses via Monotone Operators*. Cambridge University Press, Cambridge, United Kingdom, 2022. ISBN: 9781009160865.

[36] C. W. Scherer, C. Ebenbauer, and T. Holicki. "Optimization algorithm synthesis based on integral quadratic constraints: A tutorial". In: *Conference on Decision and Control (CDC)*. 2023. DOI: 10.1109/cdc49753.2023.10384198.

[37] A. Taylor and F. Bach. "Stochastic first-order methods: Non-asymptotic and computer-aided analyses via potential functions". In: *Conference on Learning Theory (COLT)*. 2019.

[38] A. Taylor, B. Van Scoy, and L. Lessard. "Lyapunov functions for first-order methods: Tight automated convergence guarantees". In: *International Conference on Machine Learning (ICML)*. 2018.

[39] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Exact worst-case performance of first-order methods for composite convex optimization". *SIAM Journal on Optimization* **27**:3 (2017), pp. 1283–1313. DOI: 10.1137/16m108104x.

[40] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Smooth strongly convex interpolation and exact worst-case performance of first-order methods". *Mathematical Programming* **161**:1–2 (2016), pp. 307–345. DOI: 10.1007/s10107-016-1009-3.

[41] A. B. Taylor, J. M. Hendrickx, and F. Glineur. "Performance estimation toolbox (PESTO): Automated worst-case analysis of first-order optimization methods". In: *Conference on Decision and Control (CDC)*. 2017. DOI: 10.1109/cdc.2017.8263832.

[42] M. Upadhyaya, S. Banert, A. B. Taylor, and P. Giselsson. "Automated tight Lyapunov analysis for first-order methods". *Mathematical Programming* **209** (2025), pp. 133–170. DOI: 10.1007/s10107-024-02061-8.

[43] F. A. Valentine. "A Lipschitz condition preserving extension for a vector function". *American Journal of Mathematics* **67**:1 (1945), pp. 83–93. DOI: 10.2307/2371917.

[44] F. A. Valentine. "On the extension of a vector function so as to preserve a Lipschitz condition". *Bulletin of the American Mathematical Society* **49**:2 (1943), pp. 100–108.

[45] B. Van Scoy and L. Lessard. "The speed-robustness trade-off for first-order methods with additive gradient noise" (2024). arXiv: 2109.05059v2 [math.OC].

Paper III

The Chambolle–Pock method converges weakly with $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$

Sebastian Banert Manu Upadhyaya Pontus Giselsson

Abstract

The Chambolle–Pock method is a versatile three-parameter algorithm designed to solve a broad class of composite convex optimization problems, which encompass two proper, lower semicontinuous, and convex functions, along with a linear operator L. The functions are accessed via their proximal operators, while the linear operator is evaluated in a forward manner. Among the three algorithm parameters τ, σ, and θ; $\tau, \sigma > 0$ serve as step sizes for the proximal operators, and θ is an extrapolation step parameter. Previous convergence results have been based on the assumption that $\theta = 1$. We demonstrate that weak convergence is achievable whenever $\theta > 1/2$ and $\tau\sigma\|L\|^2 < 4/(1+2\theta)$. Moreover, we establish tightness of the step size bound by providing an example that is nonconvergent whenever the second bound is violated.

Keywords. Chambolle–Pock, convex optimization, first-order methods

1. The Chambolle–Pock method

The Chambolle–Pock method [2], also known as the primal-dual hybrid gradient method, solves convex-concave saddle-point problems of the form

$$\underset{x\in\mathcal{H}}{\text{minimize}}\,\underset{y\in\mathcal{G}}{\text{maximize}}\;\; f(x) + \langle Lx, y\rangle - g^*(y), \tag{1.1}$$

where $f\colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ and $g\colon \mathcal{G} \to \mathbb{R} \cup \{+\infty\}$ are proper, convex, and lower semi-continuous functions, g^* is the convex conjugate of g, $L\colon \mathcal{H} \to \mathcal{G}$ is a nonzero bounded linear operator, and $\mathcal{H}$ and $\mathcal{G}$ are real Hilbert spaces. This is a primal-dual formulation of the primal composite optimization problem

$$\underset{x\in\mathcal{H}}{\text{minimize}}\;\; f(x) + g(Lx). \tag{1.2}$$

We assume that (1.1) has at least one solution $(x_\star, y_\star) \in \mathcal{H} \times \mathcal{G}$ that satisfies the Karush–Kuhn–Tucker (KKT) condition

$$-L^* y_\star \in \partial f(x_\star), \tag{1.3a}$$
$$Lx_\star \in \partial g^*(y_\star). \tag{1.3b}$$

The Chambolle–Pock method searches for such KKT points by iterating

$$x_{k+1} = \operatorname{prox}_{\tau f}(x_k - \tau L^* y_k), \tag{1.4a}$$
$$y_{k+1} = \operatorname{prox}_{\sigma g^*}(y_k + \sigma L(x_{k+1} + \theta(x_{k+1} - x_k))), \tag{1.4b}$$

where each function's proximal operators, the linear operator L and its adjoint L^* are evaluated once each in every iteration.

This algorithm is guaranteed to converge to a solution whenever $\theta = 1$ and the positive step size parameters τ and σ satisfy $\tau\sigma\|L\|^2 < 1$ as shown in the original work [2, Theorem 1], where $\|L\|$ is the operator norm. It was soon thereafter realized in [6] that for $\theta = 1$, the algorithm is an instance of the preconditioned proximal point method, applied to a specific maximally monotone inclusion problem, with a *strongly positive* and *symmetric* preconditioner. Strong positivity of the preconditioner is lost for $\theta = 1$ whenever $\tau\sigma\|L\|^2 \geq 1$, disabling analysis via the standard preconditioned proximal point method. In [3], sequence convergence with $\tau\sigma\|L\|^2 \leq 1$ is established using a modified proximal point analysis and in [8, 11], convergence to a solution is established for the wider range $\tau\sigma\|L\|^2 < 4/3$. All these results are in finite-dimensional settings and assume $\theta = 1$. A modification of the Chambolle–Pock method, which relies on symmetric preconditioning and permits larger values of $\tau\sigma\|L\|^2$, was presented in [5].

The symmetry of the proximal point preconditioner is lost whenever $\theta \neq 1$. Non-symmetric proximal point iterates are often augmented with different types of correction steps such as projection or momentum correction as in [4, 6, 7, 9] to form convergent algorithms. To the best of our knowledge, no convergence results exist for the pure Chambolle–Pock method, without additional correction, when $\theta \neq 1$.

The contribution of this paper lies in establishing, in infinite dimensional Hilbert spaces, ergodic $\mathcal{O}(1/n)$ primal-dual gap convergence whenever $\theta \geq 1/2$,

$$\tau\sigma\|L\|^2 \leq \frac{4}{1+2\theta}, \tag{1.5}$$

and one of these inequalities holds strictly, as well as weak convergence to a solution whenever both these inequalities are strict. The analysis is based on a Lyapunov inequality that has been derived with the aid of the methodology in [10]. The type of Lyapunov function that we are using is, to the best of our knowledge, novel and unrelated to the status of the algorithm as a preconditioned proximal point algorithm. We also establish tightness of the step size bound by providing an example of which sequence convergence is lost whenever

$$\tau\sigma\|L\|^2 \geq \frac{4}{1+2\theta}. \tag{1.6}$$

2. Preliminaries

Throughout this paper, $\mathcal{H}$ and $\mathcal{G}$ denote real Hilbert spaces. All norms $\|\cdot\|$ are canonical norms where the inner product will be clear from the context. Suppose that $f\colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is a proper, convex, and lower semicontinuous function. The *subdifferential* of f at a point $x \in \mathcal{H}$ is defined as

$$\partial f(x) = \{y \in \mathcal{H} \,|\, \forall z \in \mathcal{H}, f(z) \geq f(x) + \langle y, z - x\rangle\}\,. \tag{2.1}$$

The *proximal point* of f at $x \in \mathcal{H}$ with *step size* $\tau > 0$ is given by

$$\operatorname{prox}_{\tau f}(x) = \operatorname*{argmin}_{y\in\mathcal{H}} \left(f(y) + \frac{1}{2\tau}\|y - x\|^2\right),$$

which is uniquely defined. If $x, p \in \mathcal{H}$, we have from [1, Section 24.1] that

$$p = \operatorname{prox}_{\tau f}(x) \quad \text{if and only if} \quad \frac{1}{\tau}(x - p) \in \partial f(p) \tag{2.2}$$

and

$$f(\operatorname{prox}_{\tau f}(x)) < \infty. \tag{2.3}$$

Moreover, the *convex conjugate* of f, denoted $f^* : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, is the proper, convex, and lower semicontinuous function given by $f^*(u) = \sup_{x\in\mathcal{H}} (\langle u, x\rangle - f(x))$ for each $u \in \mathcal{H}$.

3. Main results

Our convergence results are derived under the following assumption.

Assumption 3.1
Assume that

i) $f\colon \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ *and* $g\colon \mathcal{G} \to \mathbb{R} \cup \{+\infty\}$ *are proper, convex, and lower semicontinuous,*

ii) $L\colon \mathcal{H} \to \mathcal{G}$ *is a nonzero bounded linear operator, and*

iii) there exists at least one point $(x_\star, y_\star) \in \mathcal{H} \times \mathcal{G}$ *such that both inclusions in (1.3) hold.*

One of our results concerns a primal-dual gap, which we introduce next. The function given by

$$\mathcal{L}(x,y) = f(x) + \langle y, Lx\rangle - g^*(y)$$

is the *Lagrangian function* associated with (1.1). Given a saddle point $(x_\star, y_\star) \in \mathcal{H}\times\mathcal{G}$ satisfying the KKT condition (1.3), we define the *primal-dual gap function* $\mathcal{D}_{x_\star,y_\star} : \mathcal{H}\times\mathcal{G} \to \mathbb{R}\cup\{+\infty\}$ as

$$\mathcal{D}_{x_\star,y_\star}(x,y) = \mathcal{L}(x,y_\star) - \mathcal{L}(x_\star,y) \tag{3.1}$$

for each $x, y \in \mathcal{H}\times\mathcal{G}$. It is straightforward to verify that $\mathcal{D}_{x_\star,y_\star}(x,y) \geq 0$ for each $x, y \in \mathcal{H}\times\mathcal{G}$ and it will serve as a replacement for function-value suboptimality. Our first main result shows $\mathcal{O}(1/K)$ ergodic convergence for this primal-dual gap.

Theorem 3.2 (Ergodic convergence rate)
Suppose that Assumption 3.1 holds. Let $(x_\star, y_\star) \in \mathcal{H}\times\mathcal{G}$ be an arbitrary point satisfying (1.3), let $(x_k)_{k=0}^\infty$ and $(y_k)_{k=0}^\infty$ be generated by (1.4) with $\theta \geq 1/2$ and $\tau, \sigma > 0$ satisfying $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$ and arbitrary $(x_0, y_0) \in \mathcal{H}\times\mathcal{G}$. Further, suppose that at least one of $\theta \geq 1/2$ and $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$ holds with strict inequality. Let $\bar{x}_K = (\sum_{k=1}^K x_k)/K$ and $\bar{y}_K = (\sum_{k=1}^K y_k)/K$. Then for each positive integer K, the ergodic duality gap $\mathcal{D}_{x_\star,y_\star}(\bar{x}_K, \bar{y}_K)$ converges as $\mathcal{O}(1/K)$.

Our second main result shows weak sequence convergence.

Theorem 3.3 (Weak convergence)
Suppose that Assumption 3.1 holds. Let $(x_k)_{k=0}^\infty$ and $(y_k)_{k=0}^\infty$ be generated by (1.4) with $\theta > 1/2$ and $\tau, \sigma > 0$ satisfying $\tau\sigma\|L\|^2 < 4/(1+2\theta)$ and arbitrary $(x_0, y_0) \in \mathcal{H}\times\mathcal{G}$. Then $(x_k, y_k) \rightharpoonup (x_\star, y_\star)$ for some KKT point $(x_\star, y_\star) \in \mathcal{H}\times\mathcal{G}$ satisfying (1.3).

Note that Theorem 3.2 allows for equality in $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$, while Theorem 3.3 requires strict inequality.

The remainder of the paper is devoted to proving these results and giving a tightness guarantee for Theorem 3.3. In Section 4, we introduce two lemmas, which subsequently find their application in Section 5, where our Lyapunov analysis is presented. Section 6 proves Theorems 3.2 and 3.3 while Section 7 provides an example for which the sequence fails to converge whenever $\tau\sigma\|L\|^2 \geq 4/(1+2\theta)$, implying that Theorem 3.3 is tight in this sense. Finally, Section 8 shows that $\|x_{k+2} - x_k\| \to 0$ on the boundary of the step-size condition, i.e., whenever $\tau\sigma\|L\|^2 = 4/(1+2\theta)$.

4. Two lemmas

Our Lyapunov analysis entails evaluating the primal-dual gap function $\mathcal{D}_{x_\star,y_\star}$ at points generated by the algorithm. To this end, we introduce the following definition.

Definition 4.1
Suppose that Assumption 3.1 holds. Let the sequences $(x_k)_{k=0}^\infty$ and $(y_k)_{k=0}^\infty$ be generated by (1.4) for some initial point $(x_0, y_0) \in \mathcal{H}\times\mathcal{G}$, and let $(x_\star, y_\star) \in \mathcal{H}\times\mathcal{G}$ satisfy (1.3). Then we define the sequences $(F_k)_{k=0}^\infty$ and $(G_k)_{k=0}^\infty$ by

$$F_k = f(x_{k+1}) - f(x_\star) + \langle L^* y_\star, x_{k+1} - x_\star\rangle$$

and

$$G_k = g^*(y_{k+1}) - g^*(y_\star) - \langle Lx_\star, y_{k+1} - y_\star\rangle,$$

respectively.

Next, we present two lemmas concerning F_k and G_k.

Lemma 4.2
Suppose that $(F_k)_{k=0}^\infty$ and $(G_k)_{k=0}^\infty$ are given as in Definition 4.1. Then

i) $F_k, G_k \geq 0$,

ii) $F_k + G_k < \infty$, *and*

iii) $F_k + G_k = \mathcal{D}_{x_\star, y_\star}(x_{k+1}, y_{k+1})$.

Proof. i) By (2.1) and (1.3a), we get that

$$f(x_{k+1}) \geq f(x_\star) - \langle L^* y_\star, x_{k+1} - x_\star\rangle,$$

which implies that F_k is nonnegative. By (2.1) and (1.3b), we get that

$$g^*(y_{k+1}) \geq g^*(y_\star) + \langle Lx_\star, y_{k+1} - y_\star\rangle,$$

which implies that G_k is nonnegative.

ii) By (2.3) and (1.3), we conclude that all terms defining F_k and G_k are finite. Hence $F_k + G_k < \infty$.

iii) We have

$$\begin{aligned}
&F_k + G_k \\
&= f(x_{k+1}) + \langle y_\star, Lx_{k+1}\rangle - g^*(y_\star) - (f(x_\star) + \langle y_{k+1}, Lx_\star\rangle - g^*(y_{k+1})) \\
&= \mathcal{L}(x_{k+1}, y_\star) - \mathcal{L}(x_\star, y_{k+1}).
\end{aligned}$$

□

We call $F_k + G_k$ a *primal-dual gap* at iteration k. Note that the primal-dual gap depends on the particular choice of KKT point $(x_\star, y_\star)$ and our analysis is valid for each such choice.

Before introducing the next lemma related to F_k and G_k, we conclude, using (2.2), that the iterative procedure in (1.4) is equivalent to the following set of inclusions:

$$\frac{1}{\tau}(x_k - x_{k+1}) - L^* y_k \in \partial f(x_{k+1}), \tag{4.1}$$

$$\frac{1}{\sigma}(y_k - y_{k+1}) + L(x_{k+1} + \theta(x_{k+1} - x_k)) \in \partial g^*(y_{k+1}). \tag{4.2}$$

Lemma 4.3
Suppose that $(F_k)_{k=0}^\infty$ and $(G_k)_{k=0}^\infty$ are given as in Definition 4.1. Then

$$\begin{aligned}
G_k &\leq \frac{1}{\sigma}\langle y_k - y_{k+1}, y_{k+1} - y_\star\rangle + \langle Lx_{k+1} - Lx_\star, y_{k+1} - y_\star\rangle \\
&\quad + \theta\langle Lx_{k+1} - Lx_k, y_{k+1} - y_\star\rangle, \\
F_{k+1} &\leq \frac{1}{\tau}\langle x_{k+1} - x_{k+2}, x_{k+2} - x_\star\rangle + \langle y_\star - y_{k+1}, Lx_{k+2} - Lx_\star\rangle, \\
F_{k+1} - F_k &\leq -\frac{1}{\tau}\|x_{k+2} - x_{k+1}\|^2 + \langle y_\star - y_{k+1}, Lx_{k+2} - Lx_{k+1}\rangle, \\
F_k - F_{k+1} &\leq \frac{1}{\tau}\langle x_k - x_{k+1}, x_{k+1} - x_{k+2}\rangle + \langle y_\star - y_k, Lx_{k+1} - Lx_{k+2}\rangle.
\end{aligned}$$

Proof. By (2.1) and (4.2),

$$\begin{aligned}
G_k &= g^*(y_{k+1}) - g^*(y_\star) - \langle Lx_\star, y_{k+1} - y_\star\rangle \\
&\leq \left\langle \frac{1}{\sigma}(y_k - y_{k+1}) + L(x_{k+1} + \theta(x_{k+1} - x_k)), y_{k+1} - y_\star \right\rangle \\
&\quad - \langle Lx_\star, y_{k+1} - y_\star\rangle \\
&= \frac{1}{\sigma}\langle y_k - y_{k+1}, y_{k+1} - y_\star\rangle + \langle Lx_{k+1} - Lx_\star, y_{k+1} - y_\star\rangle \\
&\quad + \theta\langle Lx_{k+1} - Lx_k, y_{k+1} - y_\star\rangle.
\end{aligned}$$

By (2.1) and (4.1),

$$\begin{aligned}
F_{k+1} &= f(x_{k+2}) - f(x_\star) + \langle L^*y_\star, x_{k+2} - x_\star\rangle \\
&\leq \left\langle \frac{1}{\tau}(x_{k+1} - x_{k+2}) - L^*y_{k+1}, x_{k+2} - x_\star \right\rangle + \langle L^*y_\star, x_{k+2} - x_\star\rangle \\
&= \frac{1}{\tau}\langle x_{k+1} - x_{k+2}, x_{k+2} - x_\star\rangle + \langle y_\star - y_{k+1}, Lx_{k+2} - Lx_\star\rangle
\end{aligned}$$

and

$$\begin{aligned}
&F_{k+1} - F_k \\
&= f(x_{k+2}) - f(x_{k+1}) + \langle L^*y_\star, x_{k+2} - x_{k+1}\rangle \\
&\leq \left\langle \frac{1}{\tau}(x_{k+1} - x_{k+2}) - L^*y_{k+1}, x_{k+2} - x_{k+1} \right\rangle + \langle L^*y_\star, x_{k+2} - x_{k+1}\rangle \\
&= -\frac{1}{\tau}\|x_{k+2} - x_{k+1}\|^2 + \langle y_\star - y_{k+1}, Lx_{k+2} - Lx_{k+1}\rangle
\end{aligned}$$

and

$$\begin{aligned}
F_k - F_{k+1} &= f(x_{k+1}) - f(x_{k+2}) + \langle L^*y_\star, x_{k+1} - x_{k+2}\rangle \\
&\leq \frac{1}{\tau}\langle x_k - x_{k+1}, x_{k+1} - x_{k+2}\rangle + \langle y_\star - y_k, Lx_{k+1} - Lx_{k+2}\rangle.
\end{aligned}$$

□

5. Lyapunov analysis

In this section, we provide a Lyapunov analysis of the Chambolle–Pock method, which serves as the basis for our convergence results. Before presenting this, we establish the nonnegativity of a certain coefficient that emerges in the Lyapunov analysis.

Lemma 5.1
Suppose that $\theta \geq 1/2$ and $\tau, \sigma > 0$, and that they jointly satisfy $0 < \tau\sigma\|L\|^2 \leq 4/(1+2\theta)$. Further suppose that at least one of $\theta \geq 1/2$ and $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$ holds with strict inequality. Then $8\theta - \tau\sigma\|L\|^2(4\theta^2+1)$ is positive.

Proof. We have

$$8\theta - \tau\sigma\|L\|^2(4\theta^2+1) \geq 8\theta - \frac{4(4\theta^2+1)}{1+2\theta} = \frac{8\theta - 4}{1+2\theta} = 4 - \frac{8}{1+2\theta} \geq 0,$$

where the first inequality is strict if $\tau\sigma\|L\|^2 < 4/(1+2\theta)$ and the last inequality is strict if $\theta > 1/2$. □

Lemma 5.2 (Lyapunov inequality)
Suppose that Assumption 3.1 holds. Let $(x_\star, y_\star) \in \mathcal{H} \times \mathcal{G}$ be an arbitrary point satisfying (1.3), let $(x_k)_{k=0}^\infty$ and $(y_k)_{k=0}^\infty$ be generated by (1.4) with $\theta \geq 1/2$ and $\tau, \sigma > 0$ satisfying $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$, and arbitrary $(x_0, y_0) \in \mathcal{H} \times \mathcal{G}$. Further, suppose that at least one of $\theta \geq 1/2$ and $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$ holds with strict inequality. Moreover, for each nonnegative integer k, let

$$
\begin{aligned}
V_k = \theta F_k &+ \frac{1}{2\tau}\|x_{k+1} - x_\star\|^2 + \frac{1}{2\sigma}\|y_k - y_\star + \sigma\theta(Lx_{k+1} - Lx_k)\|^2 \\
&+ \frac{\theta}{2\tau}\|x_{k+1} - x_k\|^2 - \frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+1} - Lx_k\|^2
\end{aligned}
$$

and F_k and G_k be as in Definition 4.1. Then, for each nonnegative integer k, $V_k \geq 0$ and

$$
\begin{aligned}
&V_{k+1} - V_k + F_k + G_k \\
&\leq -\frac{1}{2\sigma}\left\|y_{k+1} - y_k - \sigma\Big(\frac{1}{2}(Lx_{k+1} - Lx_{k+2}) - \theta(Lx_k - Lx_{k+1})\Big)\right\|^2 \\
&\quad - \frac{8\theta - \tau\sigma\|L\|^2(4\theta^2+1)}{16\tau}\left\|x_{k+2} - x_{k+1} - \frac{4(1-\tau\sigma\theta\|L\|^2)}{8\theta - \tau\sigma\|L\|^2(4\theta^2+1)}(x_{k+1} - x_k)\right\|^2 \\
&\quad - \frac{(4\theta^2-1)(4-\tau\sigma\|L\|^2(2\theta+1))(4-\tau\sigma\|L\|^2(2\theta-1))}{16\tau(8\theta - \tau\sigma\|L\|^2(4\theta^2+1))}\|x_{k+1} - x_k\|^2,
\end{aligned}
$$

where the latter inequality is called a Lyapunov inequality.

Proof. We start by proving nonnegativity of V_k. Since $\theta \geq 1/2$, $F_k \geq 0$ by Lemma 4.2, and $\tau, \sigma > 0$, we conclude that $V_k \geq 0$ since

$$
\begin{aligned}
&\frac{\theta}{2\tau}\|x_{k+1} - x_k\|^2 - \frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+1} - Lx_k\|^2 \\
&\geq \frac{\theta}{2\tau}\|x_{k+1} - x_k\|^2 - \frac{\sigma\|L\|^2(4\theta^2+1)}{16}\|x_{k+1} - x_k\|^2 \\
&= \frac{1}{\tau}\left(\frac{8\theta - \tau\sigma\|L\|^2(4\theta^2+1)}{16}\right)\|x_{k+1} - x_k\|^2 \geq 0,
\end{aligned}
$$

where the last inequality follows from Lemma 5.1.

We next prove the Lyapunov inequality. By Lemma 4.3, we have

$$
\begin{aligned}
&\theta F_{k+1} - \theta F_k + F_k + G_k \\
&= \Big(\theta - \frac{1}{2}\Big)(F_{k+1} - F_k) + F_{k+1} + \frac{1}{2}(F_k - F_{k+1}) + G_k \\
&\leq \Big(\theta - \frac{1}{2}\Big)\Big(-\frac{1}{\tau}\|x_{k+2} - x_{k+1}\|^2 + \langle y_\star - y_{k+1}, Lx_{k+2} - Lx_{k+1}\rangle\Big) \\
&\quad + \frac{1}{\tau}\langle x_{k+1} - x_{k+2}, x_{k+2} - x_\star\rangle + \langle y_\star - y_{k+1}, Lx_{k+2} - Lx_\star\rangle \\
&\quad + \frac{1}{2}\Big(\frac{1}{\tau}\langle x_k - x_{k+1}, x_{k+1} - x_{k+2}\rangle + \langle y_\star - y_k, Lx_{k+1} - Lx_{k+2}\rangle\Big) \\
&\quad + \frac{1}{\sigma}\langle y_k - y_{k+1}, y_{k+1} - y_\star\rangle + \langle Lx_{k+1} - Lx_\star, y_{k+1} - y_\star\rangle \\
&\quad + \theta\langle Lx_{k+1} - Lx_k, y_{k+1} - y_\star\rangle
\end{aligned}
$$

$$
\begin{aligned}
&= -\frac{\theta}{\tau}\|x_{k+2}-x_{k+1}\|^2+\theta\langle y_\star-y_{k+1}, Lx_{k+2}-2Lx_{k+1}+Lx_k\rangle \\
&\quad+\frac{1}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1}, x_{k+1}-x_{k+2}\rangle \\
&\quad+\frac{1}{\tau}\langle x_{k+1}-x_{k+2}, x_{k+2}-x_\star\rangle+\langle y_\star-y_{k+1}, Lx_{k+2}-Lx_{k+1}\rangle \\
&\quad+\frac{1}{2}\langle 2y_\star-y_k-y_{k+1}, Lx_{k+1}-Lx_{k+2}\rangle \\
&\quad+\frac{1}{\sigma}\langle y_k-y_{k+1}, y_{k+1}-y_\star\rangle.
\end{aligned}
$$

Moreover,

$$
\begin{aligned}
&\frac{1}{2\tau}\|x_{k+2}-x_\star\|^2-\frac{1}{2\tau}\|x_{k+1}-x_\star\|^2 \\
&=\frac{1}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{\tau}\langle x_{k+2}-x_{k+1}, x_{k+1}-x_\star\rangle
\end{aligned}
$$

and

$$
\begin{aligned}
&\frac{1}{2\sigma}\|y_{k+1}-y_\star+\sigma\theta(Lx_{k+2}-Lx_{k+1})\|^2-\frac{1}{2\sigma}\|y_k-y_\star+\sigma\theta(Lx_{k+1}-Lx_k)\|^2 \\
&=\frac{1}{2\sigma}\|y_{k+1}-y_k\|^2+\frac{1}{\sigma}\langle y_{k+1}-y_k, y_k-y_\star\rangle \\
&\quad+\frac{\sigma\theta^2}{2}\|Lx_{k+2}-Lx_{k+1}\|^2-\frac{\sigma\theta^2}{2}\|Lx_{k+1}-Lx_k\|^2 \\
&\quad+\theta\langle y_{k+1}-y_\star, Lx_{k+2}-Lx_{k+1}\rangle-\theta\langle y_k-y_\star, Lx_{k+1}-Lx_k\rangle.
\end{aligned}
$$

Hence,

$$
\begin{aligned}
&V_{k+1}-V_k+F_k+G_k \\
&=\theta F_{k+1}-\theta F_k+F_k+G_k+\frac{1}{2\tau}\|x_{k+2}-x_\star\|^2-\frac{1}{2\tau}\|x_{k+1}-x_\star\|^2 \\
&\quad+\frac{1}{2\sigma}\|y_{k+1}-y_\star+\sigma\theta(Lx_{k+2}-Lx_{k+1})\|^2 \\
&\quad-\frac{1}{2\sigma}\|y_k-y_\star+\sigma\theta(Lx_{k+1}-Lx_k)\|^2 \\
&\quad+\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2-\frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+2}-Lx_{k+1}\|^2 \\
&\quad-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2+\frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+1}-Lx_k\|^2
\end{aligned}
$$

$$
\begin{aligned}
\leq\ & -\frac{\theta}{\tau}\|x_{k+2}-x_{k+1}\|^2+\theta\langle y_\star-y_{k+1},Lx_{k+2}-2Lx_{k+1}+Lx_k\rangle\\
&+\frac{1}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1},x_{k+1}-x_{k+2}\rangle\\
&+\frac{1}{\tau}\langle x_{k+1}-x_{k+2},x_{k+2}-x_\star\rangle+\langle y_\star-y_{k+1},Lx_{k+2}-Lx_{k+1}\rangle\\
&+\frac{1}{2}\langle 2y_\star-y_k-y_{k+1},Lx_{k+1}-Lx_{k+2}\rangle\\
&+\frac{1}{\sigma}\langle y_k-y_{k+1},y_{k+1}-y_\star\rangle\\
&+\frac{1}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{\tau}\langle x_{k+2}-x_{k+1},x_{k+1}-x_\star\rangle\\
&+\frac{1}{2\sigma}\|y_{k+1}-y_k\|^2+\frac{1}{\sigma}\langle y_{k+1}-y_k,y_k-y_\star\rangle\\
&+\frac{\sigma\theta^2}{2}\|Lx_{k+2}-Lx_{k+1}\|^2-\frac{\sigma\theta^2}{2}\|Lx_{k+1}-Lx_k\|^2\\
&+\theta\langle y_{k+1}-y_\star,Lx_{k+2}-Lx_{k+1}\rangle-\theta\langle y_k-y_\star,Lx_{k+1}-Lx_k\rangle\\
&+\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2-\frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+2}-Lx_{k+1}\|^2\\
&-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2+\frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+1}-Lx_k\|^2\\
=\ & -\frac{1}{2\sigma}\|y_{k+1}-y_k\|^2+\left\langle y_{k+1}-y_k,\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\right\rangle\\
&-\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1},x_{k+1}-x_{k+2}\rangle-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2\\
&+\frac{\sigma(4\theta^2-1)}{16}\|Lx_{k+2}-Lx_{k+1}\|^2-\frac{\sigma(4\theta^2-1)}{16}\|Lx_{k+1}-Lx_k\|^2.
\end{aligned}
$$

Completing the square yields

$$
\begin{aligned}
&V_{k+1}-V_k+F_k+G_k\\
&\leq\ -\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2\\
&\quad+\frac{\sigma}{2}\left\|\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\right\|^2\\
&\quad-\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1},x_{k+1}-x_{k+2}\rangle-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2\\
&\quad+\frac{\sigma(4\theta^2-1)}{16}\|Lx_{k+2}-Lx_{k+1}\|^2-\frac{\sigma(4\theta^2-1)}{16}\|Lx_{k+1}-Lx_k\|^2
\end{aligned}
$$

$$
\begin{aligned}
= & -\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2 \\
& -\frac{\sigma\theta}{2}\langle Lx_{k+1}-Lx_{k+2}, Lx_k-Lx_{k+1}\rangle \\
& -\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1}, x_{k+1}-x_{k+2}\rangle-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2 \\
& +\frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+2}-Lx_{k+1}\|^2+\frac{\sigma(4\theta^2+1)}{16}\|Lx_{k+1}-Lx_k\|^2 \\
= & -\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2 \\
& +\frac{\sigma(4\theta^2+1)}{16}\left\|Lx_{k+2}-Lx_{k+1}+\frac{4\theta}{4\theta^2+1}(Lx_k-Lx_{k+1})\right\|^2 \\
& -\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1}, x_{k+1}-x_{k+2}\rangle-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2 \\
& +\frac{\sigma(4\theta^2-1)^2}{16(4\theta^2+1)}\|Lx_{k+1}-Lx_k\|^2.
\end{aligned}
$$

Using the Lipschitz continuity of L and that $\theta \geq 1/2$ give

$$
\begin{aligned}
& V_{k+1}-V_k+F_k+G_k \\
\leq & -\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2 \\
& +\frac{\sigma\|L\|^2(4\theta^2+1)}{16}\left\|x_{k+2}-x_{k+1}+\frac{4\theta}{4\theta^2+1}(x_k-x_{k+1})\right\|^2 \\
& -\frac{\theta}{2\tau}\|x_{k+2}-x_{k+1}\|^2+\frac{1}{2\tau}\langle x_k-x_{k+1}, x_{k+1}-x_{k+2}\rangle-\frac{\theta}{2\tau}\|x_{k+1}-x_k\|^2 \\
& +\frac{\sigma\|L\|^2(4\theta^2-1)^2}{16(4\theta^2+1)}\|x_{k+1}-x_k\|^2 \\
= & -\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2 \\
& -\frac{8\theta-\tau\sigma\|L\|^2(4\theta^2+1)}{16\tau}\|x_{k+2}-x_{k+1}\|^2 \\
& +\frac{1-\tau\sigma\theta\|L\|^2}{2\tau}\langle x_{k+2}-x_{k+1}, x_{k+1}-x_k\rangle \\
& -\frac{8\theta-\tau\sigma\|L\|^2(4\theta^2+1)}{16\tau}\|x_{k+1}-x_k\|^2 \\
= & -\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2 \\
& -\frac{8\theta-\tau\sigma\|L\|^2(4\theta^2+1)}{16\tau}\left\|x_{k+2}-x_{k+1}-\frac{4(1-\tau\sigma\theta\|L\|^2)}{8\theta-\tau\sigma\|L\|^2(4\theta^2+1)}(x_{k+1}-x_k)\right\|^2 \\
& -\frac{(4\theta^2-1)(4-\tau\sigma\|L\|^2(2\theta+1))(4-\tau\sigma\|L\|^2(2\theta-1))}{16\tau(8\theta-\tau\sigma\|L\|^2(4\theta^2+1))}\|x_{k+1}-x_k\|^2,
\end{aligned}
$$

where the last completion of squares equality holds since

$$(8\theta - \tau\sigma\|L\|^2(4\theta^2 + 1)) > 0$$

by Lemma 5.1. □

6. Theorem proofs

In this section, we use Lemma 5.2 to prove Theorems 3.2 and 3.3, i.e., we prove ergodic convergence of the primal-dual gap and weak convergence of the sequences $(x_k)_{k=0}^{\infty}$ and $(y_k)_{k=0}^{\infty}$ to a KKT point. Given the Lyapunov inequality, the arguments are standard. However, before we prove these results, we state the following lemma on nonnegativity of a coefficient in the Lyapunov inequality.

Lemma 6.1
Suppose that $\theta \geq 1/2$ and $\tau, \sigma > 0$ and that they jointly satisfy $0 < \tau\sigma\|L\|^2 \leq 4/(1+2\theta)$. Further suppose that at least one of $\theta \geq 1/2$ and $\tau\sigma\|L\|^2 \leq 4/(1+2\theta)$ holds with strict inequality. Then

$$\frac{(4\theta^2 - 1)(4 - \tau\sigma\|L\|^2(2\theta+1))(4 - \tau\sigma\|L\|^2(2\theta-1))}{16\tau(8\theta - \tau\sigma\|L\|^2(4\theta^2+1))} \geq 0$$

with strict inequality if $\tau\sigma\|L\|^2 < 4/(1+2\theta)$ and $\theta > 1/2$ and equality if $\tau\sigma\|L\|^2 = 4/(1+2\theta)$ or $\theta = 1/2$.

Proof. We know from Lemma 5.1 that the denominator is positive. The numerator consists of three factors. The first factor is positive if $\theta > 1/2$ and 0 if $\theta = 1/2$. The third factor is larger than the second. The second factor is positive if $\tau\sigma\|L\|^2 < 4/(1+2\theta)$ and 0 if $\tau\sigma\|L\|^2 = 4/(1+2\theta)$. Combining these gives the result. □

6.1 Proof of Theorem 3.2

Proof. Due to Lemmas 4.2, 5.1, 5.2 and 6.1, we conclude, using a telescoping summation, that $F_k + G_k$ is summable and satisfies

$$0 \leq \sum_{k=0}^{K-1} (F_k + G_k) \leq V_0 < \infty$$

for each positive integer K. From Jensen's inequality, we conclude that

$$\mathcal{D}_{x_\star, y_\star}(\bar{x}_K, \bar{y}_K) \leq \frac{1}{K} \sum_{k=0}^{K-1} \mathcal{D}(x_{k+1}, y_{k+1}) = \frac{1}{K} \sum_{k=0}^{K-1} (F_k + G_k) \leq \frac{V_0}{K},$$

where $\mathcal{D}_{x_\star, y_\star}$ is defined in (3.1) and the equality follows from Lemma 4.2. □

6.2 Proof of Theorem 3.3

Proof. Due to Lemmas 4.2, 5.1, 5.2 and 6.1, we conclude that $(V_k)_{k=0}^{\infty}$ converges. This in turn implies that $(x_k)_{k=0}^{\infty}$ and $(y_k)_{k=0}^{\infty}$ remain bounded. Next, the same lemmas also imply, using a telescoping summation, that $F_k \to 0$,

$$\frac{(4\theta^2 - 1)(4 - \tau\sigma\|L\|^2(2\theta+1))(4 - \tau\sigma\|L\|^2(2\theta-1))}{16\tau(8\theta - \tau\sigma\|L\|^2(4\theta^2+1))} \|x_{k+1} - x_k\|^2 \to 0, \qquad (6.1)$$

and

$$\frac{1}{2\sigma}\left\|y_{k+1}-y_k-\sigma\Big(\frac{1}{2}(Lx_{k+1}-Lx_{k+2})-\theta(Lx_k-Lx_{k+1})\Big)\right\|^2\to 0 \tag{6.2}$$

as $k\to\infty$. Since the coefficients are nonzero due to the parameter assumptions and Lemma 6.1, we conclude from (6.1) that

$$\|x_{k+1}-x_k\|\to 0 \tag{6.3}$$

as $k\to\infty$. Combining (6.2) and (6.3) gives that

$$\|y_{k+1}-y_k\|\to 0 \tag{6.4}$$

as $k\to\infty$, since L is a bounded linear operator. Moreover, (6.3) combined with the fact $(V_k)_{k=0}^{\infty}$ converges and $F_k\to 0$ as $k\to\infty$ give that

$$\left(\frac{1}{2\tau}\|x_{k+1}-x_\star\|^2+\frac{1}{2\sigma}\|y_k-y_\star+\sigma\theta(Lx_{k+1}-Lx_k)\|^2\right)_{k=0}^{\infty}$$

converges. Next, since $(y_k)_{k=0}^{\infty}$ is bounded, (6.3) and (6.4) imply that

$$\begin{aligned}&\|y_{k+1}-y_\star\|^2-\|y_k-y_\star+\sigma\theta(Lx_{k+1}-Lx_k)\|^2\\&=\|y_{k+1}-y_k\|^2-2\langle y_k-y_\star,\sigma\theta(Lx_{k+1}-Lx_k)-(y_{k+1}-y_k)\rangle\\&\quad-\|\sigma\theta(Lx_{k+1}-Lx_k)\|^2\to 0\end{aligned}$$

as $k\to\infty$, from which we conclude that

$$\left(\frac{1}{2\tau}\|x_{k+1}-x_\star\|^2+\frac{1}{2\sigma}\|y_{k+1}-y_\star\|^2\right)_{k=0}^{\infty}$$

converges.

Construct the sequence $(z_k)_{k=0}^{\infty}$ such that $z_k=(x_{k+1},y_{k+1})$ for each nonnegative integer k and let $Z=\{(x_\star,y_\star)\in\mathcal{H}\times\mathcal{G}:(1.3)\text{ holds}\}$ be the KKT solution set. Since $(z_k)_{k=0}^{\infty}$ is bounded, it has weakly convergent subsequences. Let $(z_{k_n})_{n=0}^{\infty}$ be such a subsequence, with weak limit point $\bar{z}=(\bar{x},\bar{y})\in\mathcal{H}\times\mathcal{G}$ say. Then the inclusion formulation of the algorithm in (4.1) and (4.2) satisfies

$$\begin{aligned}\partial f(x_{k_n+1})+L^*y_{k_n}&\ni\frac{1}{\tau}(x_{k_n}-x_{k_n+1})\to 0,\\ \partial g^*(y_{k_n+1})-Lx_{k_n+1}&\ni\frac{1}{\sigma}(y_{k_n}-y_{k_n+1})+\theta L(x_{k_n+1}-x_{k_n})\to 0,\end{aligned}$$

as $n\to\infty$, since $x_{k_n+1}-x_{k_n}\to 0$ and $y_{k_n+1}-y_{k_n}\to 0$ as $n\to\infty$. Let us introduce the operator $A:\mathcal{H}\times\mathcal{G}\rightrightarrows\mathcal{H}\times\mathcal{G}$ defined as

$$A(x,y)=\partial f(x)\times\partial g^*(y)+(L^*y,-Lx)$$

for each $(x,y)\in\mathcal{H}\times\mathcal{G}$. This is a maximally monotone operator since it is the sum of two maximally monotone operators, the first one due to [1, Theorem 20.25, Proposition 20.23] and the second one due to [1, Example 20.35], one of which has full domain [1, Corollary 25.5]. Therefore,

$$\underbrace{A(x_{k_n+1},y_{k_n+1})}\ni\left(\frac{1}{\tau}(x_{k_n}-x_{k_n+1})-L^*(y_{k_n}-y_{k_n+1}),\frac{1}{\sigma}(y_{k_n}-y_{k_n+1})+\theta L(x_{k_n+1}-x_{k_n})\right)\to 0$$

and we conclude, using weak-strong closedness of maximally monotone operators [1, Proposition 20.38], that $0 \in A(\bar{x}, \bar{y})$, implying

$$0 \in \partial f(\bar{x}) + L^* \bar{y},$$
$$0 \in \partial g^*(\bar{y}) - L\bar{x},$$

i.e., the weak limit point is a KKT point. Since

$$\left(\frac{1}{2\tau} \|x_{k+1} - x_\star\|^2 + \frac{1}{2\sigma} \|y_{k+1} - y_\star\|^2 \right)_{k=0}^{\infty}$$

converges and all weak limit points belong to the KKT solution set Z, we invoke [1, Lemma 2.47] to conclude weak convergence to a KKT point. $\square$

7. Counterexample

This section proves that the result in Theorem 3.3 is tight, i.e., there is an example where (weak) convergence fails for each $\theta > 1/2$ whenever $\tau\sigma\|L\|^2 \geq 4/(1+2\theta)$. Let $\mathcal{H} = \mathbb{R}$ and $\mathcal{G} = \mathbb{R}$, and consider problem (1.1) for the case $f = g^* = 0$, and $L = 1$, i.e.,

$$\underset{x \in \mathbb{R}}{\text{minimize}} \, \underset{y \in \mathbb{R}}{\text{maximize}} \;\; xy. \tag{7.1}$$

The update rule, (1.4), for the Chambolle–Pock method then becomes

$$x_{k+1} = x_k - \tau y_k,$$
$$y_{k+1} = y_k + \sigma(x_{k+1} + \theta(x_{k+1} - x_k)),$$

or equivalently

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & -\tau \\ \sigma & 1 - \tau\sigma(1+\theta) \end{bmatrix} \begin{bmatrix} x_k \\ y_k \end{bmatrix},$$

where we assume that $\tau, \sigma > 0$. Therefore, $(x_k, y_k)_{k=0}^{\infty}$ converges for an arbitrary initial point $(x_0, y_0) \in \mathbb{R}^2$ to the solution $(x_\star, y_\star) = (0, 0) \in \mathbb{R}^2$ if and only if both eigenvalues of the matrix above have magnitude less than 1. The two eigenvalues of the matrix are

$$\lambda_1 = \frac{1}{2}\left(2 - \tau\sigma(1+\theta) + \sqrt{\tau\sigma(\tau\sigma(1+\theta)^2 - 4)} \right)$$

and

$$\lambda_2 = \frac{1}{2}\left(2 - \tau\sigma(1+\theta) - \sqrt{\tau\sigma(\tau\sigma(1+\theta)^2 - 4)} \right).$$

Suppose that the step-size condition of Theorem 3.3 is violated, i.e., that $\tau\sigma \geq 4/(1+2\theta)$. In this case, the eigenvalues are distinct real numbers satisfying

$$\lambda_2 \leq -1 \quad \text{and} \quad \lambda_2 < \lambda_1 < \frac{1}{2}\left(2 - \tau\sigma(1+\theta) + \sqrt{\tau^2\sigma^2(1+\theta)^2} \right) = 1.$$

In particular, the method fails to converge, which proves that the result in Theorem 3.3 is tight.

Note that the boundary condition $\tau\sigma = 4/(1+2\theta)$ implies that $\lambda_2 = -1$. Although the sequence $(x_k, y_k)_{k=0}^{\infty}$ does not necessarily converge in this setting, we have that $(x_{k+2}, y_{k+2}) - (x_k, y_k) \to 0$ as $k \to \infty$, i.e., the difference between every second iterate converges to 0. It turns out that this is not a coincidence as it always holds for the Chambolle–Pock algorithm on the boundary, as shown next.

8. Boundary

In this section, we show that whenever $\theta > 1/2$ and $\tau\sigma\|L\|^2 = 4/(1+2\theta)$, the sequence $(\|x_{k+2} - x_k\|^2)_{k=0}^{\infty}$ is summable, implying that the difference between every second iterate converges to 0.

Due to Lemma 6.1, we conclude that the coefficient in front of $\|x_{k+1} - x_k\|^2$ in the Lyapunov inequality in Lemma 5.2 is 0. Moreover,

$$\frac{8\theta - \tau\sigma\|L\|^2(4\theta^2+1)}{16\tau} = \frac{8\theta(1+2\theta) - 4(4\theta^2+1)}{16\tau(1+2\theta)} = \frac{2\theta - 1}{4\tau(1+2\theta)}$$

and

$$\begin{aligned}\frac{4(1-\tau\sigma\theta\|L\|^2)}{8\theta - \tau\sigma\|L\|^2(4\theta^2+1)} &= \frac{4(1-2\theta)}{(8\theta - \tau\sigma\|L\|^2(4\theta^2+1))(1+2\theta)} \\ &= \frac{4(1-2\theta)(1+2\theta)}{(8\theta-4)(1+2\theta)} \\ &= -1,\end{aligned}$$

implying that the Lyapunov inequality in Lemma 5.2 reads

$$\begin{aligned}&V_{k+1} - V_k + F_k + G_k \\ &\leq -\frac{1}{2\sigma}\left\| y_{k+1} - y_k - \sigma\Big(\frac{1}{2}(Lx_{k+1} - Lx_{k+2}) - \theta(Lx_k - Lx_{k+1})\Big)\right\|^2 \\ &\quad - \frac{2\theta-1}{4\tau(1+2\theta)}\|x_{k+2} - x_k\|^2,\end{aligned}$$

where V_k, F_k, and G_k are nonnegative for all nonnegative integers k. Since $\theta > 1/2$, we conclude using a telescoping summation argument that $(\|x_{k+2} - x_k\|^2)_{k=0}^{\infty}$ indeed is summable.

Acknowledgments

This work was partially supported by the ELLIIT Strategic Research Area and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. S. Banert and P. Giselsson acknowledge support from Vetenskapsrådet grant VR 2021-05710.

References

[1] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer International Publishing, 2017. ISBN: 9783319483115. DOI: 10.1007/978-3-319-48311-5.

[2] A. Chambolle and T. Pock. "A first-order primal-dual algorithm for convex problems with applications to imaging". *Journal of Mathematical Imaging and Vision* **40**:1 (2011), pp. 120–145. DOI: 10.1007/s10851-010-0251-1.

[3] L. Condat. "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms". *Journal of Optimization Theory and Applications* **158**:2 (2012), pp. 460–479. DOI: 10.1007/s10957-012-0245-9.

[4] P. Giselsson. "Nonlinear forward-backward splitting with projection correction". *SIAM Journal on Optimization* **31**:3 (2021), pp. 2199–2226. DOI: 10.1137/20m1345062.

[5] B. He, F. Ma, S. Xu, and X. Yuan. "A generalized primal–dual algorithm with improved convergence condition for saddle point problems". *SIAM Journal on Imaging Sciences* **15**:3 (2022), pp. 1157–1183. DOI: 10.1137/21M1453463.

[6] B. He and X. Yuan. "Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective". *SIAM Journal on Imaging Sciences* **5**:1 (2012), pp. 119–149. DOI: 10.1137/100814494.

[7] P. Latafat and P. Patrinos. "Asymmetric forward-backward-adjoint splitting for solving monotone inclusions involving three operators". *Computational Optimization and Applications* **68**:1 (2017), pp. 57–93. DOI: 10.1007/s10589-017-9909-6.

[8] Y. Ma, X. Cai, B. Jiang, and D. Han. "Understanding the convergence of the preconditioned PDHG method: A view of indefinite proximal ADMM". *Journal of Scientific Computing* **94**:3 (2023). DOI: 10.1007/s10915-023-02105-9.

[9] M. Morin, S. Banert, and P. Giselsson. "Nonlinear forward-backward splitting with momentum correction". *Set-Valued and Variational Analysis* **31**:4 (2023). DOI: 10.1007/s11228-023-00700-4.

[10] M. Upadhyaya, S. Banert, A. B. Taylor, and P. Giselsson. "Automated tight Lyapunov analysis for first-order methods". *Mathematical Programming* **209** (2025), pp. 133–170. DOI: 10.1007/s10107-024-02061-8.

[11] M. Yan and Y. Li. "On the improved conditions for some primal-dual algorithms". *Journal of Scientific Computing* **99**:3 (2024). DOI: 10.1007/s10915-024-02537-x.

Paper IV

A Lyapunov analysis of Korpelevich's extragradient method with fast and flexible extensions

Manu Upadhyaya Puya Latafat Pontus Giselsson

Abstract

We present a Lyapunov analysis of Korpelevich's extragradient method and establish a $o(1/k)$ last-iterate convergence rate of the associated Lyapunov function. Building on this, we propose flexible extensions that combine extragradient steps with user-specified directions, guided by a line-search procedure derived from the same Lyapunov analysis. These methods retain global convergence under practical assumptions and can achieve superlinear rates when directions are chosen appropriately. Numerical experiments highlight the simplicity and efficiency of this approach.

Keywords. Monotone inclusions, extragradient method, Lyapunov analysis, superlinear convergence

1. Introduction

In this work, we consider the inclusion problem

$$\text{find } z \in \mathcal{H} \text{ such that } 0 \in F(z) + \partial g(z), \tag{1.1}$$

where $F : \mathcal{H} \to \mathcal{H}$ is monotone and L_F-Lipschitz continuous for some $L_F \in \mathbb{R}_{++}$, $g : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is a proper, convex, and lower semicontinuous function, and $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ is a real Hilbert space. Inclusion problems of the form (1.1) are known as *hemivariational inequalities* [26] or *(mixed) variational inequalities* [24, 30], and frequently arise in fundamental mathematical programming problems—either directly or through reformulation—including minimization, saddle-point, complementarity, Nash equilibrium, and fixed-point problems [6, 16]. The most common methods for solving (1.1) belong to the large class of extragradient-type methods [22, 34, 44]. For a recent review, see [42, 43]. Among these methods, the first and most widely recognized is Korpelevich's extragradient method [22]. Although originally proposed for the constrained case in which g is the indicator function of a nonempty, closed, and convex set, the method also applies to the more general setting in (1.1). Specifically, given an initial point $z^0 \in \mathcal{H}$ and a step-size parameter $\gamma \in (0, 1/L_F)$, its iterations are given by

$$\bar{z}^k = \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(z^k)\right), \tag{1.2a}$$

$$z^{k+1} = \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(\bar{z}^k)\right) \tag{1.2b}$$

for each $k \in \mathbb{N}_0$. A popular alternative is Tseng's forward-backward-forward method [44], given by

$$\bar{z}^k = \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(z^k)\right), \tag{1.3a}$$

$$z^{k+1} = \bar{z}^k + \gamma(F(z^k) - F(\bar{z}^k)) \tag{1.3b}$$

for each $k \in \mathbb{N}_0$, that requires one less evaluation of the proximal operator $\operatorname{prox}_{\gamma g}$ per iteration. Classically, the convergence analyses of these methods rely on Fejér-type arguments [22, 44].

In this work, we propose an analysis centered around the Lyapunov function

$$\mathcal{V}(z^k, \bar{z}^k, z^{k+1}) = 2\gamma^{-1}\langle z^k - z^{k+1}, F(z^k) - F(\bar{z}^k)\rangle + \gamma^{-2}\|z^{k+1} - \bar{z}^k\|^2 + \gamma^{-2}\|z^k - z^{k+1}\|^2.$$

For the extragradient method, $\mathcal{V}_k$ serves as a nonnegative optimality measure for the inclusion problem (1.1) as shown in Proposition 2.2. Likewise, $\mathcal{V}_k$ is a nonnegative optimality measure for Tseng's method, since the Lyapunov function reduces to $\mathcal{V}_k = \gamma^{-2}\|z^k - \bar{z}^k\|^2$ in this case. In the particular case when $g = 0$, both methods are identical, and the Lyapunov function reduces to $\mathcal{V}_k = \|F(z^k)\|^2$.

Besides being an optimality measure, we show in Theorem 2.3 that $\mathcal{V}_k$ satisfies a descent inequality for the extragradient method. Moreover, for the extragradient method, we show a Fejér-type inequality in which $\mathcal{V}_k$ appears as the residual term (see Theorem 2.5). By combining this result with the descent property, we establish a $o(1/k)$ last-iterate convergence rate for $\mathcal{V}_k$ for the extragradient method, as shown in Corollary 2.6. In Appendix C, we show that $\mathcal{V}_k$ upper bounds some common optimality measures used to judge the quality of approximate solutions for (1.1); hence, together with Corollary 2.6, the same last-iterate convergence result automatically holds for each of those measures as well. Taken together, the results we obtain for $\mathcal{V}_k$ enable us to recover and, in some cases, extend recent last-iterate convergence-rate results for the extragradient method (cf. [20, Theorem 3.3], [11, Theorem 3], and [43, Corollary 4.1]).

Interestingly, Theorem 2.3 is particular to Korpelevich's extragradient method. We demonstrate through a simple counterexample in Example B.1 that the descent inequality in terms of $\mathcal{V}_k$ fails for Tseng's method. Moreover, even for the extragradient method, it is crucial to leverage the specific structure of (1.1). Indeed, the claimed descent inequality fails, and even the convergence of the method does not hold if we replace ∂g with a maximally monotone operator $T : \mathcal{H} \to 2^{\mathcal{H}}$ and correspondingly the proximal operators $\operatorname{prox}_{\gamma g}$ in (1.2) with the resolvent $(\mathrm{Id} + \gamma T)^{-1}$. This broader setting is ruled out by a counterexample presented in Example B.2. However, if T is also 3-cyclically monotone (see [6, Definition 22.13]), then Theorems 2.3 and 2.5 (and therefore also Corollary 2.6) remain valid; see Remark 2.7 for details.

The second objective of this work is to develop flexible extragradient-type schemes that accommodate fast local directions while maintaining global convergence. In this regard, the seminal work [38] proposes a hybrid method for solving monotone equations, i.e., when $g = 0$ in (1.1). Their scheme achieves global convergence by blending an inexact regularized Newton step with the hyperplane projection framework from [39]. At each iteration, a search direction is computed based on an inexact regularized Newton step. A line search is then performed along this direction, not to decrease a merit function, but to identify a hyperplane separating the current iterate from the solution set. The algorithm then proceeds by projecting the iterate onto this hyperplane. While this approach incorporates a line search, the convergence analysis still fundamentally relies on Fejér-type monotonicity arguments. In practice, however, the projection step can undermine the effectiveness of the Newtonian directions, resulting in slower convergence.

Another related work to ours is [41], which addresses the problem of finding fixed points of averaged operators. They propose a hybrid scheme accelerating many numerical algorithms under the Krasnosel'skiĭ–Mann framework. Similar to [38], their scheme incorporates a hyperplane projection step and achieves superlinear convergence under suitable assumptions. In addition, it allows for a general class of local directions, including quasi-Newton-type directions, providing greater flexibility in practice.

In contrast to the approaches mentioned earlier that use a line search to identify a separating hyperplane onto which the iterates are projected (see [38, 41]), our proposed schemes incorporate line search procedures grounded in our new Lyapunov analysis, directly aiming to reduce $\mathcal{V}_k$. We introduce three flexible algorithms tailored to specific instances of problem (1.1). `FLEX` (Algorithm 1) is introduced for finding zeros of F, `I-FLEX` (Algorithm 2) is applicable when F is injective, and `Prox-FLEX` (Algorithm 3) addresses problem (1.1) in its full generality. All three algorithms share the same guiding principle: at each iteration, one performs a convex combination of a standard extragradient step (1.2) and a step based on a user-specified direction. The specific weighting for this convex combination is determined by the line-search procedure, ensuring sufficient descent of the optimality measure $\mathcal{V}_k$. Similar to [41], our schemes accommodate a wide range of user-chosen directions, including quasi-Newton-type directions. A key feature enabling this approach is that $\mathcal{V}_k$ depends solely on values computed at each iteration and does not involve a solution to (1.1). This design ensures high flexibility while guaranteeing global (see Section 3) and superlinear convergence (see Section 5) when choosing suitable directions.

Our preliminary numerical experiments indicate that using quasi-Newton directions in our proposed algorithms yields favorable performance. In particular, limited-memory type-I and type-II Anderson acceleration exhibit promising results (see Section 6). In related work, [48] studies Anderson acceleration for finding fixed points of averaged operators, proposing a globalization strategy based on a stabilization and safeguarding mechanism—rather than a line search—that reverts to a nominal Krasnosel'skiĭ–Mann step whenever the Anderson acceleration step fails to sufficiently reduce the forward residual. More recently, [35] introduced an extragradient-based scheme with memory-one Anderson acceleration, which reduces overhead and allows for simple, explicit updates of the directions. Furthermore, [4] presents a quasi-Newton method tailored to minimax problems. Our theory offers a direct globalization strategy for such directions, applicable in the uniformly monotone and injective settings (see Theorem 3.4), or whenever the resulting directions are summable (see Theorem 3.2.(i)).

1.1 Organization

In Section 2, we formally introduce the new Lyapunov analysis for Korpelevich's extragradient method. Building on this framework, we present the three new algorithms in Section 3 and establish their global convergence under suitable assumptions. In Section 4, we provide detailed proofs of the results from the preceding section. Next, Section 5 focuses on superlinear convergence, including corresponding proofs for the proposed algorithms. Numerical experiments appear in Section 6, and we conclude in Section 7 with a summary of key findings and directions for future research. Finally, Appendix A offers background material on Korpelevich's extragradient method, Appendix B presents the counterexamples mentioned earlier, and Appendix C contains a comparison to recent last-iterate convergence results for the extragradient method.

1.2 Notation and preliminaries

Let $\mathbb{N}_0$ denote the set of nonnegative integers, $\mathbb{N}$ the set of positive integers, $\mathbb{Z}$ the set of integers, $[\![n,m]\!] = \{l \in \mathbb{Z} \mid n \leq l \leq m\}$ the set of integers inclusively between the integers n and m, $\mathbb{R}$ the set of real numbers, $\mathbb{R}_+$ the set of nonnegative real numbers, $\mathbb{R}_{++}$ the set of positive real numbers, $\mathbb{R}^n$ the set of all n-tuples of elements of $\mathbb{R}$, $\mathbb{R}^{m\times n}$ the set of real-valued matrices of size $m \times n$, if $M \in \mathbb{R}^{m\times n}$ then $[M]_{i,j}$ denotes the i,j-th element of M, $\mathbb{S}^n$ the set of symmetric real-valued matrices of size $n \times n$, and $\mathbb{S}^n_+ \subseteq \mathbb{S}^n$ the set of positive semidefinite real-valued matrices of size $n \times n$. Suppose that $1 \leq p < +\infty$, $K \subseteq \mathbb{N}_0$, and $\mathcal{U} \subseteq \mathcal{W}$, where $\mathcal{W}$ is a normed space. Then we define the space $\ell^p(K;\mathcal{U}) = \{(u^k)_{k\in K} \in \mathcal{U}^K \mid \sum_{k\in K}\|u^k\|^p < +\infty\}$.

Throughout this paper, $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ will denote a real Hilbert space and $\|\cdot\|$ the canonical norm, which will be clear from the context. Let $F : \mathcal{H} \to \mathcal{H}$ be an operator. Suppose that $L_F \geq 0$. The operator F is said to be *L_F-Lipschitz continuous* if $\|F(x)-F(y)\| \leq L_F\|x-y\|$ for each $x, y \in \mathcal{H}$. The operator F is said to be *monotone* if $0 \leq \langle F(x) - F(y), x - y\rangle$ for each $x, y \in \mathcal{H}$. Suppose that $\mu_F \geq 0$. The operator F is said to be *μ_F-strongly monotone* if $\mu_F\|x-y\|^2 \leq \langle F(x) - F(y), x - y\rangle$ for each $x, y \in \mathcal{H}$. Moreover, F is said to be *uniformly monotone with modulus* $\phi : \mathbb{R}_+ \to \mathbb{R}_+$ if ϕ is increasing, vanishes only at 0, and $\phi(\|x-y\|) \leq \langle F(x) - F(y), x - y\rangle$ for each $x, y \in \mathcal{H}$. For a general set-valued operator $T : \mathcal{H} \to 2^{\mathcal{H}}$, the set of *zeros* is denoted by $\operatorname{zer}(T) = \{x \in \mathcal{H} \mid 0 \in T(x)\}$. The Cauchy–Schwarz inequality states that $|\langle x, y\rangle| \leq \|x\|\|y\|$ for each $x, y \in \mathcal{H}$ and Young's inequality that $2\langle x, y\rangle \leq \alpha\|x\|^2 + \alpha^{-1}\|y\|^2$ for each $x, y \in \mathcal{H}$ and $\alpha > 0$.

Given a function $g : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, the *effective domain* of g is the set $\operatorname{dom} g = \{x \in \mathcal{H} \mid g(x) < +\infty\}$. The function g is said to be *proper* if $\operatorname{dom} g \neq \emptyset$. The *subdifferential* of a proper function g is the set-valued operator $\partial g : \mathcal{H} \to 2^{\mathcal{H}}$ defined as the mapping $x \mapsto \{u \in \mathcal{H} \mid \forall y \in \mathcal{H},\ g(y) \geq g(x) + \langle u, y - x\rangle\}$. The function g is said to be *convex* if $g((1-\lambda)x + \lambda y) \leq (1-\lambda)g(x) + \lambda g(y)$ for each $x, y \in \mathcal{H}$ and $0 \leq \lambda \leq 1$. The function g is said to be *lower semicontinuous* if $\liminf_{y\to x} g(y) \geq g(x)$ for each $x \in \mathcal{H}$. If $C \subseteq \mathcal{H}$, the *indicator function of* C, denoted $\delta_C : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$, is defined as $\delta_C(x) = 0$ if $x \in C$ and $\delta_C(x) = +\infty$ if $x \in \mathcal{H} \setminus C$.

Let $g : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ be proper, convex and lower semicontinuous, and let $\gamma > 0$. Then the *proximal operator* $\operatorname{prox}_{\gamma g} : \mathcal{H} \to \mathcal{H}$ is defined as the single-valued operator given by

$$\operatorname{prox}_{\gamma g}(x) = \operatorname*{argmin}_{z \in \mathcal{H}} \left(g(z) + \tfrac{1}{2\gamma}\|x - z\|^2 \right)$$

for each $x \in \mathcal{H}$ [6, Proposition 12.15]. If $x, p \in \mathcal{H}$, then $p = \operatorname{prox}_{\gamma g}(x) \Leftrightarrow \gamma^{-1}(x - p) \in \partial g(p) \Leftrightarrow 0 \leq g(y) - g(p) - \langle \gamma^{-1}(x - p), y - p \rangle$ for each $y \in \mathcal{H}$ [6, Proposition 16.44, Proposition 16.6].

2. A new Lyapunov analysis

Classical convergence analyses of Korpelevich's extragradient method (1.2) typically rely on Fejér-type arguments, as discussed in Appendix A. In this section, we introduce a complementary Lyapunov inequality that not only leads to a last-iterate result but also forms the basis of the new algorithms presented in Section 3. Throughout this work, we investigate (1.1) under the following assumption.

Assumption 2.1
The following hold in problem (1.1).

(i) $F : \mathcal{H} \to \mathcal{H}$ *is monotone and* L_F*-Lipschitz continuous for some* $L_F \in \mathbb{R}_{++}$.

(ii) $g : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ *is proper, convex and lower semicontinuous.*

Our analysis is centered around the Lyapunov function $\mathcal{V} : \mathcal{H}^3 \to \mathbb{R}$ given by

$$\mathcal{V}(z, \bar{z}, z^+) = 2\gamma^{-1}\langle z - z^+, F(z) - F(\bar{z})\rangle + \gamma^{-2}\|z^+ - \bar{z}\|^2 + \gamma^{-2}\|z - z^+\|^2 \tag{2.1}$$

for each $(z, \bar{z}, z^+) \in \mathcal{H}^3$. Proposition 2.2 establishes that $\mathcal{V}$ is generally a valid optimality measure for the inclusion problem in (1.1). For notational convenience, we define the algorithmic operators $T_1^\gamma, T_2^\gamma : \mathcal{H} \to \mathcal{H}$ by

$$T_1^\gamma = \operatorname{prox}_{\gamma g} \circ (\mathrm{Id} - \gamma F) \quad \text{and} \quad T_2^\gamma = \operatorname{prox}_{\gamma g} \circ (\mathrm{Id} - \gamma F \circ T_1^\gamma), \tag{2.2}$$

where $\gamma \in \mathbb{R}_{++}$ is the step-size parameter. With this notation, the iterates of (1.2) can be written compactly as $\bar{z}^k = T_1^\gamma(z^k)$ and $z^{k+1} = T_2^\gamma(z^k)$.

Proposition 2.2
Suppose that Assumption 2.1 holds. Let $\gamma \in (0, 1/L_F)$*,* $z \in \mathcal{H}$*,* $\bar{z} = T_1^\gamma(z)$ *and* $z^+ = T_2^\gamma(z)$ *where* T_1^γ *and* T_2^γ *are the algorithmic operators defined in (2.2), and* $\mathcal{V}$ *the Lyapunov function defined in (2.1). Then the following hold.*

(i) $\mathcal{V}(z, \bar{z}, z^+) \geq (1 - \gamma L_F)\gamma^{-2}(\|z^+ - \bar{z}\|^2 + \|z^+ - z\|^2) \geq 0$.

(ii) $\mathcal{V}(z, \bar{z}, z^+) = 0$ *if and only if* $z = \bar{z} = z^+ \in \operatorname{zer}(F + \partial g)$.

Proof.

2.2.(i): The inner product in the definition of $\mathcal{V}$ can be written as

$$\langle z - z^+, F(z) - F(\bar{z})\rangle = \langle z - z^+, F(z) - F(z^+)\rangle + \langle z - z^+, F(z^+) - F(\bar{z})\rangle$$

$$
\begin{aligned}
&\geq \langle z - z^+, F(z^+) - F(\bar{z})\rangle \\
&\geq -\|z - z^+\|\|F(z^+) - F(\bar{z})\| \\
&\geq -L_F\|z - z^+\|\|z^+ - \bar{z}\| \\
&\geq -\tfrac{L_F}{2}(\|z - z^+\|^2 + \|z^+ - \bar{z}\|^2),
\end{aligned}
$$

where monotonicity of F is used in the first inequality, the Cauchy–Schwarz inequality is used in the second inequality, Lipschitz continuity of F is used in the third inequality, and Young's inequality for products is used in the fourth inequality. The lower bound of $\mathcal{V}$ follows from using this inequality in (2.1) and the assumption $\gamma L_F \in (0,1)$.

2.2.(ii): Suppose that $\mathcal{V}(z,\bar{z},z^+) = 0$. Then Proposition 2.2.(i) and Proposition A.2.(ii) imply that $z = \bar{z} = z^+ \in \operatorname{zer}(F + \partial g)$. Conversely, suppose that $z = \bar{z} = z^+ \in \operatorname{zer}(F + \partial g)$. Then it is clear from (2.1) that $\mathcal{V}(z,\bar{z},z^+) = 0$. □

The following result shows that $\mathcal{V}$ is, in fact, a suitable Lyapunov function for the extragradient method (1.2), i.e., it fulfills a descent inequality. Moreover, the descent inequality neither contains a solution of (1.1) nor assumes the existence of a solution.

Theorem 2.3

Suppose that Assumption 2.1 holds and the sequence $((z^k,\bar{z}^k))_{k\in\mathbb{N}_0}$ is generated by (1.2) with initial point $z^0 \in \mathcal{H}$ and step-size parameter $\gamma \in \mathbb{R}_{++}$. Then

$$
\mathcal{V}_{k+1} \leq \mathcal{V}_k - (1 - \gamma^2 L_F^2)\gamma^{-2}\|z^{k+1} - \bar{z}^k\|^2 \tag{2.3}
$$

for each $k \in \mathbb{N}_0$, where $\mathcal{V}_k = \mathcal{V}(z^k,\bar{z}^k,z^{k+1})$ for the Lyapunov function $\mathcal{V}$ defined in (2.1).

Proof. Note that the first and second proximal steps in (1.2) can equivalently be written via their subgradient characterization as

$$
\gamma^{-1}(z^k - \bar{z}^k) - F(z^k) \in \partial g(\bar{z}^k) \quad \text{and} \quad \gamma^{-1}(z^k - z^{k+1}) - F(\bar{z}^k) \in \partial g(z^{k+1}), \tag{2.4}
$$

respectively. Using the subgradient inequality at the points z^{k+1}, z^{k+2} and $\bar{z}^{k+1}$, with the particular subgradients given in (2.4), it follows that

$$
\begin{aligned}
0 &\geq g(z^{k+1}) - g(z) + \langle \gamma^{-1}(z^k - z^{k+1}) - F(\bar{z}^k), z - z^{k+1}\rangle, && \text{(2.5a)}\\
0 &\geq g(z^{k+2}) - g(z) + \langle \gamma^{-1}(z^{k+1} - z^{k+2}) - F(\bar{z}^{k+1}), z - z^{k+2}\rangle, && \text{(2.5b)}\\
0 &\geq g(\bar{z}^{k+1}) - g(z) + \langle \gamma^{-1}(z^{k+1} - \bar{z}^{k+1}) - F(z^{k+1}), z - \bar{z}^{k+1}\rangle. && \text{(2.5c)}
\end{aligned}
$$

holds for any $z \in \mathcal{H}$, respectively. Picking $z = \bar{z}^{k+1}$ in (2.5a), $z = z^{k+1}$ in (2.5b), $z = z^{k+2}$ in (2.5c), summing the resulting inequalities, and multiplying by $2\gamma^{-1}$ gives

$$
\begin{aligned}
0 \geq\ & 2\gamma^{-2}\langle z^k - z^{k+1}, \bar{z}^{k+1} - z^{k+1}\rangle - 2\gamma^{-1}\langle F(\bar{z}^k), \bar{z}^{k+1} - z^{k+1}\rangle \\
& + 2\gamma^{-2}\|z^{k+1} - z^{k+2}\|^2 - 2\gamma^{-1}\langle F(\bar{z}^{k+1}), z^{k+1} - z^{k+2}\rangle \\
& + 2\gamma^{-2}\langle z^{k+1} - \bar{z}^{k+1}, z^{k+2} - \bar{z}^{k+1}\rangle - 2\gamma^{-1}\langle F(z^{k+1}), z^{k+2} - \bar{z}^{k+1}\rangle.
\end{aligned} \tag{2.6}
$$

The first two inner products in (2.6) can be simplified as

$$
A_k = 2\gamma^{-2}\langle z^k - z^{k+1}, \bar{z}^{k+1} - z^{k+1}\rangle - 2\gamma^{-1}\langle F(\bar{z}^k), \bar{z}^{k+1} - z^{k+1}\rangle
$$

$$= \gamma^{-2}\|z^k - z^{k+1}\|^2 + \gamma^{-2}\|\bar{z}^{k+1} - z^{k+1}\|^2 - \gamma^{-2}\|z^k - \bar{z}^{k+1}\|^2 - 2\gamma^{-1}\langle F(\bar{z}^k), \bar{z}^{k+1} - z^{k+1}\rangle,$$

where the identity $2\langle x, y\rangle = \|x\|^2 + \|y\|^2 - \|x-y\|^2$ for each $x, y \in \mathcal{H}$ is used in the second equality, while the remaining four terms in (2.6) can be simplified as

$$\begin{aligned}
B_k &= 2\gamma^{-2}\|z^{k+1} - z^{k+2}\|^2 - 2\gamma^{-1}\langle F(\bar{z}^{k+1}), z^{k+1} - z^{k+2}\rangle \\
&\quad + 2\gamma^{-2}\langle z^{k+1} - \bar{z}^{k+1}, z^{k+2} - \bar{z}^{k+1}\rangle - 2\gamma^{-1}\langle F(z^{k+1}), z^{k+2} - \bar{z}^{k+1}\rangle \\
&= 2\gamma^{-1}\langle z^{k+1} - z^{k+2}, F(z^{k+1}) - F(\bar{z}^{k+1})\rangle + \gamma^{-2}\|z^{k+2} - \bar{z}^{k+1}\|^2 + \gamma^{-2}\|z^{k+1} - z^{k+2}\|^2 \\
&\quad + \gamma^{-2}\|z^{k+1} - \bar{z}^{k+1}\|^2 - 2\gamma^{-1}\langle F(z^{k+1}), z^{k+1} - \bar{z}^{k+1}\rangle \\
&= \mathcal{V}_{k+1} + \gamma^{-2}\|z^{k+1} - \bar{z}^{k+1}\|^2 - 2\gamma^{-1}\langle F(z^{k+1}), z^{k+1} - \bar{z}^{k+1}\rangle,
\end{aligned}$$

where the identity $2\langle x, y\rangle = \|x\|^2 + \|y\|^2 - \|x-y\|^2$ for each $x, y \in \mathcal{H}$ is used in the second equality, and the explicit expression for $\mathcal{V}_{k+1} = \mathcal{V}(z^{k+1}, \bar{z}^{k+1}, z^{k+2})$ is used in the third equality. Therefore, the inequality $A_k + B_k \leq 0$ in (2.6) can be rearranged as

$$\begin{aligned}
\mathcal{V}_{k+1} &\leq \gamma^{-2}\|z^k - \bar{z}^{k+1}\|^2 - \gamma^{-2}\|z^k - z^{k+1}\|^2 - 2\gamma^{-2}\|\bar{z}^{k+1} - z^{k+1}\|^2 \\
&\quad + 2\gamma^{-1}\underbrace{\langle F(\bar{z}^k) - F(z^{k+1}), \bar{z}^{k+1} - z^{k+1}\rangle}_{=C_k}.
\end{aligned} \tag{2.7}$$

We can upper bound the term C_k as

$$\begin{aligned}
C_k &= \langle F(z^k) - F(z^{k+1}), z^{k+1} - z^k\rangle + \langle F(\bar{z}^k) - F(z^k), z^{k+1} - z^k\rangle \\
&\quad + \langle F(\bar{z}^k) - F(z^{k+1}), \bar{z}^{k+1} + z^k - 2z^{k+1}\rangle \\
&\leq \langle F(\bar{z}^k) - F(z^k), z^{k+1} - z^k\rangle \\
&\quad + \tfrac{\gamma}{2}\|F(\bar{z}^k) - F(z^{k+1})\|^2 + \tfrac{1}{2\gamma}\|\bar{z}^{k+1} - 2z^{k+1} + z^k\|^2 \\
&\leq \langle F(\bar{z}^k) - F(z^k), z^{k+1} - z^k\rangle + \tfrac{\gamma L_F^2}{2}\|\bar{z}^k - z^{k+1}\|^2 \\
&\quad + \tfrac{1}{2\gamma}(2\|z^k - z^{k+1}\|^2 + 2\|\bar{z}^{k+1} - z^{k+1}\|^2 - \|z^k - \bar{z}^{k+1}\|^2)
\end{aligned} \tag{2.8}$$

where monotonicity of F and Young's inequality is used in the first inequality, and Lipschitz continuity of F along with the identity $\|x+y\|^2 + \|x-y\|^2 = 2\|x\|^2 + 2\|y\|^2$ for each $x, y \in \mathcal{H}$ is used in the last inequality.

Combining (2.7) and (2.8), and using $\mathcal{V}_k = 2\gamma^{-1}\langle z^k - z^{k+1}, F(z^k) - F(\bar{z}^k)\rangle + \gamma^{-2}\|z^{k+1} - \bar{z}^k\|^2 + \gamma^{-2}\|z^k - z^{k+1}\|^2$ gives

$$\begin{aligned}
\mathcal{V}_{k+1} - \mathcal{V}_k &\leq 2\gamma^{-1}\langle z^{k+1} - z^k, F(\bar{z}^k) - F(z^k)\rangle + L_F^2\|\bar{z}^k - z^{k+1}\|^2 \\
&\quad + \gamma^{-2}\|z^k - z^{k+1}\|^2 - \mathcal{V}_k \\
&= -(1 - \gamma^2 L_F^2)\gamma^{-2}\|\bar{z}^k - z^{k+1}\|^2,
\end{aligned}$$

as claimed. □

Next, we present Corollary 2.4, which follows immediately from Theorem 2.3 by letting $g = 0$. Observe that Corollary 2.4 recovers known results, e.g., see [20, Lemma 3.2 and Theorem 3.3], [11, Theorem 1], and [10, Remark 2.1].

Corollary 2.4
Suppose that Assumption 2.1.(i) holds and the sequence $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ *is generated by*

$$\begin{aligned}\bar{z}^k &= z^k - \gamma F(z^k),\\ z^{k+1} &= z^k - \gamma F(\bar{z}^k)\end{aligned}$$

for each $k \in \mathbb{N}_0$*, with initial point* $z^0 \in \mathcal{H}$ *and step-size parameter* $\gamma \in \mathbb{R}_{++}$*. Then,*

$$\|F(z^{k+1})\|^2 \leq \|F(z^k)\|^2 - (1-\gamma^2 L_F^2)\|F(z^k) - F(\bar{z}^k)\|^2 \tag{2.10}$$

for each $k \in \mathbb{N}_0$*. Moreover, if* $\gamma \in (0, 1/L_F)$ *and* $\operatorname{zer}(F) \neq \emptyset$*, then*

$$\|F(z^k)\|^2 \in o(1/k) \text{ as } k \to \infty, \tag{2.11}$$

and for any $k \in \mathbb{N}_0$ *and* $z^\star \in \operatorname{zer}(F)$ *it holds that*

$$\|F(z^k)\|^2 \leq \frac{\|z^0 - z^\star\|^2}{\gamma^2(1-\gamma^2 L_F^2)(k+1)}. \tag{2.12}$$

Proof. Letting $g = 0$ in Theorem 2.3 gives (2.10). Using $g = 0$, (A.4) in Proposition A.3 gives

$$\|z^{i+1} - z^\star\|^2 \leq \|z^i - z^\star\|^2 - \gamma^2(1-\gamma^2 L_F^2)\|F(z^i)\|^2. \tag{2.13}$$

for each $i \in \mathbb{N}_0$. Inductively summing (2.13) from $i = 0$ to $i = k$, rearranging, and dividing by $\gamma^2(1-\gamma^2 L_F^2)$ gives that

$$\begin{aligned}\sum_{i=0}^{k}\|F(z^i)\|^2 &\leq \frac{\sum_{i=0}^{k}\left(\|z^i - z^\star\|^2 - \|z^{i+1} - z^\star\|^2\right)}{\gamma^2(1-\gamma^2 L_F^2)}\\ &\leq \frac{\|z^0 - z^\star\|^2}{\gamma^2(1-\gamma^2 L_F^2)}\end{aligned}$$

for each $k \in \mathbb{N}_0$. Now (2.11) and (2.12) follow from the monotonicity of $(\|F(z^i)\|)_{i\in\mathbb{N}_0}$, i.e., (2.10). $\square$

The following result shows that $\mathcal{V}_k$, when scaled by a nonnegative constant, equals the residual of a Fejér-type inequality. As a direct consequence, this gives a $o(1/k)$ last-iterate convergence result in terms of $\mathcal{V}_k$ as presented in Corollary 2.6.

Theorem 2.5
Suppose that Assumption 2.1 holds, the sequence $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ *is generated by (1.2) with initial point* $z^0 \in \mathcal{H}$ *and step-size parameter* $\gamma \in (0, 1/L_F]$*, and the sequence* $(\mathcal{V}_k)_{k\in\mathbb{N}_0}$ *is given by* $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$ *for each* $k \in \mathbb{N}_0$ *and the Lyapunov function* $\mathcal{V}$ *defined in (2.1). Then, for any* $k \in \mathbb{N}_0$ *and* $z^\star \in \operatorname{zer}(F + \partial g)$ *it holds that*

$$\|z^{k+1} - z^\star\|^2 \leq \|z^k - z^\star\|^2 - \alpha(\gamma, L_F)\mathcal{V}_k, \tag{2.14}$$

where

$$\alpha(\gamma, L_F) = \tfrac{\gamma^2}{2}\left(\sqrt{5 - 4\gamma^2 L_F^2} - 1\right) \geq 0. \tag{2.15}$$

Proof. Note that (2.4) and $-F(z^\star) \in \partial g(z^\star)$ can equivalently be characterized by

$$0 \le g(z) - g(\bar{z}^k) - \langle \gamma^{-1}(z^k - \bar{z}^k) - F(z^k), z - \bar{z}^k\rangle, \tag{2.16a}$$
$$0 \le g(z) - g(z^{k+1}) - \langle \gamma^{-1}(z^k - z^{k+1}) - F(\bar{z}^k), z - z^{k+1}\rangle, \tag{2.16b}$$
$$0 \le g(z) - g(z^\star) - \langle -F(z^\star), z - z^\star\rangle \tag{2.16c}$$

for each $z \in \mathcal{H}$. Picking $z = z^{k+1}$ in (2.16a), $z = z^\star$ in (2.16b), $z = \bar{z}^k$ in (2.16c), summing the resulting inequalities, and multiplying by 2γ gives

$$\begin{aligned}
0 \le\ & -2\langle z^k - \bar{z}^k, z^{k+1} - \bar{z}^k\rangle + 2\gamma\langle F(z^k), z^{k+1} - \bar{z}^k\rangle \\
& -2\langle z^k - z^{k+1}, z^\star - z^{k+1}\rangle + 2\gamma\langle F(\bar{z}^k), z^\star - z^{k+1}\rangle \\
& -2\gamma\langle F(\bar{z}^k) - F(z^\star), \bar{z}^k - z^\star\rangle + 2\gamma\langle F(\bar{z}^k), \bar{z}^k - z^\star\rangle \\
\le\ & \|z^k - z^{k+1}\|^2 - \|z^k - \bar{z}^k\|^2 - \|z^{k+1} - \bar{z}^k\|^2 + 2\gamma\langle F(z^k), z^{k+1} - \bar{z}^k\rangle \\
& + \|z^k - z^\star\|^2 - \|z^k - z^{k+1}\|^2 - \|z^\star - z^{k+1}\|^2 + 2\gamma\langle F(\bar{z}^k), z^\star - z^{k+1}\rangle \\
& + 2\gamma\langle F(\bar{z}^k), \bar{z}^k - z^\star\rangle \\
=\ & \|z^k - z^\star\|^2 - \|z^\star - z^{k+1}\|^2 - \|z^k - \bar{z}^k\|^2 - \|z^{k+1} - \bar{z}^k\|^2 \\
& - 2\gamma\langle F(z^k) - F(\bar{z}^k), \bar{z}^k - z^{k+1}\rangle,
\end{aligned} \tag{2.17}$$

where the identity $-2\langle x, y\rangle = \|x - y\|^2 - \|x\|^2 - \|y\|^2$ for each $x, y \in \mathcal{H}$ and monotonicity of F is used in the second inequality. Picking $z = z^{k+1}$ in (2.16a), $z = \bar{z}^k$ in (2.16b), and summing the resulting inequalities gives

$$\begin{aligned}
0 \le\ & g(z^{k+1}) - g(\bar{z}^k) - \langle \gamma^{-1}(z^k - \bar{z}^k) - F(z^k), z^{k+1} - \bar{z}^k\rangle \\
& + g(\bar{z}^k) - g(z^{k+1}) - \langle \gamma^{-1}(z^k - z^{k+1}) - F(\bar{z}^k), \bar{z}^k - z^{k+1}\rangle \\
=\ & -\langle F(z^k) - F(\bar{z}^k), \bar{z}^k - z^{k+1}\rangle - \gamma^{-1}\|\bar{z}^k - z^{k+1}\|^2.
\end{aligned} \tag{2.18}$$

For notational simplicity, we let $\alpha = \alpha(\gamma, L_F)$ for $\alpha(\gamma, L_F)$ as in (2.15), where simple algebra shows that $\alpha \ge 0$ if and only if $\gamma L_F \le 1$. Multiplying (2.18) with $2\alpha\gamma^{-1}$, and adding the result to (2.17) gives

$$\begin{aligned}
0 \le\ & \|z^k - z^\star\|^2 - \|z^{k+1} - z^\star\|^2 - \|z^k - \bar{z}^k\|^2 - (1 + 2\alpha\gamma^{-2})\|z^{k+1} - \bar{z}^k\|^2 \\
& - 2\gamma(1 + \alpha\gamma^{-2})\langle F(z^k) - F(\bar{z}^k), \bar{z}^k - z^{k+1}\rangle \\
=\ & \|z^k - z^\star\|^2 - \|z^{k+1} - z^\star\|^2 - \alpha\mathcal{V}_k + A_k,
\end{aligned}$$

where

$$\begin{aligned}
A_k =\ & -\|z^k - \bar{z}^k\|^2 - (1 + 2\alpha\gamma^{-2})\|z^{k+1} - \bar{z}^k\|^2 - 2\gamma(1 + \alpha\gamma^{-2})\langle F(z^k) - F(\bar{z}^k), \bar{z}^k - z^{k+1}\rangle \\
& + \alpha\mathcal{V}_k \\
=\ & -\|z^k - \bar{z}^k\|^2 - (1 + \alpha\gamma^{-2})\|z^{k+1} - \bar{z}^k\|^2 + \alpha\gamma^{-2}\|z^k - z^{k+1}\|^2 \\
& + 2\gamma\underbrace{\langle \alpha\gamma^{-2}(z^k - z^{k+1}) - (1 + \alpha\gamma^{-2})(\bar{z}^k - z^{k+1}), F(z^k) - F(\bar{z}^k)\rangle}_{=B_k},
\end{aligned} \tag{2.19}$$

where we substituted $\mathcal{V}_k$.

To complete the proof, it is enough to show that $A_k \leq 0$. The last inner product in (2.19) can be upper bounded using Young's inequality as

$$\begin{aligned}B_k &\leq \tfrac{\gamma}{2}\|F(z^k) - F(\bar{z}^k)\|^2 + \tfrac{1}{2\gamma}\|\alpha\gamma^{-2}(z^k - z^{k+1}) - (1+\alpha\gamma^{-2})(\bar{z}^k - z^{k+1})\|^2 \\ &\leq \tfrac{\gamma L_F^2}{2}\|z^k - \bar{z}^k\|^2 \\ &\quad + \tfrac{1}{2\gamma}((1+\alpha\gamma^{-2})\|\bar{z}^k - z^{k+1}\|^2 + \alpha\gamma^{-2}(1+\alpha\gamma^{-2})\|z^k - \bar{z}^k\|^2 - \alpha\gamma^{-2}\|z^k - z^{k+1}\|^2),\end{aligned} \tag{2.20}$$

where Lipschitz continuity of F and the identity $\|\beta x - (1+\beta)y\|^2 = (1+\beta)\|y\|^2 + \beta(1+\beta)\|x-y\|^2 - \beta\|x\|^2$ for each $x, y \in \mathcal{H}$ and each $\beta \in \mathbb{R}$ [6, Corollary 2.15] are used in the second inequality. Substituting (2.20) in (2.19) gives

$$A_k \leq \; -(1 - \gamma^2 L_F^2 - \alpha\gamma^{-2}(1+\alpha\gamma^{-2}))\|z^k - \bar{z}^k\|^2 = 0, \tag{2.21}$$

where the last equality follows from simple algebra after substituting $\alpha = \alpha(\gamma, L_F)$ as in (2.15). $\square$

Corollary 2.6
Suppose that Assumption 2.1 and $\operatorname{zer}(F + \partial g) \neq \emptyset$ *hold, and the sequence* $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ *is generated by (1.2) with initial point* $z^0 \in \mathcal{H}$ *and step-size parameter* $\gamma \in (0, 1/L_F)$. *Then,*

$$\mathcal{V}_k \in o(1/k) \text{ as } k \to \infty, \tag{2.22}$$

and for any $k \in \mathbb{N}_0$ *and* $z^\star \in \operatorname{zer}(F + \partial g)$ *it holds that*

$$\mathcal{V}_k \leq \frac{\|z^0 - z^\star\|^2}{\alpha(\gamma, L_F)(k+1)}, \tag{2.23}$$

where $\alpha(\gamma, L_F) > 0$ *is defined in (2.15) and* $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$ *for the Lyapunov function* $\mathcal{V}$ *defined in (2.1).*

Proof. The last iterate-convergence results in (2.22) and (2.23) follows by inductively summing (2.14), rearranging, dividing by $\alpha(\gamma, L_F)$, and using monotonicity of $(\mathcal{V}_k)_{k\in\mathbb{N}_0}$ as shown in (2.3). $\square$

Remark 2.7
A close review of the proofs of Theorems 2.3 and 2.5 (and therefore also Corollary 2.6) shows that the arguments remain valid when ∂g *in (1.1) is replaced with a maximally monotone and 3-cyclically monotone operator* $T : \mathcal{H} \to \mathcal{H}$ *(see [6, Definition 22.13]) and the proximal operators* $\operatorname{prox}_{\gamma g}$ *in (1.2) with the resolvent* $(\mathrm{Id} + \gamma T)^{-1}$*; the only change is to use cyclic-monotonicity inequalities in place of the subgradient inequalities. We nonetheless choose to present the slightly more restrictive subdifferential-based formulation, as it avoids the added abstraction of cyclic monotonicity and is likely more immediately accessible to a broader audience.*

3. Algorithms for monotone inclusions

In light of the descent property established in Theorem 2.3, we propose line-search extensions of the extragradient method that combine the nominal steps of (1.2) with user-specified directions. This section focuses on identifying appropriate conditions to guarantee global convergence, with detailed proofs deferred to Section 4. We deliberately leave the choice of directions open at this stage and postpone the superlinear convergence analysis to Section 5. This abstraction offers flexibility in choosing methods—such as (inexact) (quasi-)Newton approaches, Anderson acceleration, or other suitable algorithms—for computing the directions.

In the first subsection, we consider the classical extragradient setting where $g = 0$ and introduce a line search based on $\|F(z^k)\|^2$ and its descent inequality in (2.10). We then extend our approach in Section 3.2 to the more general setting of (1.1). Separating the analysis in this way reflects the stronger convergence results available when $g = 0$, as well as the fact that, in this case, the line search is more computationally efficient.

3.1 Fast line-search extragradient

In this subsection, we focus on the case $g = 0$ in (1.1), where the Lyapunov inequality (2.3) simplifies to (2.10). The first algorithm introduced here is `FLEX` (Algorithm 1), which can be viewed as a hybrid scheme in the same spirit as [21, Algorithm 5.16]. At each iteration, it computes a suitable direction d^k (see Section 5) and performs the updates $z^{k+1} = z^k + d^k$ whenever the contraction condition in Step 4 holds. Otherwise, it conducts a line search based on the descent inequality (2.10), serving as a *performance safeguard*.

Before we present the convergence results for `FLEX`, we offer some observations on the line-search procedure.

Remark 3.1

The line-search interpolation strategy in `FLEX` *is designed to ensure global convergence while infusing local update directions in the algorithm. It differs from standard line-search procedures in some respects.*

(i) *After a finite number of backtracks, the method defaults to $\tau_k = 0$, at which point (3.1) is satisfied due to (2.10) in Corollary 2.4. Taking the nominal step after a finite number of trials is not just a practical consideration but is also theoretically grounded. Without additional assumptions, it is possible that $\|F(z^k) - F(\bar{z}^k)\| = 0$ even when no solution has been found, and (3.1) is not satisfied by any $\tau_k > 0$ for some ill-chosen user-specified direction d^k. Therefore, additional assumptions are required for such edge cases if an infinite backtracking strategy with known finite termination is to be employed. This is further explored in Section 3.1.1.*

(ii) *Enforcing a descent inequality as in (3.1) of Step 7 can be viewed as a performance safeguarding. As it is shown in Theorem 3.2.(i) below, the convergence of* `FLEX` *can be guaranteed provided that $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$. Therefore, the descent inequality within the line-search procedure ensures that the directions contribute effectively to the convergence, preventing arbitrarily poor performance.*

The next theorem establishes global convergence of `FLEX` under the assumption that the directions are summable, a setting that will be revisited in Section 5, see also Theorem 5.6.(iii). Alternatively, if F is uniformly monotone, the summability assumption is dropped, as shown in Theorem 3.2.(ii). Moreover, when F is μ_F-strongly monotone, as in Theorem 3.2.(iii), a linear convergence rate is achieved.

Theorem 3.2

Suppose that Assumption 2.1.(i) holds, $\operatorname{zer}(F) \neq \emptyset$, *and the sequence $(z^k)_{k\in\mathbb{N}_0}$ is generated by* `FLEX` *(Algorithm 1). Then, the following hold.*

(i) If $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$, then $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to some point in $\operatorname{zer}(F)$.

Algorithm 1: FLEX (Fast Line-search EXtragradient)

Initialize: $z^0 \in \mathcal{H}$, $\gamma \in (0, 1/L_F)$, $(\rho, \sigma, \beta) \in (0,1)^3$, $M \in \mathbb{N}_0$

for $k = 0, 1, 2, \ldots$ **do**
  $\bar{z}^k = z^k - \gamma F(z^k)$; $w^k = z^k - \gamma F(\bar{z}^k)$
  Compute a direction $d^k \in \mathcal{H}$ at z^k
  if $\|F(z^k + d^k)\| \leq \rho \|F(z^k)\|$ **then**
    $z^{k+1} = z^k + d^k$
  else
    Set $z^{k+1} = (1 - \tau_k) w^k + \tau_k (z^k + d^k)$ where τ_k is the largest number in $\{\beta^i \mid i \in [\![1, M]\!]\} \cup \{0\}$ such that

$$\|F(z^{k+1})\|^2 \leq \|F(z^k)\|^2 - \sigma(1 - \gamma^2 L_F^2)\|F(z^k) - F(\bar{z}^k)\|^2 \tag{3.1}$$

  end if
end for

(ii) If F is uniformly monotone, then $(z^k)_{k \in \mathbb{N}_0}$ converges weakly to some point in $\operatorname{zer}(F)$*.*

(iii) If there exists $0 < \mu_F \leq L_F$ such that $\mu_F \|x - y\| \leq \|F(x) - F(y)\|$ for each $x, y \in \mathcal{H}$, then $(z^k)_{k \in \mathbb{N}_0}$ converges strongly to some point in $\operatorname{zer}(F)$ *and*

$$\|F(z^{k+1})\|^2 \leq \underbrace{\max\left(\rho^2, 1 - \sigma\gamma^2\mu_F^2(1 - \gamma^2 L_F^2)\right)}_{\in(0,1)}\|F(z^k)\|^2 \tag{3.2}$$

for each $k \in \mathbb{N}_0$.

3.1.1 Variant of FLEX under injectivity

As highlighted in Remark 3.1(i), since $\|F(z^k) - F(\bar{z}^k)\| = 0$ can occur without reaching a solution, special considerations are necessary. In FLEX, this is addressed by employing an explicit finite termination in the line-search procedure and assuming that the directions d^k are summable. However, when the operator F is injective, it is possible to exploit the Lyapunov inequality in (2.10) directly to establish convergence results without additional assumptions. To this end, we introduce I-FLEX, which incorporates a more traditional line-search procedure similar to that used in [40, Algorithm PANOC]. However, the PANOC algorithm is developed for minimization problems and utilizes a fundamentally different Lyapunov function. Importantly, I-FLEX uses an infinite backtracking strategy with guaranteed finite termination since injectivity ensures that $\|F(z^k) - F(\bar{z}^k)\| = 0$ only when a solution has been found. Moreover, I-FLEX has two fewer parameters than FLEX, simplifying its implementation.

Proposition 3.3
Suppose that Assumption 2.1.(i) holds and F is injective. Then, independent of the choice of the direction d^k in Step 3 of I-FLEX *(Algorithm 2), either there exists an iteration $k \in \mathbb{N}_0$ such that $z^k \in$* $\operatorname{zer}(F)$ *or the line search in Step 4 is well-defined for each iteration $k \in \mathbb{N}_0$.*

Proof. Follows from $\sigma \in (0,1)$, (2.10), continuity of F, and that $\|F(z^k) - F(\bar{z}^k)\| \neq 0$ if and only if $z^k \notin \operatorname{zer}(F)$. □

Theorem 3.4
Suppose that Assumption 2.1.(i) holds, $\operatorname{zer}(F) \neq \emptyset$*, and the sequence $(z^k)_{k \in \mathbb{N}_0}$ is generated by* I-FLEX *(Algorithm 2).*

Algorithm 2: I-FLEX (Injective-FLEX)

Initialize: $z^0 \in \mathcal{H}$, $\gamma \in (0, 1/L_F)$, $(\sigma, \beta) \in (0,1)^2$
for $k = 0, 1, 2, \dots$ **do**
  $\bar{z}^k = z^k - \gamma F(z^k)$; $w^k = z^k - \gamma F(\bar{z}^k)$
  Compute a direction $d^k \in \mathcal{H}$ at z^k
  Set $z^{k+1} = (1-\tau_k)w^k + \tau_k(z^k + d^k)$ where τ_k is the largest number in $\{\beta^i \mid i \in \mathbb{N}_0\}$ such that

$$\|F(z^{k+1})\|^2 \le \|F(z^k)\|^2 - \sigma(1-\gamma^2 L_F^2)\|F(z^k) - F(\bar{z}^k)\|^2 \tag{3.3}$$

end for

(i) If F is injective and weakly continuous, then each weak sequential cluster point of $(z^k)_{k\in\mathbb{N}_0}$ is in zer (F).

(ii) If F is injective and weakly continuous, and $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$, then $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to some point in zer (F).

(iii) If F is uniformly monotone, then $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to some point in zer (F).

(iv) If there exists $0 < \mu_F \le L_F$ such that $\mu_F\|x-y\| \le \|F(x) - F(y)\|$ for each $x, y \in \mathcal{H}$, then $(z^k)_{k\in\mathbb{N}_0}$ converges strongly to some point in zer (F) *and*

$$\|F(z^{k+1})\|^2 \le \underbrace{(1-\sigma\gamma^2\mu_F^2(1-\gamma^2 L_F^2))}_{\in(0,1)}\|F(z^k)\|^2 \tag{3.4}$$

for each $k \in \mathbb{N}_0$.

3.2 Proximal fast line-search extragradient

A direct generalization of FLEX (Algorithm 1) in Section 3.1 is provided in Prox-FLEX (Algorithm 3) for the case when g in (1.1) is nonzero. Here, the Lyapunov inequality (2.3) from Theorem 2.3 is used to modify the standard extragradient method in (1.2); otherwise, the underlying approach remains the same. However, there is one important difference between FLEX and Prox-FLEX in terms of computations required per line-search trial. The condition (3.1) in FLEX requires only one additional F evaluation per trial while condition (3.5) in Prox-FLEX requires two additional F evaluations and two additional $\mathrm{prox}_{\gamma g}$ evaluations per trial. Next, we present a convergence result of Prox-FLEX.

Theorem 3.5
Suppose that Assumption 2.1 holds, zer $(F + \partial g) \neq \emptyset$, *the sequence $(z^k)_{k\in\mathbb{N}_0}$ is generated by* Prox-FLEX *(Algorithm 3), and $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$. Then $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to some point in* zer $(F + \partial g)$.

Remark 3.6
(i) *If F is μ_F-strongly monotone, then the Lyapunov inequality* (2.3) *in Theorem 2.3 can be strengthened to include the additional term $-2\gamma^{-1}\mu_F\|z^{k+1} - z^k\|^2$ in the right-hand side; this follows from using strong monotonicity of F instead of monotonicity of F in the first inequality in* (2.8). *This observation suggests that the line-search*

Algorithm 3: `Prox-FLEX` (Proximal-`FLEX`)

Initialize: $z^0 \in \mathcal{H}$, $\gamma \in (0, 1/L_F)$, $(\rho, \sigma, \beta) \in (0,1)^3$, $M \in \mathbb{N}_0$
Require: Lyapunov function $\mathcal{V}$ as in (2.1) and algorithmic operators T_1^γ, T_2^γ as in (2.2)

1: **for** $k = 0, 1, 2, \ldots$ **do**
2: $\quad \bar{z}^k = T_1^\gamma(z^k)$ $\qquad = \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(z^k)\right)$
3: $\quad w^k = T_2^\gamma(z^k)$ $\qquad = \operatorname{prox}_{\gamma g}\left(z^k - \gamma F(\bar{z}^k)\right)$
4: $\quad$ Compute a direction $d^k \in \mathcal{H}$ at z^k
5: $\quad$ **if** $\mathcal{V}(z^k + d^k, T_1^\gamma(z^k + d^k), T_2^\gamma(z^k + d^k)) \le \rho^2 \mathcal{V}(z^k, \bar{z}^k, w^k)$ **then**
6: $\qquad z^{k+1} = z^k + d^k$
7: $\quad$ **else**
8: $\qquad$ Set $z^{k+1} = (1 - \tau_k) w^k + \tau_k (z^k + d^k)$ where τ_k is the largest number in $\{\beta^i \mid i \in [\![1, M]\!]\} \cup \{0\}$ such that

$$\begin{aligned} &\mathcal{V}(z^{k+1}, T_1^\gamma(z^{k+1}), T_2^\gamma(z^{k+1})) \\ &\le \mathcal{V}(z^k, \bar{z}^k, w^k) - \sigma(1 - \gamma^2 L_F^2)\gamma^{-2}\|w^k - \bar{z}^k\|^2 \end{aligned} \tag{3.5}$$

9: $\quad$ **end if**
10: **end for**

condition (3.5) in `Prox-FLEX` can be replaced by

$$\begin{aligned} \mathcal{V}(z^{k+1}, \bar{z}^{k+1}, w^{k+1}) \le \mathcal{V}(z^k, \bar{z}^k, w^k) &- \sigma(1 - \gamma^2 L_F^2)\gamma^{-2}\|w^k - \bar{z}^k\|^2 \\ &- 2\gamma^{-1}\sigma\mu_F\|w^k - z^k\|^2. \end{aligned} \tag{3.6}$$

Note that using Young's inequality, we get

$$\mathcal{V}(z^k, \bar{z}^k, w^k) \le (2\gamma^{-1} + \gamma^{-1}L_F + \gamma^{-2})\|w^k - z^k\|^2 + (\gamma^{-1}L_F + \gamma^{-2})\|w^k - \bar{z}^k\|^2. \tag{3.7}$$

Combining (3.6), (3.7) and Step 5 in `Prox-FLEX` gives

$$\mathcal{V}(z^{k+1}, \bar{z}^{k+1}, w^{k+1}) \le \underbrace{\max\left(\rho^2, 1 - \frac{\sigma \min\left((1 - \gamma^2 L_F^2), 2\gamma\mu_F\right)}{2\gamma + \gamma L_F + 1}\right)}_{\in(0,1)} \mathcal{V}(z^k, \bar{z}^k, w^k)$$

for each $k \in \mathbb{N}_0$, i.e. $(\mathcal{V}(z^k, \bar{z}^k, w^k))_{k\in\mathbb{N}_0}$ converges at least Q-linearly to zero. However, the resulting line-search condition is not always actionable since μ_F may not be known in many practical problems. Therefore, we have chosen not to consider the strongly monotone case further.

(ii) *Similar to `I-FLEX`, `Prox-FLEX` can be modified to perform infinite backtracking on (3.5) with guaranteed finite termination, even without the strengthened line-search condition described above in Remark 3.6(i). This modification requires $\|w^k - \bar{z}^k\|$ to be an optimality measure, which holds when both F and $\operatorname{prox}_{\gamma g}$ are injective. However, since $\operatorname{prox}_{\gamma g}$ is rarely injective in practical applications, we omit this modification from our analysis.*

4. Global convergence

This section provides detailed proofs of the results presented in Section 3. We start by providing two useful lemmas. The first lemma establishes that the iterates generated by `FLEX`, `I-FLEX`, and `Prox-FLEX` are quasi-Fejér monotone with respect to the solution set, which is an important tool in establishing global convergence. The second lemma contains some auxiliary results.

Lemma 4.1

Suppose that Assumption 2.1 holds, $z^\star \in \operatorname{zer}(F + \partial g)$, $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$, T_1^γ and T_2^γ are the algorithmic operators defined in (2.2), and $\mathcal{V}$ is the Lyapunov function given in (2.1). Let $(z^k)_{k\in\mathbb{N}_0} \in \mathcal{H}^{\mathbb{N}_0}$ such that $z^{k+1} = (1-\tau_k)w^k + \tau_k(z^k + d^k)$, where $\tau_k \in [0,1]$ and $w^k = T_2^\gamma(z^k)$ for each $k \in \mathbb{N}_0$. Then there exists a sequence $(\varepsilon_k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathbb{R}_+)$ such that

$$\|z^{k+1} - z^\star\|^2 \le \|z^k - z^\star\|^2 + \varepsilon_k - (1-\tau_k)\alpha(\gamma, L_F)\mathcal{V}_k, \tag{4.1}$$

where $\alpha(\gamma, L_F)$ is defined in (2.15), $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, w^k)$, and $\bar{z}^k = T_1^\gamma(z^k)$ for each $k \in \mathbb{N}_0$.

Proof. Note that Proposition 2.2.(i) gives that $\mathcal{V}_k \ge 0$ for each $k \in \mathbb{N}_0$. Using the identity

$$\|\tau x + (1-\tau)y\|^2 = \tau\|x\|^2 + (1-\tau)\|y\|^2 - \tau(1-\tau)\|x-y\|^2$$

for each $x, y \in \mathcal{H}$ and $\tau \in \mathbb{R}$ [6, Corollary 2.15], and $z^{k+1} - z^\star = \tau_k(z^k + d^k - z^\star) + (1-\tau_k)(w^k - z^\star)$ for each $k \in \mathbb{N}_0$, we get that

$$\begin{aligned}
&\|z^{k+1} - z^\star\|^2 \\
&\quad= \tau_k\|z^k + d^k - z^\star\|^2 + (1-\tau_k)\|w^k - z^\star\|^2 - \tau_k(1-\tau_k)\|z^k + d^k - w^k\|^2 \\
&\quad\le \tau_k\|z^k + d^k - z^\star\|^2 + (1-\tau_k)\|z^k - z^\star\|^2 - (1-\tau_k)\alpha(\gamma, L_F)\mathcal{V}_k \\
&\quad\le \tau_k(\|z^k - z^\star\| + \|d^k\|)^2 + (1-\tau_k)\|z^k - z^\star\|^2 - (1-\tau_k)\alpha(\gamma, L_F)\mathcal{V}_k \\
&\quad\le \|z^k - z^\star\|^2 + 2\|z^k - z^\star\|\|d^k\| + \|d^k\|^2 - (1-\tau_k)\alpha(\gamma, L_F)\mathcal{V}_k && (4.2)\\
&\quad\le (\|z^k - z^\star\| + \|d^k\|)^2 && (4.3)
\end{aligned}$$

for each $k \in \mathbb{N}_0$, where (2.14) and $\tau_k(1-\tau_k) \ge 0$ is used in the first inequality, the triangle inequality is used in the second inequality, $\tau_k \le 1$ is used in the third inequality, and $(1-\tau_k)\alpha(\gamma, L_F) \ge 0$ is used in the last inequality. Taking the square root of (4.3) and inductively applying the resulting inequality gives that

$$\|z^k - z^\star\| \le \|z^0 - z^\star\| + \sum_{i=0}^{k-1}\|d^i\| \le \underbrace{\|z^0 - z^\star\| + \sum_{i=0}^{\infty}\|d^i\|}_{=E} < \infty \tag{4.4}$$

for each $k \in \mathbb{N}_0$, where the empty sum is interpreted as zero and E is finite since $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$. Therefore, (4.2) and (4.4) imply that

$$\|z^{k+1} - z^\star\|^2 \le \|z^k - z^\star\|^2 + \underbrace{2E\|d^k\| + \|d^k\|^2}_{=\varepsilon_k} - (1-\tau_k)\alpha(\gamma, L_F)\mathcal{V}_k$$

for each $k \in \mathbb{N}_0$, where summability of $(\varepsilon_k)_{k\in\mathbb{N}_0}$ follows from $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$. ∎

Lemma 4.2
Suppose that Assumption 2.1 holds, the sequences $(z^k)_{k\in\mathbb{N}_0}$, $(\bar{z}^k)_{k\in\mathbb{N}_0}$ and $(w^k)_{k\in\mathbb{N}_0}$ are generated by either FLEX *(Algorithm 1),* I-FLEX *(Algorithm 2), or* Prox-FLEX *(Algorithm 3), and $\mathcal{V}$ is the Lyapunov function given in (2.1). Then, the following hold.*

(i) $(\mathcal{V}(z^k, \bar{z}^k, w^k))_{k\in\mathbb{N}_0}$ *is convergent, which for* FLEX *and* I-FLEX *reduces to* $(F(z^k))_{k\in\mathbb{N}_0}$ *being convergent.*

(ii) $(\|w^k - \bar{z}^k\|^2)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathbb{R}_+)$*, which for* FLEX *and* I-FLEX *can be written as* $(\|F(z^k) - F(\bar{z}^k)\|^2)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathbb{R}_+)$.

(iii) If F is uniformly monotone, then $(\|F(z^k)\|)_{k\in\mathbb{N}_0}$ *converges to zero for* FLEX *and* I-FLEX.

Proof. First, we establish for Prox-FLEX that

$$\begin{aligned}&\mathcal{V}(z^{k+1}, \bar{z}^{k+1}, w^{k+1})\\ &\leq \mathcal{V}(z^k, \bar{z}^k, w^k)\\ &\quad - \min((1-\gamma L_F)(1-\rho^2), \sigma(1-\gamma^2 L_F^2))\gamma^{-2}\|w^k - \bar{z}^k\|^2\end{aligned} \tag{4.5}$$

for each $k \in \mathbb{N}_0$. Note that Step 5 in Prox-FLEX implies that

$$\begin{aligned}\mathcal{V}(z^{k+1}, \bar{z}^{k+1}, w^{k+1}) &\leq \rho^2 \mathcal{V}(z^k, \bar{z}^k, w^k)\\ &= \mathcal{V}(z^k, \bar{z}^k, w^k) - (1-\rho^2)\mathcal{V}(z^k, \bar{z}^k, w^k)\\ &\leq \mathcal{V}(z^k, \bar{z}^k, w^k) - (1-\gamma L_F)(1-\rho^2)\gamma^{-2}\|w^k - \bar{z}^k\|^2\end{aligned}$$

for each iteration k when the condition in Step 5 of Prox-FLEX is true, where Proposition 2.2.(i) is used in the last inequality. This combined with (3.5) in Prox-FLEX gives (4.5).

Second, since Prox-FLEX reduced to FLEX when $g = 0$, (4.5) implies that

$$\begin{aligned}&\|F(z^{k+1})\|^2\\ &\leq \|F(z^k)\|^2\\ &\quad - \min((1-\gamma L_F)(1-\rho^2), \sigma(1-\gamma^2 L_F^2))\|F(z^k) - F(\bar{z}^k)\|^2\end{aligned} \tag{4.6}$$

for each $k \in \mathbb{N}_0$, for FLEX.

4.2.(i): Follows from (4.6) for FLEX, (3.3) for I-FLEX, and (4.5) for Prox-FLEX, combined with the monotone convergence theorem.

4.2.(ii): Note that $(\|w^k - \bar{z}^k\|^2)_{k\in\mathbb{N}_0} = (\gamma^2\|F(z^k) - F(\bar{z}^k)\|^2)_{k\in\mathbb{N}_0}$ for FLEX and I-FLEX. The statement follows from (4.6) for FLEX, (3.3) for I-FLEX, and (4.5) for Prox-FLEX, combined with a telescoping summation argument.

4.2.(iii): Suppose that F is uniformly monotone, i.e., there exists an increasing function $\phi : \mathbb{R}_+ \to \mathbb{R}_+$, that vanishes only at 0, such that

$$\phi(\|x - y\|) \leq \langle x - y, F(x) - F(y)\rangle$$

for each $x, y \in \mathcal{H}$. Note that

$$\phi\left(\gamma\|F(z^k)\|\right) = \phi\left(\|z^k - \bar{z}^k\|\right)$$

$$
\begin{aligned}
&\le \langle z^k - \bar{z}^k, F(z^k) - F(\bar{z}^k)\rangle \\
&\le \|z^k - \bar{z}^k\| \|F(z^k) - F(\bar{z}^k)\| \\
&= \gamma \|F(z^k)\| \|F(z^k) - F(\bar{z}^k)\| \xrightarrow[k\to\infty]{} 0,
\end{aligned}
$$

where $\bar{z}^k = z^k - \gamma F(z^k)$ is used in the first equality, the Cauchy–Schwarz inequality is used in the second inequality, $\bar{z}^k = z^k - \gamma F(z^k)$ is used in the last equality, and the convergence to zero in the last line follows from Lemma 4.2.(i) and Lemma 4.2.(ii). This proves the claim. □

4.1 Proofs regarding FLEX

Proof of Theorem 3.2.(i). This follows from Theorem 3.5, since Prox-FLEX (Algorithm 3) reduces to FLEX (Algorithm 1) when $g = 0$. □

Proof of Theorem 3.2.(ii). See Lemma 4.2.(iii). □

Proof of Theorem 3.2.(iii). Note that (3.1) gives that

$$
\begin{aligned}
\|F(z^{k+1})\|^2 \le& \|F(z^k)\|^2 - \sigma(1-\gamma^2 L_F^2)\|F(z^k) - F(\bar{z}^k)\|^2 \\
\le& \|F(z^k)\|^2 - \sigma\mu_F^2(1-\gamma^2 L_F^2)\|z^k - \bar{z}^k\|^2 \\
=& (1 - \sigma\gamma^2\mu_F^2(1-\gamma^2 L_F^2))\|F(z^k)\|^2 \qquad (4.7)
\end{aligned}
$$

for each iteration k such that the condition in Step 4 in FLEX is false, where Step 2 in FLEX is used in the last equality. Combining (4.7) and Step 4 in FLEX gives (3.2). Moreover, $(\|F(z^k)\|)_{k\in\mathbb{N}_0}$ converges to zero since $\max(\rho^2, 1 - \sigma\gamma^2\mu_F^2(1-\gamma^2 L_F^2)) \in (0,1)$. Since

$$
\|z^k - z^\star\| \le \mu_F^{-1}\|F(z^k) - F(z^\star)\| = \mu_F^{-1}\|F(z^k)\|
$$

for each $k \in \mathbb{N}_0$, $(z^k)_{k\in\mathbb{N}_0}$ converges strongly to $z^\star \in \operatorname{zer}(F)$. □

4.2 Proofs regarding I-FLEX

Proof of Theorem 3.4.(i). Suppose that $(z^k)_{k\in K} \rightharpoonup z^\infty$. Weak continuity of F and $\bar{z}^k = z^k - \gamma F(z^k)$ give that $(\bar{z}^k)_{k\in K} \rightharpoonup \bar{z}^\infty = z^\infty - \gamma F(z^\infty)$. On the other hand, it follows from Lemma 4.2.(ii) and weak continuity of F that $F(z^\infty) = F(\bar{z}^\infty)$, which in view of the injectivity assumption of F, implies $z^\infty = \bar{z}^\infty = z^\infty - \gamma F(z^\infty)$. Therefore, $z^\infty \in \operatorname{zer}(F)$, as claimed. □

Proof of Theorem 3.4.(ii). Note that Theorem 3.4.(i) gives that each weak sequential cluster point of $(z^k)_{k\in\mathbb{N}_0}$ is in $\operatorname{zer}(F)$. Moreover, [6, Lemma 5.31] and (4.1) in Lemma 4.1 give that $(\|z^k - z^\star\|)_{k\in\mathbb{N}_0}$ converges. Thus, [6, Lemma 2.47] gives that $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to some point in $\operatorname{zer}(F)$, as claimed. □

Proof of Theorem 3.4.(iii). See Lemma 4.2.(iii). □

Proof of Theorem 3.4.(iv). Note that (3.3) gives that

$$
\|F(z^{k+1})\|^2 \le \|F(z^k)\|^2 - \sigma(1-\gamma^2 L_F^2)\|F(z^k) - F(\bar{z}^k)\|^2
$$

$$\begin{aligned}&\leq \|F(z^k)\|^2 - \sigma\mu_F^2(1-\gamma^2L_F^2)\|z^k - \bar{z}^k\|^2\\&=(1-\sigma\gamma^2\mu_F^2(1-\gamma^2L_F^2))\|F(z^k)\|^2\end{aligned}$$

for each $k \in \mathbb{N}_0$, where Step 2 in `I-FLEX` is used in the last equality. Therefore, $(\|F(z^k)\|)_{k\in\mathbb{N}_0}$ converges to zero since $1-\sigma\gamma^2\mu_F^2(1-\gamma^2L_F^2) \in (0,1)$. Since

$$\|z^k - z^\star\| \leq \mu_F^{-1}\|F(z^k) - F(z^\star)\| = \mu_F^{-1}\|F(z^k)\|$$

for each $k \in \mathbb{N}_0$, $(z^k)_{k\in\mathbb{N}_0}$ converges strongly to $z^\star \in \operatorname{zer}(F)$. □

4.3 Proofs regarding `Prox-FLEX`

Proof of Theorem 3.5. Set $\tau_k = 1$ for the iterations when the condition in Step 5 in `Prox-FLEX` is true and let $z^\star \in \operatorname{zer}(F + \partial g)$. Then (4.1) in Lemma 4.1 and [6, Lemma 5.31] imply that $(\|z^k - z^\star\|)_{k\in\mathbb{N}_0}$ converges. Thus, the proof is complete if we can show that weak sequential cluster points of $(z^k)_{k\in\mathbb{N}_0}$ belong to $\operatorname{zer}(F+\partial g)$, due to [6, Lemma 2.47].

For this, it suffices to show that $(\|\bar{z}^k - z^k\|)_{k\in\mathbb{N}_0}$ converges to zero. Indeed, suppose that $(z^k)_{k\in K} \rightharpoonup z^\infty$ for some $z^\infty \in \mathcal{H}$ and $(\|\bar{z}^k - z^k\|)_{k\in\mathbb{N}_0}$ converges to zero. Then $(\bar{z}^k)_{k\in K} \rightharpoonup z^\infty$. Moreover, the proximal evaluation in Step 2 in `Prox-FLEX` can equivalently be written as

$$\gamma^{-1}(z^k - \bar{z}^k) - F(z^k) + F(\bar{z}^k) \in (F + \partial g)(\bar{z}^k). \tag{4.8}$$

The left-hand side of (4.8) converges strongly to zero since F is continuous and $(\|z^k - \bar{z}^k\|)_{k\in\mathbb{N}_0}$ converges to zero. Moreover, the operator $F+\partial g$ is maximally monotone, since F is maximally monotone (by continuity and monotonicity [6, Corollary 20.28]), ∂g is maximally monotone [6, Theorem 20.48], and F has full domain [6, Corollary 25.5]. Thus, [6, Proposition 20.38] gives that $z^\infty \in \operatorname{zer}(F+\partial g)$, and by [6, Lemma 2.47] we conclude that $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to a point in $\operatorname{zer}(F+\partial g)$.

It remains to show that $(\|\bar{z}^k - z^k\|)_{k\in\mathbb{N}_0}$ converges to zero, which we do by showing that $(\mathcal{V}_k)_{k\in\mathbb{N}_0}$ converges to zero and applying Proposition 2.2.(i). Let $K_{<1} = \{k \in \mathbb{N}_0 \mid \tau_k < 1\}$. Suppose that $|K_{<1}| < \infty$. Then $\mathcal{V}_{k+1} \leq \rho^2\mathcal{V}_k$ for each $k \in \mathbb{N}_0$ such that $k > \max K_{<1}$, and $(\mathcal{V}_k)_{k\in\mathbb{N}_0}$ converges to zero since $\rho \in (0,1)$. On the contrary, suppose that $|K_{<1}| = \infty$. Let $\Gamma : K_{<1} \to K_{<1}$ such that $\Gamma(k) = \min\{i \in K_{<1} \mid k < i\}$ for each $k \in K_{<1}$. Let $k \in K_{<1}$, and notice that $\tau_k \leq \bar{\tau}$ for any such index, where $\bar{\tau} = \max\{\beta^i \mid i \in [\![1, M]\!]\} \cup \{0\} < 1$. Inductively summing (4.1) in Lemma 4.1 from k to $\Gamma(k) - 1$ gives

$$\|z^{\Gamma(k)} - z^\star\|^2 \leq \|z^k - z^\star\|^2 - (1-\bar{\tau})\alpha(\gamma, L_F)\mathcal{V}_k + \sum_{i=k}^{\Gamma(k)-1} \varepsilon_i, \tag{4.9}$$

where we used the fact that $\tau_i = 1$ for any $i \in K_1$. Inductively summing over all $k \in K_{<1}$ in (4.9), rearranging, and dividing by $(1-\bar{\tau})\alpha(\gamma, L_F) > 0$ gives

$$\sum_{k\in K_{<1}} \mathcal{V}_k \leq \frac{\sum_{k\in K_{<1}} (\|z^k - z^\star\|^2 - \|z^{\Gamma(k)} - z^\star\|^2 + \sum_{i=k}^{\Gamma(k)-1}\varepsilon_i)}{(1-\bar{\tau})\alpha(\gamma, L_F)}$$

$$\leq \frac{\|z^{\min(K_{<1})} - z^\star\|^2 + \sum_{k=0}^{\infty} \varepsilon_k}{(1-\bar{\tau})\alpha(\gamma, L_F)} < \infty, \tag{4.10}$$

where summability of $(\varepsilon_k)_{k\in\mathbb{N}_0}$ is used in the last inequality. Note that

$$\begin{aligned}\sum_{k=0}^{\infty} \mathcal{V}_k &= \sum_{k\in K_{<1}} \sum_{i=k}^{\Gamma(k)-1} \mathcal{V}_i \\ &\leq \sum_{k\in K_{<1}} \sum_{i=k}^{\Gamma(k)-1} \rho^{2(i-k)} \mathcal{V}_k \\ &\leq \frac{1}{1-\rho^2} \sum_{k\in K_{<1}} \mathcal{V}_k < \infty,\end{aligned}$$

where Step 5 in `Prox-FLEX` is used in the first inequality, the expression for the geometric series is used in the second inequality, and (4.10) is used in the last inequality. This completes the proof. □

5. Superlinear convergence

The convergence analyses presented so far have been blind to the choice of directions $(d^k)_{k\in\mathbb{N}_0}$; nevertheless, attaining a fast convergence rate relies on their precise choice. This section presents a minimal set of assumptions on the directions that ensure superlinear convergence. Our main focus will be on quasi-Newton-type directions that are computed as

$$d^k = -H_k R_\gamma(z^k), \quad \text{where} \quad R_\gamma = \tfrac{1}{\gamma}(\mathrm{Id} - \mathrm{prox}_{\gamma g} \circ (\mathrm{Id} - \gamma F)), \tag{5.1}$$

$\gamma \in \mathbb{R}_{++}$, $H_k : \mathcal{H} \to \mathcal{H}$ is a linear operator encapsulating information of the geometry of the residual mapping R_γ at z^k, and F and g satisfy Assumption 2.1. The specific way H_k is computed determines the underlying quasi-Newton method (see Section 6 for details). Notice that the zeros of R_γ coincide with the set of solutions of (1.1). Moreover, when $g = 0$, R_γ reduces to F, and the directions are given by $d^k = -H_k F(z^k)$. The following assumption on the directions $(d^k)_{k\in\mathbb{N}_0}$ can be seen as a boundedness assumption on the linear operators $(H_k)_{k\in\mathbb{N}_0}$. However, note that the assumption applies to directions beyond the ones given in (5.1).

Assumption 5.1
The sequence of directions $(d^k)_{k\in\mathbb{N}_0}$ used in `FLEX`*,* `I-FLEX`*, or* `Prox-FLEX` *satisfies $\|d^k\| \leq D\|R_\gamma(z^k)\|$ for each $k \in \mathbb{N}_0$ such that $k \geq K$, for some constants $D \geq 0$ and $K \in \mathbb{N}_0$, where R_γ denotes the residual operator as defined in (5.1) (the function g is set to zero in the particular cases of* `FLEX` *and* `I-FLEX`*).*

Assumption 5.1 is a natural assumption for directions defined in (5.1). For example, under suitable regularity conditions for regularized Newton directions—specifically when $g = 0$—we demonstrate this in Proposition 5.2. Note that in Proposition 5.2, we assume that F is continuously Fréchet differentiable; however, this assumption is made solely for illustrative purposes and is not required elsewhere in the paper. Assumption 5.1 has also been utilized in the context of minimization and in finding zeros of nonexpansive maps, as seen in [1, Theorem 5.7.A3] and [41, Assumption 2], respectively.

Proposition 5.2
Let $F : \mathcal{H} \to \mathcal{H}$ be monotone and continuously Fréchet differentiable, and suppose that the Fréchet derivative $\mathsf{D}F$ at $z^\star \in \operatorname{zer}(F)$ is left invertible. Suppose that $(z^k)_{k\in\mathbb{N}_0}, (d^k)_{k\in\mathbb{N}_0} \in \mathcal{H}^{\mathbb{N}_0}$ are such that

$$(r_k \operatorname{Id} + \mathsf{D}F(z^k))d^k = -F(z^k) \tag{5.2}$$

for some sequence $(r_k)_{k\in\mathbb{N}_0} \in \mathbb{R}_{++}^{\mathbb{N}_0}$, and that $(z^k)_{k\in\mathbb{N}_0}$ converges strongly to $z^\star$. Then, Assumption 5.1 is satisfied with $g = 0$.

Proof. Let $t > 0$ and note that

$$0 \le \frac{\langle F(z+tv) - F(z), z + tv - z\rangle}{t^2} \xrightarrow[t\downarrow 0]{} \langle \mathsf{D}F(z)v, v\rangle \tag{5.3}$$

for any $v \in \mathcal{H}$ and for any $z \in \mathcal{H}$, by monotonicity of F. This implies that the bounded linear operator $r_k \operatorname{Id} + \mathsf{D}F(z^k)$ is r_k-strongly monotone, and therefore invertible for each $k \in \mathbb{N}_0$. This, in turn, ensures that the regularized Newton update (5.2) is well-defined, i.e., d^k is uniquely defined at each iteration.

Moreover, since $\mathsf{D}F(z^\star)$ is left invertible, there exists $c_1 > 0$ such that $\|\mathsf{D}F(z^\star)v\| \ge c_1\|v\|$ for any $v \in \mathcal{H}$ [5, Proposition 10.29]. This observation combined with $z^k \to z^\star$ and continuity of $\mathsf{D}F(\cdot)$ implies that there exists $c_2 > 0$ and $K \in \mathbb{N}_0$ such that $\|\mathsf{D}F(z^k)v\| \ge c_2\|v\|$ for any $v \in \mathcal{H}$ and for any $k \ge K$. Therefore,

$$\begin{aligned}
\|F(z^k)\|^2 &= \|r_k \operatorname{Id} + \mathsf{D}F(z^k)d^k\|^2 \\
&= \|r_k d^k\|^2 + 2r_k\langle \mathsf{D}F(z^k)d^k, d^k\rangle + \|\mathsf{D}F(z^k)d^k\|^2 \\
&\ge \|\mathsf{D}F(z^k)d^k\|^2 \\
&\ge c_2^2\|d^k\|^2
\end{aligned}$$

for each $k \ge K$, where the first inequality follows from (5.3) and $r_k > 0$. This establishes that Assumption 5.1 is satisfied with $D = 1/c_2$, when $g = 0$. □

Remark 5.3
In the case of `FLEX`*, when the operator is strongly monotone, the sequence $(\|F(z^k)\|)_{k\in\mathbb{N}_0}$ converges Q-linearly to zero, as established in Theorem 3.2.(iii). This observation, combined with Assumption 5.1 is sufficient to conclude that $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$, thereby yielding global convergence as demonstrated in Theorem 3.2.(i). An analogous argument extends to* `Prox-FLEX` *after incorporating the strengthening discussed in Remark 3.6(i).*

We proceed to quantify the quality of the directions used in the algorithms that guarantee fast convergence. The classical condition of [17, Chapter 7.5] for Newton-type methods identifies a sequence of directions $(d^k)_{k\in\mathbb{N}_0}$ relative to a sequence $(z^k)_{k\in\mathbb{N}_0}$ converging to $z^\star$ as superlinear if

$$\lim_{k\to\infty} \frac{\|z^k + d^k - z^\star\|}{\|z^k - z^\star\|} = 0. \tag{5.4}$$

This notion a priori assumes the convergence of the sequence $(z^k)_{k\in\mathbb{N}_0}$. Here, we use a slightly refined notion and define superlinear directions similar to [41, Definition VI.2].

Definition 5.4
Suppose that $\gamma \in \mathbb{R}_{++}$, $(z^k)_{k\in\mathbb{N}_0}, (d^k)_{k\in\mathbb{N}_0} \in \mathcal{H}^{\mathbb{N}_0}$, Assumption 2.1 holds, T_1^γ and T_2^γ are the algorithmic operators defined in (2.2), and $\mathcal{V}$ is the Lyapunov function given in (2.1). Then we say that the sequence of directions $(d^k)_{k\in\mathbb{N}_0}$ is superlinear relative to $(z^k)_{k\in\mathbb{N}_0}$ *if*

$$\lim_{k\to\infty} \frac{\mathcal{V}(z^k + d^k, T_1^\gamma(z^k + d^k), T_2^\gamma(z^k + d^k))}{\mathcal{V}(z^k, T_1^\gamma(z^k), T_2^\gamma(z^k))} = 0. \tag{5.5}$$

In the case of solving monotone equations where $g = 0$, addressed by FLEX and I-FLEX, (5.5) reduces to

$$\lim_{k\to\infty} \frac{\|F(z^k + d^k)\|}{\|F(z^k)\|} = 0. \tag{5.6}$$

This condition is closely related to the classical Dennis-Moré assumption [13]. Specifically, when F is strictly differentiable at its zeros, the Dennis-Moré assumption implies (5.6), as shown in [41, Theorem VI.7]. In particular, the condition is satisfied by Broyden's method under mild regularity assumptions at the limit points [41, Theorem VI.8].

Remark 5.5
The superlinear convergence results presented in Theorem 5.6 also hold under (5.4) of [17] since it implies the notion in Definition 5.4. Indeed, by Assumption 5.1

$$\begin{aligned}
\|d^k\|^2 &\le D^2 \left(\|z^k - T_2^\gamma(z^k)\| + \|T_2^\gamma(z^k) - T_1^\gamma(z^k)\|\right)^2 \\
&\le 2D^2 \left(\|z^k - T_2^\gamma(z^k)\|^2 + \|T_2^\gamma(z^k) - T_1^\gamma(z^k)\|^2\right) \\
&\le \frac{2\gamma^2 D^2}{1-\gamma L_F} \mathcal{V}(z^k, T_1^\gamma(z^k), T_2^\gamma(z^k))
\end{aligned}$$

for each $k \in \mathbb{N}_0$ such that $k \ge K$, where the triangle inequality is used in the first inequality and Proposition 2.2.(i) is used in the last inequality. Hence

$$\frac{\mathcal{V}(z^k + d^k, T_1^\gamma(z^k + d^k), T_2^\gamma(z^k + d^k))}{\mathcal{V}(z^k, T_1^\gamma(z^k), T_2^\gamma(z^k))} \le \frac{2\gamma^2 D^2}{(1-\gamma L_F)\alpha(\gamma, L_F)} \frac{\|z^k + d^k - z^\star\|^2}{\|d^k\|}, \tag{5.7}$$

for each $k \in \mathbb{N}_0$ such that $k \ge K$, where $\mathcal{V}(z^k + d^k, T_1^\gamma(z^k + d^k), T_2^\gamma(z^k + d^k)) \le \|z^k + d^k - z^\star\|^2/\alpha(\gamma, L_F)$ is used (see Theorem 2.5). Combining (5.7) with (5.4) and the fact that $\lim_{k\to\infty}\|z^k - z^\star\|/\|d^k\| = 1$ (see [17, Lemmma 7.5.7]) shows that the ratio on the left-hand-side of (5.7) vanishes. Therefore, (5.5) is a weaker condition than (5.4) under Assumption 5.1. We also refer the reader to [17] for further details.

As shown below in Theorem 5.6.(iii), Definition 5.4 in conjunction with Assumption 5.1 is sufficient to conclude $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$, establishing global weak sequential convergence by Theorem 3.5. See also Theorems 3.2.(i) and 3.4.(ii) for FLEX and I-FLEX, respectively.

Theorem 5.6
Suppose that Assumption 2.1 and Assumption 5.1 hold, $\operatorname{zer}(F + \partial g) \ne \emptyset$, *$T_1^\gamma$ and T_2^γ are the algorithmic operators defined in (2.2), $\mathcal{V}$ is the Lyapunov function given in (2.1), and $(d^k)_{k\in\mathbb{N}_0}$ is superlinear relative to the sequence $(z^k)_{k\in\mathbb{N}_0}$ generated by either* FLEX *(Algorithm 1),* I-FLEX *(Algorithm 2), or* Prox-FLEX *(Algorithm 3). Then, the following hold.*

(i) $z^{k+1} = z^k + d^k$ *for all* $k \in \mathbb{N}_0$ *sufficiently large.*

(ii) $(\mathcal{V}(z^k, T_1^\gamma(z^k), T_2^\gamma(z^k)))_{k\in\mathbb{N}_0}$ *converges to zero at least* Q*-superlinearly.*

(iii) $(d^k)_{k\in\mathbb{N}_0} \in \ell^1(\mathbb{N}_0; \mathcal{H})$ *with* $(d^k)_{k\in\mathbb{N}_0}$ *converging to zero at least* R*-superlinearly.*

(iv) If $\dim \mathcal{H} < \infty$*, then* $(z^k)_{k\in\mathbb{N}_0}$ *converges to some point* $z^\star \in \operatorname{zer}(F + \partial g)$ *at least* R*-superlinearly.*

Proof. The proof is presented for `Prox-FLEX`, with the necessary adjustments for `FLEX` and `I-FLEX` outlined at the end of the proof.

5.6.(i) Follows from (5.5) since Step 5 in `Prox-FLEX` is true for all $k \in \mathbb{N}_0$ sufficiently large.

5.6.(ii) Follows from 5.6.(i) and (5.5).

5.6.(iii) Note that Assumption 5.1 and Proposition 2.2.(i) give that

$$\|d^k\| \leq \frac{D}{\gamma}\|z^k - T_1^\gamma(z^k)\| \leq \frac{2D}{\sqrt{1-\gamma L_F}}\sqrt{\mathcal{V}(z^k, T_1^\gamma(z^k), T_2^\gamma(z^k))}$$

for each $k \in \mathbb{N}_0$ such that $k \geq K$. The claim now follows from 5.6.(ii).

5.6.(iv) Theorem 3.5 and 5.6.(iii) imply that the sequence $(z^k)_{k\in\mathbb{N}_0}$ converges to some point $z^\star \in \operatorname{zer}(F + \partial g)$. Since $z^{k+1} - z^k = d^k$ for all $k \in \mathbb{N}_0$ sufficiently large, 5.6.(iii) implies that $(z^{k+1} - z^k)_{k\in\mathbb{N}_0}$ converges to zero at least R-(super)linearly. In particular, there exists $\kappa \in \mathbb{R}_{++}$ and $(c_k)_{k\in\mathbb{N}_0} \in \mathbb{R}_{++}^{\mathbb{N}_0}$ such that $\lim_{k\to\infty} c_k = 0$ and

$$\|z^{k+1} - z^k\| \leq \kappa \prod_{i=1}^{k} c_i \tag{5.8}$$

for each $k \in \mathbb{N}_0$. Let $k, j \in \mathbb{N}_0$ such that $j > k$. The triangle inequality and (5.8) give that

$$\|z^k - z^\star\| \xleftarrow[j\to\infty]{} \|z^k - z^j\| \leq \sum_{\ell=k}^{j-1}\|z^\ell - z^{\ell+1}\| \leq \kappa \sum_{\ell=k}^{j-1}\prod_{i=1}^{\ell} c_i \xrightarrow[j\to\infty]{} \kappa \sum_{\ell=k}^{\infty}\prod_{i=1}^{\ell} c_i = \mu_k.$$

The sequence $(\mu_k)_{k\in\mathbb{N}_0} \in \mathbb{R}_{++}^{\mathbb{N}_0}$ converges to zero at least Q-superlinearly since

$$\frac{\mu_k}{\mu_{k-1}} = \frac{\sum_{\ell=k}^{\infty}\prod_{i=1}^{\ell} c_i}{\sum_{\ell=k-1}^{\infty}\prod_{i=1}^{\ell} c_i} = \frac{\left(\prod_{i=1}^{k-1} c_i\right)\left(\sum_{\ell=k}^{\infty}\prod_{i=k}^{\ell} c_i\right)}{\left(\prod_{i=1}^{k-1} c_i\right)\left(1 + \sum_{\ell=k}^{\infty}\prod_{i=k}^{\ell} c_i\right)} = \frac{\left(\sum_{\ell=k}^{\infty}\prod_{i=k}^{\ell} c_i\right)}{\left(1 + \sum_{\ell=k}^{\infty}\prod_{i=k}^{\ell} c_i\right)} \to 0$$

and $\lim_{k\to\infty} \sum_{\ell=k}^{\infty}\prod_{i=k}^{\ell} c_i = 0$. Thus, $(z^k)_{k\in\mathbb{N}_0}$ converges to $z^\star$ at least R-superlinearly, as claimed.

The assertions for `FLEX` follow directly, as setting $g = 0$ reduces `Prox-FLEX` and its underlying assumptions to those of `FLEX`. For `I-FLEX`, the only distinction is that, for sufficiently large k, $\tau_k = 1$ is always accepted in Step 4 of `I-FLEX`, due to (5.6). All other arguments remain unchanged. □

6. Numerical experiments

In this section, we assess the performance of the proposed algorithms in Section 3 through a series of simulations on standard problems using both synthetic and real-world datasets. Code to replicate the experiments is made available online.[1] Table 1 contains a description of the algorithms used.

Table 1. Algorithms used in the numerical simulations (when applicable).

Method	Description
EG	Extragradient method (1.2) with $\gamma = 0.9/L_F$.
EAG-C	Extra anchored gradient with constant step size $\alpha = 1/(8L_F)$ [47, Section 2.1].
GRAAL	Golden ratio algorithm with $\phi = 2$ and $\alpha = 0.999/L_F$ [2, Algorithm 2], [24].
aGRAAL	Adaptive golden ratio algorithm with $\phi = (2+\sqrt{5})/2$, $\gamma = 1/\phi + 1/\phi^2$ and $\alpha_0 = 0.1$ [2, Algorithm 1], [24].
EG-AA	An extragradient-type method with type-II Anderson acceleration with memory $m = 1$ [35, Algorithm 1] using the parameter values described in [35, Section 4].
FISTA	Fast iterative shrinkage-thresholding algorithm with constant step size [7, Section 4].
FLEX	Algorithm 1 with $\gamma = 0.9/L_F$, $\beta = 0.3$, $\sigma = 0.1$, $\rho = 0.99$, and $M = 2$.
I-FLEX	Algorithm 2 with $\beta = 0.01$ and $\sigma = 0.1$.
Prox-FLEX	Algorithm 3 with $\gamma = 0.9/L_F$, $\beta = 0.3$, $\sigma = 0.1$, $\rho = 0.99$, and $M = 2$.

In the numerical experiments for FLEX, I-FLEX, and Prox-FLEX, we use directions $(d^k)_{k\in\mathbb{N}_0}$ based on quasi-Newton directions.

Anderson acceleration. The first set of quasi-Newton directions we use are the standard limited-memory type-I and type-II Anderson acceleration methods [3, 18]. These directions are computed via (5.1), i.e., $d^k = -H_k R_\gamma(z^k)$, where H_k differs between the type-I and type-II variants. Both methods employ a memory parameter $m \in \mathbb{N}$ and define $m_k = \min\{m, k\}$. They also maintain two buffer matrices:

$$Y_k = \begin{bmatrix} y^{k-m_k} & \cdots & y^{k-1} \end{bmatrix} \quad \text{and} \quad S_k = \begin{bmatrix} s^{k-m_k} & \cdots & s^{k-1} \end{bmatrix},$$

where $y^i = R_\gamma(z^{i+1}) - R_\gamma(z^i)$ and $s^i = z^{i+1} - z^i$. For type-I Anderson acceleration (denoted AA-I), we have

$$H_k = I + (S_k - Y_k)\,(S_k^\top Y_k)^{-1}\, S_k^\top,$$

whereas for type-II Anderson acceleration (denoted AA-II), we have

$$H_k = I + (S_k - Y_k)\,(Y_k^\top Y_k)^{-1}\, Y_k^\top.$$

Additional discussion can be found in [48].

[1] https://github.com/manuupadhyaya/flex

J-symmetric directions. We also incorporate directions derived from the J-symmetric quasi-Newton approach proposed in [4], which is developed for unconstrained minimax problems. This method exploits the so-called J-symmetric structure of the Hessian in such problems, allowing a rank-2 update of the (inverse) Hessian estimate that naturally generalizes the classic Powell's symmetric Broyden method from standard minimization to minimax optimization. The formula for updating H_k in (5.1) can be found in [4, Proposition 2.2]. We refer to this method as `J-sym`.

6.1 Quadratic minimax problem

Consider the quadratic convex-concave minimax problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\ \underset{y\in\mathbb{R}^n}{\text{maximize}}\ \mathcal{L}(x,y) \tag{6.1}$$

for the saddle function $\mathcal{L}:\mathbb{R}^n\times\mathbb{R}^n\to\mathbb{R}$ such that

$$\mathcal{L}(x,y)=\tfrac{1}{2}(x-x^\star)^\top A(x-x^\star)+(x-x^\star)^\top C(y-y^\star)-\tfrac{1}{2}(y-y^\star)^\top B(y-y^\star)$$

for each $(x,y)\in\mathbb{R}^n\times\mathbb{R}^n$, where $x^\star$, $y^\star\in\mathbb{R}^n$, A, $B\in\mathbb{S}^n_+$, and $C\in\mathbb{R}^{n\times n}$. A solution to the minimax problem (6.1) can be obtained by solving an associated saddle point problem, which in turn can equivalently be written as (1.1) by letting $\mathcal{H}=\mathbb{R}^{2n}$ with the inner product set to the dot product, $g=0$, and $F:\mathbb{R}^{2n}\to\mathbb{R}^{2n}$ as the monotone and L_F-Lipschitz continuous operator given by

$$F(z)=\begin{bmatrix}\nabla_x\mathcal{L}(x,y)\\ -\nabla_y\mathcal{L}(x,y)\end{bmatrix}=\begin{bmatrix}A(x-x^\star)+C(y-y^\star)\\ B(y-y^\star)-C^\top(x-x^\star)\end{bmatrix} \tag{6.2}$$

for each $z=(x,y)\in\mathbb{R}^n\times\mathbb{R}^n$, where[2]

$$L_F=\left\|\begin{bmatrix}A & C\\ -C^\top & B\end{bmatrix}\right\|.$$

We generate problem data as in [4, Section 5.1], which is outlined below. The results of the numerical experiments are presented in Figure 1. We see that `FLEX` and `I-FLEX` do very well for small problems, while larger ones are more challenging. Nevertheless, the use of `AA-II` directions in our algorithms systematically performs at the top.

Input: $\omega\in\mathbb{R}_+$ and $n\in\mathbb{N}$
Output: $x^\star$, $y^\star\in\mathbb{R}^n$, A, $B\in\mathbb{S}^n_+$ and $C\in\mathbb{R}^{n\times n}$

1: Let $x^\star$, $y^\star\in\mathbb{R}^n$ such that $x^\star_i$, $y^\star_i\sim\mathcal{N}(0,1)$ for each $i\in[\![1,n]\!]$
2: Let $S\in\mathbb{R}^{n\times n}$ such that $[S]_{i,j}\sim\mathcal{N}(0,1/\sqrt{n})$ for each $i,j\in[\![1,n]\!]$
3: $S\leftarrow(S+S^\top)/2$
4: $S\leftarrow S+(|\lambda_{\min}(S)|+1)I$
5: $A\leftarrow\omega S$
6: Repeat steps 2-4 with a different random seed and let $B\leftarrow\omega S$
7: Repeat step 2 with a different random seed and let $C\leftarrow S$

[2] The matrix norm is taken as the spectral norm.

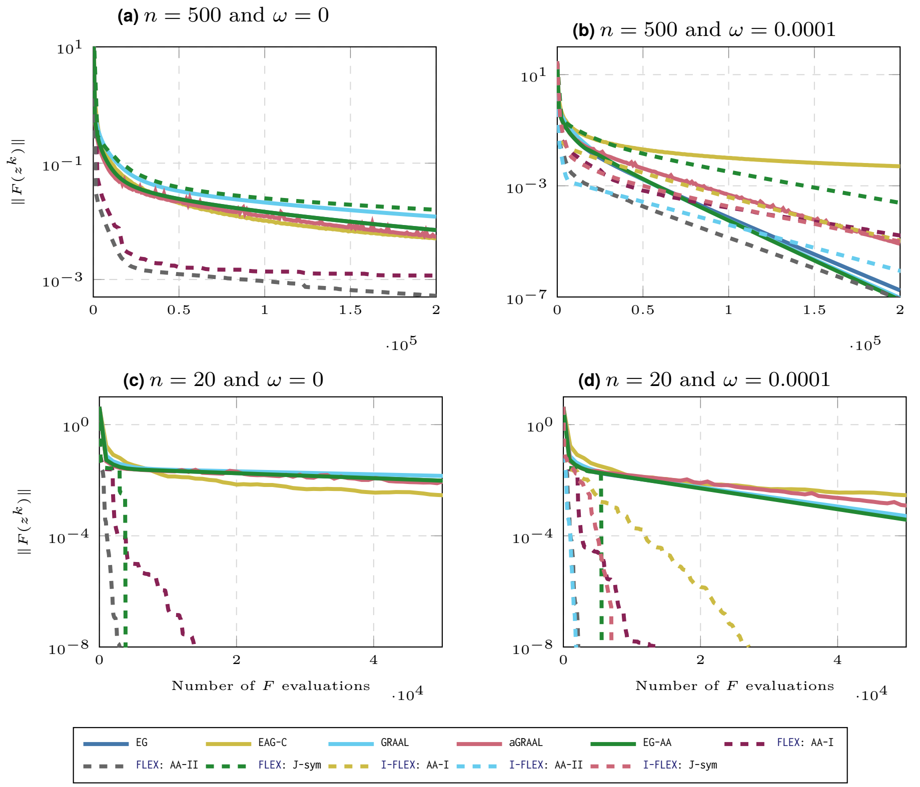


Figure 1. Convergence of algorithms on the quadratic minimax problem (6.1). Both AA-I and AA-II use memory parameter $m = 20$. When $\omega = 0$, the operator F in (6.2) is monotone; for $\omega > 0$, it becomes strongly monotone.

6.2 Bilinear zero-sum game with simplex constraints

Consider the bilinear zero-sum game with simplex constraints given by

$$\underset{x \in \Delta^n}{\text{minimize}} \ \underset{y \in \Delta^n}{\text{maximize}} \ x^\top A y \tag{6.3}$$

where $A \in \mathbb{R}^{n \times n}$ is the payoff matrix and $\Delta^n = \{w \in \mathbb{R}^n_+ \mid w^\top \mathbf{1} = 1\}$ is the probability simplex in $\mathbb{R}^n$, which is equivalent to finding a saddle point $(x^\star, y^\star) \in \Delta^n \times \Delta^n$ (which is guaranteed to exist), i.e.,

$$(x^\star)^\top A y \leq (x^\star)^\top A y^\star \leq x^\top A y^\star$$

for each $(x, y) \in \Delta^n \times \Delta^n$. This, in turn, is equivalent to solving (1.1) by letting $\mathcal{H} = \mathbb{R}^{2n}$ with the inner product set to the dot product, $g = \delta_{\Delta^n \times \Delta^n}$, and $F : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ as the

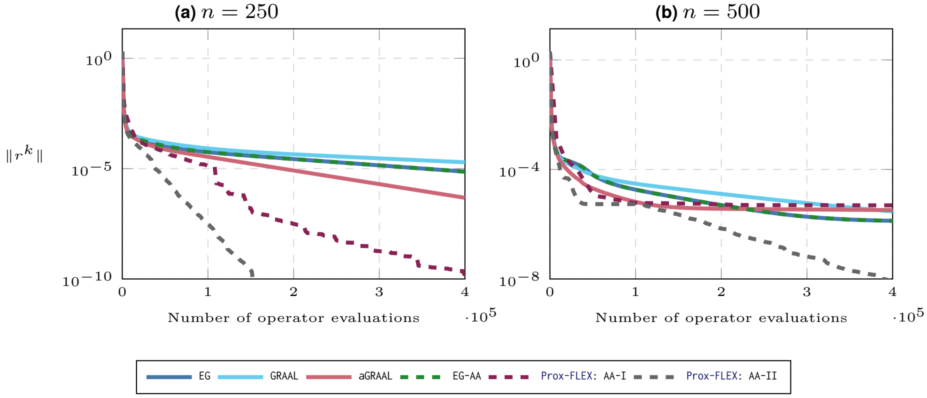


Figure 2. Convergence of algorithms on the bilinear zero-sum game with simplex constraints (6.3) where $r^k = R_{1/2L_F}(z^k)$ and R is the residual mapping in (5.1). Both AA-I and AA-II use memory parameter $m = 10$ for Figure 2a and $m = 20$ for Figure 2b. The number of operator evaluations equals the number of F and $\operatorname{prox}_{\gamma g}$ evaluations.

monotone and L_F-Lipschitz continuous operator given by

$$F(z) = \begin{bmatrix} Ay \\ -A^\top x \end{bmatrix}$$

for each $z = (x, y) \in \mathbb{R}^n \times \mathbb{R}^n$, where

$$L_F = \left\| \begin{bmatrix} 0 & A \\ -A^\top & 0 \end{bmatrix} \right\|.$$

We generate $A = S - S^\top$ for a random matrix $S \in \mathbb{R}^{n\times n}$ such that $[S]_{i,j} \sim \mathcal{N}(0,1)$ for each $i, j \in [\![1, n]\!]$, resulting in a skew-symmetric matrix A. The results of the numerical experiments are presented in Figure 2. We see that using AA-II directions in Prox-FLEX gives good performance.

6.3 Cournot–Nash equilibrium problem

Consider a noncooperative game with $n \in \mathbb{N}$ players, in which each player $i \in [\![1, n]\!]$ has to pick a strategy z_i that lies in $\mathcal{Z}_i$, a subset of a real Hilbert space $\mathcal{H}_i$, and has an associated loss function $\varphi_i : \mathcal{H} \to \mathbb{R}$, where $\mathcal{H} = \prod_{j=1}^n \mathcal{H}_j$. In this case, a pure strategy Nash equilibrium is a strategy profile $z = (z_1, \ldots, z_n) \in \mathcal{H}$ that solves the problem

$$\text{find } z \in \mathcal{H} \text{ such that } z_i \in \operatorname*{Argmin}_{x \in \mathcal{Z}_i} \varphi_i(x; z_{\smallsetminus i}) \text{ for each } i \in [\![1, n]\!], \tag{6.4}$$

where we have used the notation $(x; z_{\smallsetminus i}) = (z_1, \ldots, z_{i-1}, x, z_{i+1}, \ldots, z_n)$ for each $x \in \mathcal{H}_i$ and $i \in [\![1, n]\!]$. In particular, assume that, for each $i \in [\![1, n]\!]$, the function $\varphi_i(\,\cdot\,; z_{\smallsetminus i}) : \mathcal{H}_i \to \mathbb{R}$ is convex for each $z \in \mathcal{H}$, the gradient $\nabla_{z_i}\varphi_i : \mathcal{H} \to \mathcal{H}_i$ exists and is Lipschitz continuous, and the set $\mathcal{Z}_i \subseteq \mathcal{H}_i$ is nonempty, closed and convex. Then (6.4) can equivalently be written as (1.1) by letting $F : \mathcal{H} \to \mathcal{H} : z \mapsto (\nabla_{z_i} \varphi_i(z))_{i=1}^n$ and $g = \delta_{\mathcal{Z}}$, where $\mathcal{Z} = \prod_{i=1}^n \mathcal{Z}_i$, and it is straightforward to verify that Assumption 2.1 holds.

Input: $n \in \mathbb{N}$
Output: $((T_i, a_i, b_i, m_i, d_i))_{i=1}^n$

repeat
  For each $i \in [\![1, n]\!]$, sample m_i uniformly from $[150, 250]$
  For each $i \in [\![1, n]\!]$, sample b_i uniformly from $[30, 50]$
  For each $i \in [\![1, n]\!]$, sample T_i uniformly from $[3, 7]$
  For each $i \in [\![1, n]\!]$, sample d_i uniformly from $[5, 20]$
  Sort $(d_i)_{i=1}^n$ in increasing order
  For each $i \in [\![1, n]\!]$, sample u_i uniformly from $[-10, -5]$
  For each $i \in [\![1, n]\!]$, compute $a_i = d_i/u_i$
  Sort $(a_i)_{i=1}^n$ in decreasing order
  valid $\leftarrow$ `True`
  for $i \in [\![1, n]\!]$ **do**
    if $b_i < -2a_iT_i$ **or** $m_i \leq b_i$ **or** $d_i \leq -a_i$ **then**
      valid $\leftarrow$ `False`
      break
    end if
  end for
until valid is `True`

Let us further specialize the model to the Cournot–Nash equilibrium problem for oligopolistic markets with concave-quadratic cost functions and a differentiated commodity, as presented in [8]. Such models are useful for policymakers and economists in analyzing market outcomes, assessing welfare effects, and evaluating the impact of various market interventions [9, 19, 46, 27, 31, 45]. In particular, in the model of [8], each producer $i \in [\![1, n]\!]$ chooses to produce and supply a quantity $z_i \in [0, T_i]$ of a differentiated commodity at a cost $c_i : \mathbb{R} \to \mathbb{R}$ such that

$$c_i(z_i) = a_i z_i^2 + b_i z_i,$$

for each $z_i \in \mathbb{R}$, where $T_i > 0$ denotes the maximum capacity of production, and $a_i < 0$ and $b_i > 0$ are numbers such that $b_i \geq -2T_i a_i$, ensuring that c_i is increasing on $[0, T_i]$. Moreover, each producer $i \in [\![1, n]\!]$ has a price per produced unit of the differentiated commodity[3], denoted by $p_i : \mathbb{R}^n \to \mathbb{R}$, that also depends on the other producers' supply, and is modeled by

$$p_i(z) = m_i - d_i \sum_{j=1}^n z_j$$

for each $z = (z_1, \ldots, z_n) \in \mathbb{R}^n$, for some $m_i > b_i$ and $d_i > -a_i$, where the last two assumptions guarantee a positive profit in a monopolistic setting, i.e., when $n = 1$. Thus, given that the goal of each producer is to maximize profit, or equivalently minimize losses, an equilibrium state where no producer has any incentive to deviate unidirectionally from its production plan can be modeled by (6.4), with $\mathcal{Z}_i = [0, T_i]$, $\mathcal{H}_i = \mathbb{R}$, and

$$\varphi_i(z) = c_i(z_i) - z_i p(z)$$

[3] Also known as the inverse demand function.

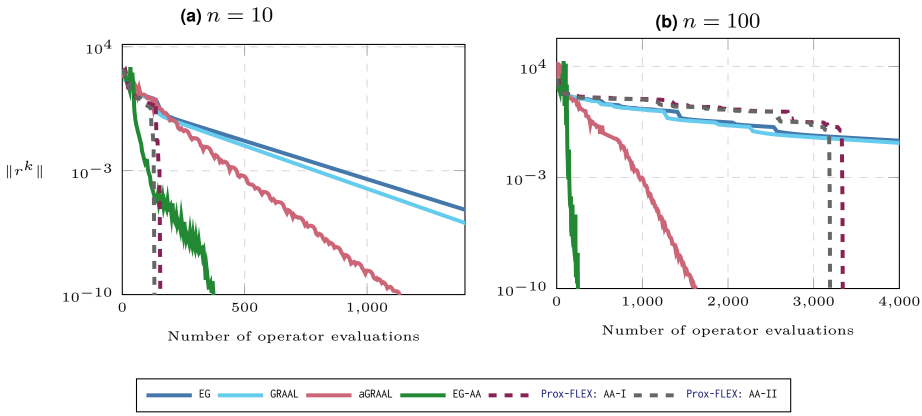


Figure 3. Convergence of algorithms on the Cournot–Nash equilibrium problem where $r^k = R_{1/2L_F}(z^k)$ and R is the residual mapping in (5.1). Both AA-I and AA-II use memory parameter $m = 3$. The number of operator evaluations equals the number of F and $\operatorname{prox}_{\gamma g}$ evaluations.

for each $z = (z_1, \ldots, z_n) \in \mathbb{R}^n$ and $i \in [\![1, n]\!]$, which fulfill the assumptions in the first paragraph of this section. We identify F as

$$F(z) = \underbrace{\begin{bmatrix} 2(a_1 + d_1) & d_1 & d_1 & \cdots & d_1 \\ d_2 & 2(a_2 + d_2) & d_2 & \cdots & d_2 \\ d_3 & d_3 & 2(a_3 + d_3) & \cdots & d_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_n & d_n & d_n & \cdots & 2(a_n + d_n) \end{bmatrix} z}_{=A}$$
$$+ \begin{bmatrix} b_1 - m_1 \\ \vdots \\ b_n - m_n \end{bmatrix}$$

with Lipschitz constant $L_F = \|A\|$. We also note that [36] provides the existence of a solution in this case. We generate the data similar to the approach in [8, Section 4.1], as outlined below. The results of the numerical experiments are presented in Figure 3. Although $n = 100$ in Figure 3b is not representative of a real oligopolistic market, we include this larger problem size to evaluate the performance and scalability of the algorithms. We observe that Prox-FLEX has a superlinear drop-off in both cases and that EG-AA and aGRAAL scale well for this particular problem.

6.4 Sparse logistic regression

Consider the sparse logistic regression problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \sum_{i=1}^{m} \log\left(1 + \exp\left(-b_i a_i^\top x\right)\right) + \lambda \|x\|_1 \tag{6.5}$$

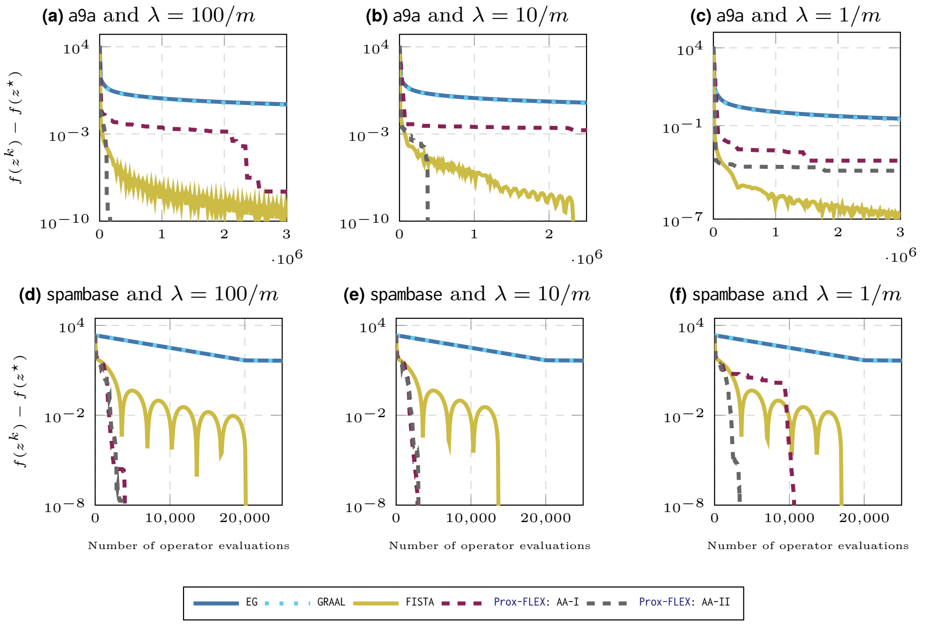


Figure 4. Convergence of algorithms on the sparse logistic regression problem (6.5), using the datasets a9a from [12] and spambase from [15]. Both AA-I and AA-II use memory parameter $m = 10$ for Figures 4a to 4c and $m = 6$ for Figures 4d to 4f. The number of operator evaluations equals the number of F and $\text{prox}_{\gamma g}$ evaluations.

where $(a_i, b_i) \in \mathbb{R}^n \times \{\pm 1\}$ for each $i = 1, \dots, m$. The minimization problem (6.5) can equivalently be written as the inclusion problem (1.1) by letting $\mathcal{H} = \mathbb{R}^n$ with the inner product set to the dot product, $F : \mathbb{R}^n \to \mathbb{R}^n$ such that $F(x) = K^\top \sigma(Kx)$ for each $x \in \mathbb{R}^n$ where

$$\sigma : \mathbb{R}^m \to \mathbb{R}^m : \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \mapsto \begin{bmatrix} \frac{\exp(u_1)}{1+\exp(u_1)} \\ \vdots \\ \frac{\exp(u_m)}{1+\exp(u_m)} \end{bmatrix}, \qquad K = \begin{bmatrix} -b_1 a_1^\top \\ \vdots \\ -b_m a_m^\top \end{bmatrix} \in \mathbb{R}^{m \times n},$$

and $g = \lambda \|\cdot\|_1$. Moreover, note that Assumption 2.1 holds with $L_F = (1/4)\|K\|^2$. The results of the numerical experiments are presented in Figure 4. Although not designed specifically for minimization problems, we observe that Prox-FLEX with AA-II directions performs at the top in all but one problem.

7. Conclusions

This paper investigated algorithms for solving inclusion problems involving the sum of a monotone and Lipschitz continuous operator and the subdifferential of a proper, convex, and lower semicontinuous function. We proposed a new Lyapunov function for Korpelevich's extragradient method and established a last-iterate convergence result. Departing from the standard Fejér-type analysis, this Lyapunov-based optimality measure did not rely on a known solution to the inclusion problem. It underpinned three novel algorithms that extend the extragradient method. These algorithms balanced user-specified directions and standard extragradient steps, guided by carefully designed line search steps based on the new Lyapunov analysis. In addition to providing global convergence results under various assumptions, we showed that when the directions are superlinear, no backtracking is triggered, leading to superlinear convergence.

Future research directions include developing similar solution-independent Lyapunov functions for, e.g., the forward-reflected-backward method by Malitsky and Tam [25]. Another promising direction is to broaden the scope of the analysis beyond the monotone setting to include cohypomonotone operators [32], and the more general class of problems characterized by the weak Minty condition [14, 23, 33]. Additionally, further exploration is warranted to adapt the approach to the mirror prox framework [28].

Appendix A: Background on Korpelevich's extragradient method

In the original paper [22], the extragradient method (1.2) was analyzed under the assumption that g is the indicator function of a nonempty, closed, and convex set, making the proximal operator reduce to the projection onto that set. However, as noted in [26], the extragradient method extends to the more general setting (1.1). The remainder of this section presents results in this more general context, with proofs included for completeness.

Definition A.1
Suppose that Assumption 2.1 holds and let $\gamma \in \mathbb{R}_{++}$. A point $z \in \mathcal{H}$ is said to be a fixed point of the extragradient method (1.2) if

$$\bar{z} = \operatorname{prox}_{\gamma g}(z - \gamma F(z)), \tag{A.1a}$$
$$z = \operatorname{prox}_{\gamma g}(z - \gamma F(\bar{z})). \tag{A.1b}$$

Proposition A.2
Suppose that Assumption 2.1 holds and let $\gamma \in \mathbb{R}_{++}$. Then, the following hold:

(i) If $z \in \operatorname{zer}(F + \partial g)$, then z is a fixed point of the extragradient method, i.e., (A.1) holds, and $z = \bar{z}$.

(ii) If $\gamma \in (0, 1/L_F)$, z is a fixed point of the extragradient method, and $\bar{z}$ is defined as in (A.1a), then $z = \bar{z} \in \operatorname{zer}(F + \partial g)$.

Proof. The proximal evaluations in (A.1a) and (A.1b) can equivalently be written via their subgradient characterization as

$$\gamma^{-1}(z - \bar{z}) - F(z) \in \partial g(\bar{z}), \tag{A.2a}$$
$$-F(\bar{z}) \in \partial g(z), \tag{A.2b}$$

respectively.

A.2.(i): Note that $z \in \operatorname{zer}(F + \partial g)$ and (A.1a) is equivalent to $-F(z) \in \partial g(z)$ and (A.2a), respectively. Using monotonicity of ∂g [6, Theorem 20.48], we get that

$$
\begin{aligned}
0 &\leq \langle \gamma^{-1}(z - \bar{z}) - F(z) + F(z), \bar{z} - z \rangle \\
&= -\gamma^{-1}\|z - \bar{z}\|^2 \leq 0,
\end{aligned}
$$

since $\gamma \in \mathbb{R}_{++}$. We conclude that $z = \bar{z}$ and that (A.1) holds.

A.2.(ii): By using monotonicity of ∂g at the points $\bar{z}$ and z, and the corresponding subgradients in (A.2), we get that

$$
\begin{aligned}
0 &\leq \langle \gamma^{-1}(z - \bar{z}) - F(z) + F(\bar{z}), \bar{z} - z \rangle \\
&= -\gamma^{-1}\|z - \bar{z}\|^2 + \langle F(\bar{z}) - F(z), \bar{z} - z \rangle \\
&\leq -\gamma^{-1}\|z - \bar{z}\|^2 + \|F(\bar{z}) - F(z)\|\|\bar{z} - z\| \\
&\leq (L_F - \gamma^{-1})\|z - \bar{z}\|^2,
\end{aligned}
\tag{A.3}
$$

where the Cauchy–Schwarz inequality is used in the second inequality, and Lipschitz continuity of F in the third inequality. Since $L_F - \gamma^{-1} < 0$, we conclude from (A.3) that $z = \bar{z}$. That $z = \bar{z} \in \operatorname{zer}(F + \partial g)$ now follows from (A.2a) or (A.2b). □

Proposition A.3

Suppose that Assumption 2.1 holds, the sequence $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ is generated by (1.2) with initial point $z^0 \in \mathcal{H}$ and step-size parameter $\gamma \in \mathbb{R}_{++}$, and $z^\star \in \operatorname{zer}(F + \partial g)$. Then

$$
\|z^{k+1} - z^\star\|^2 \leq \|z^k - z^\star\|^2 - (1 - \gamma^2 L_F^2)\|\bar{z}^k - z^k\|^2 \tag{A.4}
$$

for each $k \in \mathbb{N}_0$. Moreover, if $\gamma \in (0, 1/L_F)$, then $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to a point in $\operatorname{zer}(F + \partial g)$.

Proof. Note that the first and second proximal evaluations in (1.2) are equivalent to

$$
0 \leq g(z) - g(\bar{z}^k) - \langle \gamma^{-1}(z^k - \bar{z}^k) - F(z^k), z - \bar{z}^k \rangle \text{ for each } z \in \mathcal{H}, \tag{A.5}
$$

and

$$
0 \leq g(z) - g(z^{k+1}) - \langle \gamma^{-1}(z^k - z^{k+1}) - F(\bar{z}^k), z - z^{k+1} \rangle \text{ for each } z \in \mathcal{H}, \tag{A.6}
$$

respectively, and the assumption $z^\star \in \operatorname{zer}(F + \partial g)$ is equivalent to

$$
0 \leq g(z) - g(z^\star) - \langle -F(z^\star), z - z^\star \rangle \text{ for each } z \in \mathcal{H}. \tag{A.7}
$$

Picking $z = z^{k+1}$ in (A.5), $z = z^\star$ in (A.6), $z = \bar{z}^k$ in (A.7), summing the resulting inequalities, and multiplying by 2γ gives

$$
\begin{aligned}
0 \leq\ & 2\gamma g(z^{k+1}) - 2\gamma g(\bar{z}^k) - 2\langle z^k - \bar{z}^k - \gamma F(z^k), z^{k+1} - \bar{z}^k \rangle \\
&+ 2\gamma g(z^\star) - 2\gamma g(z^{k+1}) - 2\langle z^k - z^{k+1} - \gamma F(\bar{z}^k), z^\star - z^{k+1} \rangle \\
&+ 2\gamma g(\bar{z}^k) - 2\gamma g(z^\star) - 2\langle -\gamma F(z^\star), \bar{z}^k - z^\star \rangle \\
=\ & A_k + B_k,
\end{aligned}
$$

where

$$\begin{aligned}
A_k = &-2\langle z^k - \bar{z}^k, z^{k+1} - \bar{z}^k\rangle - 2\langle z^k - z^{k+1}, z^\star - z^{k+1}\rangle \\
&= \|z^k - z^{k+1}\|^2 - \|z^k - \bar{z}^k\|^2 - \|z^{k+1} - \bar{z}^k\|^2 + \|z^k - z^\star\|^2 - \|z^k - z^{k+1}\|^2 \\
&\quad - \|z^\star - z^{k+1}\|^2 \\
&= \|z^k - z^\star\|^2 - \|z^\star - z^{k+1}\|^2 - \|z^k - \bar{z}^k\|^2 - \|z^{k+1} - \bar{z}^k\|^2,
\end{aligned}$$

and

$$\begin{aligned}
B_k &= 2\gamma\langle F(z^k), z^{k+1} - \bar{z}^k\rangle + 2\gamma\langle F(\bar{z}^k), z^\star - z^{k+1}\rangle - 2\gamma\langle -F(z^\star), \bar{z}^k - z^\star\rangle \\
&= 2\gamma\langle F(z^k), z^{k+1} - \bar{z}^k\rangle + 2\gamma\langle F(\bar{z}^k), z^\star - z^{k+1}\rangle + 2\gamma\langle F(\bar{z}^k), \bar{z}^k - z^\star\rangle \\
&\quad - 2\gamma\langle F(\bar{z}^k) - F(z^\star), \bar{z}^k - z^\star\rangle \\
&\le 2\gamma\langle F(z^k), z^{k+1} - \bar{z}^k\rangle + 2\gamma\langle F(\bar{z}^k), z^\star - z^{k+1}\rangle + 2\gamma\langle F(\bar{z}^k), \bar{z}^k - z^\star\rangle \\
&= 2\gamma\langle F(z^k) - F(\bar{z}^k), z^{k+1} - \bar{z}^k\rangle \\
&\le \gamma^2\|F(z^k) - F(\bar{z}^k)\|^2 + \|z^{k+1} - \bar{z}^k\|^2 \\
&\le \gamma^2 L_F^2\|z^k - \bar{z}^k\|^2 + \|z^{k+1} - \bar{z}^k\|^2,
\end{aligned}$$

where monotonicity of F is used in the first inequality, Young's inequality is used in the second inequality, and Lipschitz continuity of F in the third inequality. We conclude that

$$0 \le A_k + B_k \le \|z^k - z^\star\|^2 - \|z^\star - z^{k+1}\|^2 - (1 - \gamma^2 L_F^2)\|z^k - \bar{z}^k\|^2,$$

which proves (A.4).

Next, note that (A.4) gives that $(\|z^k - z^\star\|)_{k\in\mathbb{N}_0}$ converges. Thus, $(z^k)_{k\in\mathbb{N}_0}$ is bounded and there exists a subsequence $(z^k)_{k\in K} \rightharpoonup z^\infty$ for some $z^\infty \in \mathcal{H}$ [6, Lemma 2.45]. Moreover, (A.4) and the requirement $\gamma \in (0, 1/L_F)$ give that $(\|\bar{z}^k - z^k\|^2)_{k\in\mathbb{N}_0}$ is summable, and therefore, $(\bar{z}^k)_{k\in K} \rightharpoonup z^\infty$. The first proximal evaluation in (1.2) can equivalently be written as

$$\gamma^{-1}(z^k - \bar{z}^k) - F(z^k) + F(\bar{z}^k) \in (F + \partial g)(\bar{z}^k). \tag{A.8}$$

The left-hand side of (A.8) converges strongly to zero since F is continuous and $(\|z^k - \bar{z}^k\|)_{k\in\mathbb{N}_0}$ converges to zero. Moreover, the operator $F + \partial g$ is maximally monotone, since F is maximally monotone (by continuity and monotonicity [6, Corollary 20.28]), ∂g is maximally monotone [6, Theorem 20.48], and F has full domain [6, Corollary 25.5]. Thus, [6, Proposition 20.38] gives that $z^\infty \in \operatorname{zer}(F + \partial g)$, and by [6, Lemma 2.47] we conclude that $(z^k)_{k\in\mathbb{N}_0}$ converges weakly to a point in $\operatorname{zer}(F + \partial g)$, as claimed. □

Remark A.4

Similar to Remark 2.7, the results in this section remain valid when ∂g in (1.1) is replaced with a maximally monotone and 3-cyclically monotone operator $T : \mathcal{H} \to \mathcal{H}$ and the proximal operators $\operatorname{prox}_{\gamma g}$ in (1.2) with the resolvent $(\mathrm{Id} + \gamma T)^{-1}$.

Appendix B: Counterexamples

Example B.1

Let $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$ for the Lyapunov function $\mathcal{V}$ defined in (2.1) and iterates $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ generated by Tseng's method (1.3). This example contains a particular *instance of the inclusion problem* (1.1)*, initial point* $z^0 \in \mathcal{H}$*, and step size* $\gamma \in (0, 1/L_F)$ *for which* $\mathcal{V}_k$ *increases between the first two consecutive iterations, thereby establishing that* $\mathcal{V}_k$ *has no (one-step) descent inequality in this case. In particular, consider* $\mathcal{H} = \mathbb{R}^4$*,* $F : \mathbb{R}^4 \to \mathbb{R}^4$*, and* $g : \mathbb{R}^4 \to \mathbb{R} \cup \{+\infty\}$ *such that*

$$F(z) = \begin{bmatrix} Ax \\ -A^\top y \end{bmatrix} \quad \text{and} \quad g(z) = \begin{cases} 0 & \text{if } x \in [-7, 6]^2 \text{ and } y \in [1, 8]^2, \\ +\infty & \text{otherwise} \end{cases}$$

for each $z = (x, y) \in \mathbb{R}^2 \times \mathbb{R}^2$*, respectively, where*

$$A = \begin{bmatrix} 7 & 6 \\ 1 & 0 \end{bmatrix}.$$

It is straightforward to verify that Assumption 2.1 holds with[4]

$$L_F = \left\| \begin{bmatrix} 0 & A \\ -A^\top & 0 \end{bmatrix} \right\| \approx 9.25091.$$

By letting $z^0 = (-1, -7, -1, 7)$ *and* $\gamma = 1/10$*, Tseng's method gives that*

$$\begin{aligned} \bar{z}^0 &= \left(-\frac{9}{2}, -\frac{69}{10}, 1, \frac{32}{5}\right), \\ z^1 &= \left(-\frac{277}{50}, -\frac{71}{10}, -\frac{36}{25}, \frac{43}{10}\right), \\ \bar{z}^1 &= \left(-7, -\frac{1739}{250}, 1, 1\right), \end{aligned}$$

and therefore

$$\mathcal{V}_0 = 1662 \quad \text{and} \quad \mathcal{V}_1 = \frac{1187246}{625} = 1899.5936,$$

establishing the claim.[5]

Example B.2
Consider the inclusion problem

$$\text{find } z \in \mathcal{H} \text{ such that } 0 \in F(z) + T(z)$$

where $F : \mathcal{H} \to \mathcal{H}$ *satisfies Assumption 2.1.(i) and* $T : \mathcal{H} \to 2^{\mathcal{H}}$ *is a maximally monotone operator. Let*

$$\begin{aligned} \bar{z}^k &= (\mathrm{Id} + \gamma T)^{-1}(z^k - \gamma F(z^k)), \\ z^{k+1} &= (\mathrm{Id} + \gamma T)^{-1}(z^k - \gamma F(\bar{z}^k)), \end{aligned} \tag{B.1}$$

and $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$ *for the Lyapunov function* $\mathcal{V}$ *defined in* (2.1)*. This example contains a particular problem instance for which* $(z^k)_{k \in \mathbb{N}_0}$ *diverges and* $\mathcal{V}_k$ *increases*

[4] The matrix norm is taken as the spectral norm.
[5] Code to reproduce this example can be found at https://github.com/ManuUpadhyaya/flex/blob/main/Counterexample_B1.ipynb.

between the first two consecutive iterations. In particular, consider $\mathcal{H} = \mathbb{R}^2$, $z^0 = (10, 10)$, $\gamma = 1/10$, *and* $F, T : \mathbb{R}^2 \to \mathbb{R}^2$ *such that*

$$F(z) = \underbrace{\begin{bmatrix} 0 & 9 \\ -9 & 0 \end{bmatrix}}_{=A} \begin{bmatrix} x \\ y \end{bmatrix} \quad \text{and} \quad T(z) = \underbrace{\begin{bmatrix} 0 & -4 \\ 4 & 0 \end{bmatrix}}_{=B} \begin{bmatrix} x \\ y \end{bmatrix}$$

for each $z = (x, y) \in \mathbb{R} \times \mathbb{R}$, *where* $L_F = 9$. *Note that* (B.1) *reduces to*

$$z^{k+1} = \underbrace{(I + \gamma B)^{-1} \left(I - \gamma A \left((I + \gamma B)^{-1} (I - \gamma A)\right)\right)}_{=C} z^k,$$

where $C \in \mathbb{R}^{2\times 2}$ *has full rank and spectral radius* ≈ 1.132596, *which is greater than one. Therefore, we can conclude that* $(z^k)_{k\in\mathbb{N}_0}$ *diverges. Moreover,* (B.1) *gives that*

$$\begin{aligned}
\bar{z}^0 &= \left(\frac{215}{29}, \frac{465}{29}\right), \\
z^1 &= \left(\frac{3245}{1682}, \frac{26745}{1682}\right), \\
\bar{z}^1 &= \left(-\frac{447965}{97556}, \frac{1899785}{97556}\right), \\
z^2 &= \left(-\frac{53118995}{5658248}, \frac{87834005}{5658248}\right),
\end{aligned}$$

and therefore

$$\begin{aligned}
\mathcal{V}_0 &= \frac{5875000}{841} \approx 6985.73127229489, \\
\mathcal{V}_1 &= \frac{12676046875}{1414562} \approx 8961.11084208398,
\end{aligned}$$

establishing the second claim.[6]

Appendix C: Comparison to standard optimality measures

This section presents some standard optimality measures for (1.1) and compares them to $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$ for the Lyapunov function $\mathcal{V}$ defined in (2.1) and iterates $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ generated by the extragradient method (1.2). It also includes a comparison to recent last-iterate convergence results for the extragradient method.

Definition C.1
Suppose that Assumption 2.1 holds.

(i) The natural residual *is defined as*

$$\|z - \operatorname{prox}_g(z - F(z))\|$$

for each $z \in \mathcal{H}$.

[6] Code to reproduce this example can be found at https://github.com/ManuUpadhyaya/flex/blob/main/Counterexample_B2.ipynb.

(ii) The tangent residual *is defined as*

$$\inf_{\xi \in \partial g(z)} \|F(z) + \xi\|$$

for each $z \in \mathcal{H}$.

(iii) Suppose that $z^0 \in \mathcal{H}$, $z^\star \in \operatorname{zer}(F + \partial g)$ *and* $\delta = \|z^0 - z^\star\| > 0$. *Then the restricted gap function* $\mathsf{Gap} : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ *is defined as*

$$\mathsf{Gap}(z) = \sup_{w \in \operatorname{dom} \partial g \cap \mathbb{B}(z^\star;\delta)} \left(\langle F(w), z - w\rangle + g(z) - g(w)\right) \tag{C.1}$$

for each $z \in \mathcal{H}$, *where* $\mathbb{B}(z^\star;\delta) = \{z \in \mathcal{H} : \|z - z^\star\| \leq \delta\}$.

Remark C.2
In this remark, we establish that the measures in Definition C.1 are indeed optimality measures for (1.1).

(i) The natural residual is nonnegative and zero if and only if $z \in \operatorname{zer}(F + \partial g)$, *since*

$$\|z - \operatorname{prox}_g(z - F(z))\| = 0 \iff z = \operatorname{prox}_g(z - F(z)) \iff -F(z) \in \partial g(z),$$

where the last equivalence follows from the subgradient characterization of the proximal operator.

(ii) The tangent residual is nonnegative and zero if and only if $z \in \operatorname{zer}(F + \partial g)$, *since the tangent residual upper bounds the natural residual by Proposition C.3.(ii).*

(iii) It is well-known that (e.g., see [29, Lemma 1] and [24, Lemma 3])

- $\mathsf{Gap}(z) \geq 0$ *for each* $z \in \mathcal{H}$,
- *if* $z \in \operatorname{zer}(F + \partial g) \cap \mathbb{B}(z^\star;\delta)$, *then* $\mathsf{Gap}(z) = 0$, *and*
- *if* $z \in \operatorname{dom} \partial g \cap \mathbb{B}(z^\star;\tilde{\delta})$ *for some* $\tilde{\delta} < \delta$, *then* $\mathsf{Gap}(z) = 0$ *implies that* $z \in \operatorname{zer}(F + \partial g)$.

Note that the second proximal step in (1.2) can equivalently be written via its subgradient characterization as

$$\underbrace{\gamma^{-1}(z^k - z^{k+1}) - F(\bar{z}^k)}_{=\xi^{k+1}} \in \partial g(z^{k+1}) \tag{C.2}$$

for each $k \in \mathbb{N}_0$. This notation allows us to introduce another standard optimality measure, namely $(\|F(z^k) + \xi^k\|)_{k\in\mathbb{N}}$. It is clear that this measure upper bounds the tangent residual, i.e.,

$$\inf_{\xi \in \partial g(z^k)} \|F(z^k) + \xi\| \leq \|F(z^k) + \xi^k\| \tag{C.3}$$

for each $k \in \mathbb{N}$.

Next, we present a result that, when combined with (C.3), shows that $\mathcal{V}_k$ upper bounds all the squared optimality measures considered above, except the squared restricted gap function, which is upper bounded up to a positive constant.

Proposition C.3

Suppose that Assumption 2.1 holds, the sequence $((z^k, \bar{z}^k))_{k\in\mathbb{N}_0}$ is generated by (1.2) with initial point $z^0 \in \mathcal{H}$ and step-size parameter $\gamma \in (0, 1/L_F)$, the sequence $(\mathcal{V}_k)_{k\in\mathbb{N}_0}$ is given by $\mathcal{V}_k = \mathcal{V}(z^k, \bar{z}^k, z^{k+1})$ for each $k \in \mathbb{N}_0$ and the Lyapunov function $\mathcal{V}$ defined in (2.1), and the sequence $(\xi^k)_{k\in\mathbb{N}}$ is given by (C.2). Then,

(i) $\|F(z^{k+1}) + \xi^{k+1}\|^2 \le \mathcal{V}_k$ *for each* $k \in \mathbb{N}_0$,

(ii) $\|z - \operatorname{prox}_g(z - F(z))\| \le \inf_{\xi\in\partial g(z)} \|F(z) + \xi\|$ *for each* $z \in \mathcal{H}$,

(iii) $\mathsf{Gap}(z) \le (\delta + \|z - z^\star\|) \inf_{\xi\in\partial g(z)} \|F(z) + \xi\|$ *for each* $z \in \mathcal{H}$*, and in particular,* $\mathsf{Gap}(z^k) \le 2\delta \inf_{\xi\in\partial g(z^k)} \|F(z^k) + \xi\|$ *for each* $k \in \mathbb{N}$*, where* Gap *is defined in* (C.1).

Proof. The proofs of Propositions C.3.(ii) and C.3.(iii) are simple generalizations of corresponding results found in [37], which we include for completeness.

C.3.(i) Note that

$$
\begin{aligned}
&\|F(z^{k+1}) + \xi^{k+1}\|^2 \\
&= \|\tfrac{1}{\gamma}\left(z^k - z^{k+1}\right) + F(z^{k+1}) - F(\bar{z}^k)\|^2 \\
&= \|F(z^{k+1}) - F(\bar{z}^k)\|^2 + \tfrac{1}{\gamma^2}\|z^k - z^{k+1}\|^2 + \tfrac{2}{\gamma}\langle F(z^{k+1}) - F(\bar{z}^k), z^k - z^{k+1}\rangle \\
&\le L_F^2\|z^{k+1} - \bar{z}^k\|^2 + \tfrac{1}{\gamma^2}\|z^k - z^{k+1}\|^2 + \tfrac{2}{\gamma}\langle F(z^{k+1}) - F(\bar{z}^k), z^k - z^{k+1}\rangle \\
&\le \tfrac{1}{\gamma^2}\|z^{k+1} - \bar{z}^k\|^2 + \tfrac{1}{\gamma^2}\|z^k - z^{k+1}\|^2 + \tfrac{2}{\gamma}\langle F(z^k) - F(\bar{z}^k), z^k - z^{k+1}\rangle \\
&= \mathcal{V}_k
\end{aligned}
$$

where Lipschitz continuity of F was used in the first inequality, and $\gamma L_F \le 1$ and monotonicity of F was used in the second inequality.

C.3.(ii) We claim that the natural residual is upper bounded by the tangent residual, i.e.,

$$
\|z - \operatorname{prox}_g(z - F(z))\| \le \inf_{\xi\in\partial g(z)} \|F(z) + \xi\| \tag{C.4}
$$

for each $z \in \mathcal{H}$. When $z \notin \operatorname{dom}\partial g$, then (C.4) holds trivially. Thus, suppose that $z \in \operatorname{dom}\partial g$ and let $\xi \in \partial g(z)$. The latter inclusion holds if and only if $z = \operatorname{prox}_g(z + \xi)$. Therefore,

$$
\|z - \operatorname{prox}_g(z - F(z))\| = \|\operatorname{prox}_g(z + \xi) - \operatorname{prox}_g(z - F(z))\| \le \|F(z) + \xi\|
$$

where the inequality follows from the nonexpansivity of the proximal operator [6, Proposition 12.28]. However, since $\xi \in \partial g(z)$ is arbitrary, (C.4) follows.

C.3.(iii) We claim that the restricted gap function is upper bounded by the tangent residual up to a positive quantity, i.e.,

$$
\mathsf{Gap}(z) \le (\delta + \|z - z^\star\|) \inf_{\xi\in\partial g(z)} \|F(z) + \xi\| \tag{C.5}
$$

for each $z \in \mathcal{H}$, and therefore,

$$
\mathsf{Gap}(z^k) \le 2\delta \inf_{\xi\in\partial g(z^k)} \|F(z^k) + \xi\| \tag{C.6}
$$

for each $k \in \mathbb{N}$, by Proposition A.3. Let us prove (C.5). If $z \notin \operatorname{dom} \partial g$, then (C.5) holds trivially. Thus, assume that $z \in \operatorname{dom} \partial g$, and let $\xi \in \partial g(z)$ and $w \in \operatorname{dom} \partial g \cap \mathbb{B}(z^\star; \delta)$. Then we have that

$$\begin{aligned}
&\langle F(w), z - w\rangle + g(z) - g(w) \\
&= \underbrace{\langle F(w) - F(z), z - w\rangle}_{\leq 0} + \langle F(z), z - w\rangle + \underbrace{g(z) - g(w)}_{\leq \langle \xi, z - w\rangle} \\
&\leq \langle F(z) + \xi, z - z^\star\rangle + \langle F(z) + \xi, z^\star - w\rangle \\
&\leq \|F(z) + \xi\| \|z - z^\star\| + \|F(z) + \xi\| \underbrace{\|z^\star - w\|}_{\leq \delta} \\
&\leq (\delta + \|z - z^\star\|)\|F(z) + \xi\|,
\end{aligned}$$

or

$$\langle F(w), z - w\rangle + g(z) - g(w) \leq (\delta + \|z - z^\star\|)\|F(z) + \xi\|. \tag{C.7}$$

Maximizing over $w \in \operatorname{dom} \partial g \cap \mathbb{B}(z^\star; \delta)$ and minimizing over $\xi \in \partial g(z)$ in (C.7) gives (C.5), as claimed. □

Corollary C.4

Suppose that Assumption 2.1 and $\operatorname{zer}(F + \partial g) \neq \emptyset$ *hold, the sequence* $((z^k, \bar{z}^k))_{k \in \mathbb{N}_0}$ *is generated by (1.2) with initial point* $z^0 \in \mathcal{H}$ *and step-size parameter* $\gamma \in (0, 1/L_F)$*, and the sequence* $(\xi^k)_{k \in \mathbb{N}}$ *is given by (C.2). Then,*

$$\begin{aligned}
\|F(z^k) + \xi^k\| &\in o(1/\sqrt{k}) \text{ as } k \to \infty, \\
\inf_{\xi \in \partial g(z^k)} \|F(z^k) + \xi\| &\in o(1/\sqrt{k}) \text{ as } k \to \infty, \\
\|z^k - \operatorname{prox}_g(z - F(z^k))\| &\in o(1/\sqrt{k}) \text{ as } k \to \infty, \\
\mathsf{Gap}(z^k) &\in o(1/\sqrt{k}) \text{ as } k \to \infty,
\end{aligned}$$

and for any $k \in \mathbb{N}$ *and* $z^\star \in \operatorname{zer}(F + \partial g)$ *it holds that*

$$\begin{aligned}
\|z^k - \operatorname{prox}_g(z - F(z^k))\| \leq \inf_{\xi \in \partial g(z^k)} \|F(z^k) + \xi\| \leq \|F(z^k) + \xi^k\| &\leq \frac{\|z^0 - z^\star\|}{\sqrt{\alpha(\gamma, L_F) k}}, \\
\mathsf{Gap}(z^k) &\leq \frac{2\delta\|z^0 - z^\star\|}{\sqrt{\alpha(\gamma, L_F) k}},
\end{aligned}$$

where $\alpha(\gamma, L_F) > 0$ *is defined in (2.15) and* Gap *is defined in (C.1).*

Proof. Follows immediately from (C.3), Proposition C.3, and Corollary 2.6. □

Remark C.5

Simple computation shows that Corollary C.4 sharpens the recent last-iterate rates in [11, Theorem 3] and [43, Corollary 4.1].

Remark C.6

Similar to Remark 2.7, the results in this section remain valid (except the ones involving the restricted gap function Gap*) when* ∂g *in (1.1) is replaced with a maximally monotone and 3-cyclically monotone operator* $T : \mathcal{H} \to \mathcal{H}$ *and the proximal operators* $\operatorname{prox}_{\gamma g}$ *in (1.2) with the resolvent* $(\mathrm{Id} + \gamma T)^{-1}$.

Acknowledgments

M. Upadhyaya and P. Giselsson acknowledge support from the ELLIIT Strategic Research Area and the Wallenberg AI, Autonomous Systems, and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation. Additionally, P. Giselsson acknowledges support from the Swedish Research Council. P. Latafat is a member of the Gruppo Nazionale per l'Analisi Matematica, la Probabilità e le loro Applicazioni (GNAMPA - National Group for Mathematical Analysis, Probability and their Applications) of the Istituto Nazionale di Alta Matematica (INdAM - National Institute of Higher Mathematics). We thank Yurong Chen for highlighting some details in Section 6.3. We also thank Max Nilsson and the anonymous reviewers for providing valuable comments and suggestions for a previous version of this work.

References

[1] M. Ahookhosh, A. Themelis, and P. Patrinos. "A Bregman forward-backward linesearch algorithm for nonconvex composite optimization: Superlinear convergence to nonisolated local minima". *SIAM Journal on Optimization* **31**:1 (2021), pp. 653–685. DOI: 10.1137/19m1264783.

[2] A. Alacaoglu, A. Böhm, and Y. Malitsky. "Beyond the golden ratio for variational inequality algorithms". *Journal of Machine Learning Research* **24**:172 (2023), pp. 1–33.

[3] D. G. Anderson. "Iterative procedures for nonlinear integral equations". *Journal of the ACM* **12**:4 (1965), pp. 547–560. DOI: 10.1145/321296.321305.

[4] A. Asl, H. Lu, and J. Yang. "A J-symmetric quasi-Newton method for minimax problems". *Mathematical Programming* **204**:1–2 (2024), pp. 207–254. DOI: 10.1007/s10107-023-01957-1.

[5] S. Axler. *Measure, Integration & Real Analysis*. Springer International Publishing, 2020. ISBN: 9783030331436. DOI: 10.1007/978-3-030-33143-6.

[6] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer International Publishing, 2017. ISBN: 9783319483115. DOI: 10.1007/978-3-319-48311-5.

[7] A. Beck and M. Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". *SIAM Journal on Imaging Sciences* **2**:1 (2009), pp. 183–202. DOI: 10.1137/080716542.

[8] G. Bigi and M. Passacantando. "Differentiated oligopolistic markets with concave cost functions via Ky Fan inequalities". *Decisions in Economics and Finance* **40**:1–2 (2017), pp. 63–79. DOI: 10.1007/s10203-017-0187-7.

[9] G. Bonanno. "General equilibrium theory with imperfect competition". *Journal of Economic Surveys* **4**:4 (1990), pp. 297–328. DOI: 10.1111/j.1467-6419.1990.tb00091.x.

[10] R. I. Bot and E. Chenchene. "Extra-gradient method with flexible anchoring: Strong convergence and fast residual decay" (2024). arXiv: 2410.14369v1 [math.OC].

[11] Y. Cai, A. Oikonomou, and W. Zheng. "Tight last-iterate convergence of the extragradient and the optimistic gradient descent-ascent algorithm for constrained monotone variational inequalities" (2022). arXiv: 2204.09228v3 [math.OC].

[12] C.-C. Chang and C.-J. Lin. "LIBSVM: A library for support vector machines". *ACM Transactions on Intelligent Systems and Technology* **2**:3 (2011), pp. 1–27. DOI: 10.1145/1961189.1961199.

[13] J. E. Dennis Jr. and J. J. Moré. "Quasi-Newton methods, motivation and theory". *SIAM Review* **19**:1 (1977), pp. 46–89. DOI: 10.1137/1019005.

[14] J. Diakonikolas, C. Daskalakis, and M. Jordan. "Efficient methods for structured nonconvex-nonconcave min-max optimization". In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2021.

[15] D. Dua and C. Graff. *UCI machine learning repository*. 2017. URL: http://archive.ics.uci.edu/ml.

[16] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Vol. I. Springer New York, 2004. ISBN: 9780387955803. DOI: 10.1007/b97543.

[17] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Vol. II. Springer New York, 2004. ISBN: 9780387955810. DOI: 10.1007/b97544.

[18] H.-r. Fang and Y. Saad. "Two classes of multisecant methods for nonlinear acceleration". *Numerical Linear Algebra with Applications* **16**:3 (2009), pp. 197–221. DOI: 10.1002/nla.617.

[19] D. Fudenberg and J. Tirole. *Dynamic Models of Oligopoly*. 2nd ed. Fundamentals of pure and applied economics 3. Harwood Academic Publishers, 1990. 83 pp. ISBN: 3718602792.

[20] E. Gorbunov, N. Loizou, and G. Gidel. "Extragradient method: $\mathcal{O}(1/K)$ last-iterate convergence for monotone variational inequalities and connections with cocoercivity". In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2022.

[21] A. F. Izmailov and M. V. Solodov. *Newton-Type Methods for Optimization and Variational Problems*. Springer International Publishing, 2014. ISBN: 9783319042473. DOI: 10.1007/978-3-319-04247-3.

[22] G. M. Korpelevich. "The extragradient method for finding saddle points and other problems". *Matecon* **12** (1976), pp. 747–756.

[23] S. Lee and D. Kim. "Fast extra gradient methods for smooth structured nonconvex-nonconcave minimax problems". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2021.

[24] Y. Malitsky. "Golden ratio algorithms for variational inequalities". *Mathematical Programming* **184**:1–2 (2020), pp. 383–410. DOI: 10.1007/s10107-019-01416-w.

[25] Y. Malitsky and M. K. Tam. "A forward-backward splitting method for monotone inclusions without cocoercivity". *SIAM Journal on Optimization* **30**:2 (2020), pp. 1451–1472. DOI: 10.1137/18m1207260.

[26] R. D. C. Monteiro and B. F. Svaiter. "Complexity of variants of Tseng's modified F-B splitting and Korpelevich's methods for hemivariational inequalities with applications to saddle-point and convex optimization problems". *SIAM Journal on Optimization* **21**:4 (2011), pp. 1688–1720. DOI: 10.1137/100801652.

[27] L. D. Muu, V. H. Nguyen, and N. V. Quy. "On Nash-Cournot oligopolistic market equilibrium models with concave cost functions". *Journal of Global Optimization* **41**:3 (2007), pp. 351–364. DOI: 10.1007/s10898-007-9243-0.

[28] A. Nemirovski. “Prox-method with rate of convergence $\mathcal{O}(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems”. *SIAM Journal on Optimization* **15**:1 (2004), pp. 229–251. DOI: 10.1137/s1052623403425629.

[29] Y. Nesterov. “Dual extrapolation and its applications to solving variational inequalities and related problems”. *Mathematical Programming* **109**:2–3 (2006), pp. 319–344. DOI: 10.1007/s10107-006-0034-z.

[30] M. A. Noor. “Mixed variational inequalities”. *Applied Mathematics Letters* **3**:2 (1990), pp. 73–75. DOI: 10.1016/0893-9659(90)90018-7.

[31] K. Okuguchi and F. Szidarovszky. *The Theory of Oligopoly with Multi-Product Firms*. Springer Berlin Heidelberg, 1999. ISBN: 9783642601699. DOI: 10.1007/978-3-642-60169-9.

[32] T. Pennanen. “Local convergence of the proximal point algorithm and multiplier methods without monotonicity”. *Mathematics of Operations Research* **27**:1 (2002), pp. 170–191. DOI: 10.1287/moor.27.1.170.331.

[33] T. Pethick, P. Latafat, P. Patrinos, O. Fercoq, and V. Cevher. “Escaping limit cycles: Global convergence for constrained nonconvex-nonconcave minimax problems”. In: *International Conference on Learning Representations*. 2022.

[34] L. D. Popov. “A modification of the Arrow-Hurwicz method for search of saddle points”. *Mathematical Notes of the Academy of Sciences of the USSR* **28**:5 (1980), pp. 845–848. DOI: 10.1007/bf01141092.

[35] X. Qu, W. Bian, and X. Chen. “An extra gradient Anderson-accelerated algorithm for pseudomonotone variational inequalities” (2024). arXiv: 2408.06606v1 [math.OC].

[36] J. B. Rosen. “Existence and uniqueness of equilibrium points for concave N-person games”. *Econometrica* **33**:3 (1965), pp. 520–534. DOI: 10.2307/1911749.

[37] M. Sedlmayer, D.-K. Nguyen, and R. I. Bot. “A fast optimistic method for monotone variational inequalities”. In: *International Conference on Machine Learning (ICML)*. 2023.

[38] M. V. Solodov and B. F. Svaiter. “A globally convergent inexact Newton method for systems of monotone equations”. In: *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*. Springer US, 1998, pp. 355–369. ISBN: 9781475763881. DOI: 10.1007/978-1-4757-6388-1_18.

[39] M. V. Solodov and B. F. Svaiter. “A hybrid approximate extragradient-proximal point algorithm using the enlargement of a maximal monotone operator”. *Set-Valued Analysis* **7**:4 (1999), pp. 323–345. DOI: 10.1023/a:1008777829180.

[40] L. Stella, A. Themelis, P. Sopasakis, and P. Patrinos. “A simple and efficient algorithm for nonlinear model predictive control”. In: *Conference on Decision and Control (CDC)*. 2017. DOI: 10.1109/cdc.2017.8263933.

[41] A. Themelis and P. Patrinos. “SuperMann: A superlinearly convergent algorithm for finding fixed points of nonexpansive operators”. *IEEE Transactions on Automatic Control* **64**:12 (2019), pp. 4875–4890. DOI: 10.1109/tac.2019.2906393.

[42] Q. Tran-Dinh and N. Nguyen-Trung. “Accelerated extragradient-type methods – Part 2: Generalization and sublinear convergence rates under co-hypomonotonicity” (2025). arXiv: 2501.04585v2 [math.OC].

[43] Q. Tran-Dinh and N. Nguyen-Trung. “Revisiting extragradient-type methods – Part 1: Generalizations and sublinear convergence rates” (2024). arXiv: 2409.16859v1 [math.OC].

[44] P. Tseng. “A modified forward-backward splitting method for maximal monotone mappings”. *SIAM Journal on Control and Optimization* **38**:2 (2000), pp. 431–446. DOI: 10.1137/s0363012998338806.

[45] X. Vives. “Cournot and the oligopoly problem”. *European Economic Review* **33**:2–3 (1989), pp. 503–514. DOI: 10.1016/0014-2921(89)90129-3.

[46] P. von Mouche and F. Quartieri. *Equilibrium Theory for Cournot Oligopolies and Related Games*. Springer International Publishing, 2016. ISBN: 9783319292540. DOI: 10.1007/978-3-319-29254-0.

[47] T. Yoon and E. K. Ryu. “Accelerated algorithms for smooth convex-concave minimax problems with $\mathcal{O}(1/k^2)$ rate on squared gradient norm”. In: *International Conference on Machine Learning (ICML)*. 2021.

[48] J. Zhang, B. O’Donoghue, and S. Boyd. “Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations”. *SIAM Journal on Optimization* **30**:4 (2020), pp. 3170–3197. DOI: 10.1137/18m1232772.

Lyapunovanalyser av optimeringsmetoder

Manu Upadhyaya

Institutionen för Reglerteknik

Populärvetenskaplig sammanfattning av doktorsavhandlingen *Lyapunov analyses for first-order methods: Theory, automation, and algorithm design*, september 2025.

Optimeringsmetoder spelar en central roll i modern teknik och vetenskap genom att möjliggöra systematisk förbättring av prestanda under givna begränsningar. Exempelvis används det för att konstruera styrlagar inom reglertekniska system; styrning av värme och ventilation i byggnader är ett tillämpningsområde som kan minska energiförbrukning och koldioxidutsläpp. Vid strålbehandling av cancer används optimering för att bestämma en dosfördelning som levererar tillräckligt hög dos till tumörområdet för att uppnå effektiv behandling, samtidigt som det minimerar stråldosen som träffar frisk vävnad. Inom medicinsk bildbehandling används optimering för att förbättra bildkvaliteten och rekonstruktionen i tekniker som magnetresonanstomografi, datortomografi och positronemissionstomografi. Vid design av elektroniska kretsar används optimering för att förbättra prestanda, minska energiförbrukning, optimera den fysiska utformningen av kretsen och säkerställa att teknologiska specifikationer uppfylls. Inom maskininlärning och artificiell intelligens är optimering en grundläggande komponent för att träna prediktiva modeller, såsom neurala nätverk, som lär sig mönster från stora datamängder. På senare tid har optimeringsmetoder även möjliggjort utvecklingen av generativa modeller som kan skapa realistiska data i form av text, bilder och video. Men hur vet vi egentligen att dessa metoder är effektiva och tillförlitliga?

Denna avhandling undersöker detta med hjälp av Lyapunovanalys, en klassisk matematisk teknik som ursprungligen utvecklades för att studera stabiliteten hos mekaniska system. Lyapunovanalysen används här för att analysera hur snabbt optimeringsalgoritmer konvergerar till en korrekt lösning.

Närmare bestämt fokuserar avhandlingen på utvecklingen av automatiserade verktyg som låter datorer själva utvärdera algoritmers prestanda. Denna typ av verktyg automatiserar de mest utmanande delarna av analysprocessen, nämligen att skapa matematiska bevis, vilket tidigare utfördes manuellt av experter. Genom att formulera Lyapunovanalysen som ett litet ekvationssystem, som en dator enkelt kan lösa, kan verktyget ta fram resultat betydligt snabbare än en människa. Dessutom kan det hitta bevis som i praktiken ligger bortom vad en expert kan åstadkomma, eftersom det systematiskt söker igenom alla möjliga bevis inom en viss struktur. Det innebär att ett enormt antal alternativ kan undersökas, något som vore omöjligt att göra för hand. Genom att automatisera analysen kan vi inte bara verifiera algoritmers prestanda, utan även systematiskt vägleda utvecklingen av ännu bättre metoder.

Printed by Media-Tryck, Lund 2025 NORDIC SWAN ECOLABEL 3041 0903

LUND UNIVERSITY

Department of Automatic Control
P.O. Box 118, 221 00 Lund, Sweden
www.control.lth.se

PhD Thesis TFRT-1150
ISBN 978-91-8104-589-5
ISSN 0280–5316