# Negative Momentum for Convex-Concave Optimization

Henry Shugart
UPenn
hshugart@upenn.edu

Shuyi Wang
Yale
shuyi.wang@yale.edu

Jason M. Altschuler
UPenn
alts@upenn.edu

April 18, 2026

**Abstract**

This paper revisits momentum in the context of min-max optimization. Momentum is a celebrated mechanism for accelerating gradient dynamics in settings like convex minimization, but its direct use in min-max optimization makes gradient dynamics diverge. Surprisingly, [16] showed that *negative* momentum can help fix convergence. However, despite these promising initial results and progress since, the power of momentum remains unclear for min-max optimization in two key ways. (1) Generality: is global convergence possible for the foundational setting of convex-concave optimization? This is the direct analog of convex minimization and is a standard testing ground for min-max algorithms. (2) Fast convergence: is accelerated convergence possible for strongly-convex-strong-concave optimization (the only non-linear setting where global convergence is known)? Recent work has even argued that this is impossible. We answer both these questions in the affirmative. Together, these results put negative momentum on more equal footing with competitor algorithms, and show that negative momentum enables convergence significantly faster and more generally than was known possible.

## 1 Introduction

Efficiently solving min-max problems (also called saddle-point problems) of the form

$$\min_x \max_y f(x, y)$$

is essential for high-impact applications across machine learning, game theory, robust optimization, distributed optimization, constrained optimization, and more [4, 5, 6, 8, 17, 21, 30, 45]. A major challenge for min-max optimization is that, despite close connections to standard optimization, many algorithmic phenomena are fundamentally different. An infamous example is that although gradient descent (GD) converges for convex optimization with appropriately chosen positive step-sizes, the direct analog is false for min-max optimization [23].

This paper revisits *momentum*, a powerful algorithmic technique in standard optimization whose potential remains somewhat unclear for min-max optimization. The traditional intuition behind momentum is to augment an iterative algorithm $z_{t+1} = h(z_t)$ to $z_{t+1} = h(z_t) + \beta(z_t - z_{t-1})$ for some momentum parameter $\beta > 0$, in order to accelerate the dynamics along important directions. Celebrated results dating back to Polyak [35] and Nesterov [34] establish this phenomenon formally for convex minimization, proving accelerated convergence rates in fundamental settings such as (strongly) convex, smooth objectives. In recent decades, momentum has been established more generally as a powerful mechanism for accelerating iterative algorithms across a broad swathe of optimization settings; see the survey [13].

**Direct adaptation of momentum fails.** However, a fundamental challenge in min-max optimization is that the straightforward adaptation of momentum fails to make gradient dynamics converge. To explain this, first recall that the analog of gradient descent for min-max optimization is *gradient-descent-ascent* (GDA), in which $x$ takes a descent step while $y$ takes an ascent step:

$$\begin{aligned} x_{t+1} &= x_t - \eta \nabla_x f(x_t, y_t)\,, \\ y_{t+1} &= y_t + \eta \nabla_y f(x_t, y_t)\,. \end{aligned} \tag{1.1}$$

The direct adaptation of momentum for GDA would then be

$$\begin{aligned} x_{t+1} &= x_t - \eta \nabla_x f(x_t, y_t) + \beta(x_t - x_{t-1})\,, \\ y_{t+1} &= y_t + \eta \nabla_y f(x_t, y_t) + \beta(y_t - y_{t-1})\,. \end{aligned} \tag{1.2}$$

Yet this algorithm (1.2) fails to converge for any choice of momentum parameter $\beta > 0$, even in simple settings such as the 1-dimensional unconstrained bilinear problem $\min_x \max_y xy$ [16].

**Negative momentum and alternation.** Intriguingly, *negative momentum* $\beta < 0$ can help remedy this issue—an influential idea proposed by the seminal paper [16]. This is counterintuitive from the perspective of standard (non min-max) optimization: negative momentum downweights past motion rather than upweights. In min-max optimization, however, negative momentum helps because it partially suppresses the non-convergent rotational dynamics of GDA, so that the iterates spend less effort cycling around the saddle and more effort moving in genuinely improving directions. Crucial for making this negative momentum work, even in the simple setting of bilinear $f$, is to *alternate* the $x$ and $y$ updates:

$$\begin{aligned} x_{t+1} &= x_t - \eta \nabla_x f(x_t, y_t) + \beta(x_t - x_{t-1})\,, \\ y_{t+1} &= y_t + \eta \nabla_y f(x_{t+1}, y_t) + \beta(y_t - y_{t-1}). \end{aligned} \tag{1.3}$$

(Note the use of $x_{t+1}$ to update $y_{t+1}$.) Specifically, [16] showed theoretically that this algorithm (1.3) (with negative momentum and alternation) enables global convergence for bilinear objectives $f$, and helps empirically for complicated non-convex-non-concave settings like GANs.

However, despite these initial promising results and progress in nearly the decade since, the power of negative momentum remains unclear for two key reasons: *generality* and *slow convergence*.

**Generality of negative momentum?** Indeed, it has proven quite difficult to establish global[1] convergence rates for negative momentum for settings beyond linear $\nabla f$ (i.e., beyond bilinear and quadratic objectives $f$) because then the dynamics of the algorithm are non-linear and more complicated. In the min-max literature, the canonical proving ground for algorithms is convex-concave objectives $f$. Competitor algorithms (e.g., GDA with extragradients, optimism, etc.) are all classically known to converge in this setting [23, 36]. In sharp contrast, it remains unknown whether negative momentum can ensure global convergence for this foundational setting.

**Question 1.1** (Convergence for convex-concave $f$)**.** *For convex-concave optimization, is any global convergence result true for GDA with negative momentum?*

Answering this question is essential for understanding the power of negative momentum, because there are many situations where optimization algorithms work well in linear settings but fail beyond.

[1] Of course, local convergence rates can be shown by linearizing the dynamics in order to reduce to the setting where $\nabla f$ is linear [16]. However, such arguments are unable to give non-asymptotic global convergence rates.

An infamous example is momentum in convex optimization. Indeed, Polyak [35] showed in the 1960s that momentum accelerates GD on convex quadratics $f$ (i.e., linear $\nabla f$); however, it remained open for half a century whether momentum makes GD converge at a similarly accelerated rate for non-quadratic convex optimization (i.e., beyond linear $\nabla f$), and recently this was answered in the negative for Polyak's heavy ball method [28] and more generally for a wide range of momentum parameters [20]. Does momentum similarly have fundamental failures for min-max optimization?

**Fast convergence of negative momentum?** Beyond linear $\nabla f$, convergence is known for negative momentum only in the setting of strongly-convex-strong-concave $f$ [49]. This is a standard benchmark setting in the literature that serves as a stepping stone towards general convex-concave $f$. Competitor algorithms (e.g., GDA with extragradients, optimism, etc.) are all known to converge to an $\varepsilon$-optimal solution in $\mathcal{O}(\kappa \log \frac{1}{\varepsilon})$ iterations, where $\kappa$ denotes the condition number [32]. This rate is optimal among arbitrary first-order algorithms [51]. In contrast, negative momentum is believed to be unable to achieve these fast rates: recent work [48] has argued that negative momentum is fundamentally suboptimal in that it requires at least $\Omega(\kappa^{1.5} \log \frac{1}{\varepsilon})$ iterations, which is substantially slower.

**Question 1.2** (Optimal convergence for strongly-convex-strongly-concave $f$)**.** *For strongly-convex-strongly-concave optimization with condition number $\kappa$, can GDA with negative momentum converge faster than $\mathcal{O}(\kappa^{1.5} \log \frac{1}{\varepsilon})$? At the optimal rate of $\mathcal{O}(\kappa \log \frac{1}{\varepsilon})$?*

A potential hope for circumventing the aforementioned lower bound $\Omega(\kappa^{1.5} \log \frac{1}{\varepsilon})$ is that it applies only to negative momentum *without* alternation, i.e., the algorithm (1.2) rather than (1.3). As described above, alternation is necessary for negative momentum to converge for bilinear $f$ [16]; however, it is not necessary for convergence under the strong growth conditions enjoyed by strongly-convex-strongly-concave $f$. A technical challenge for analyzing alternation is that it requires analyzing multi-step progress (at least 2 updates rather than 1), which is particularly difficult for non-linear $\nabla f$ (see also Question 1.2). This challenge has limited prior work on both upper and lower bounds, and therefore the power of negative momentum remains unclear.

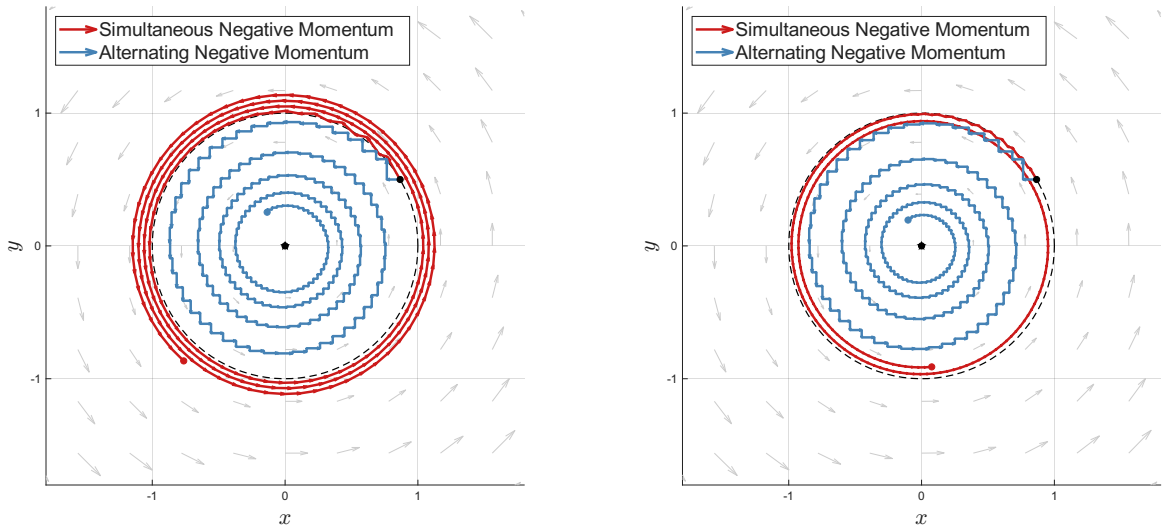
## 1.1 Contributions

This paper resolves both questions in the affirmative. Together, these results show that negative momentum enables GDA to converge substantially faster and more generally than was known possible. These results put negative momentum on more equal footing with competitor algorithms (e.g., extragradients, optimism, etc.) which are well-known to have positive answers to both questions.

**Theorem 1.3** (Informal version of Theorem 3.1)**.** *Suppose $f$ is convex-concave and smooth. With appropriate choices of the stepsize $\eta > 0$ and negative momentum $\beta < 0$, the algorithm (1.3) converges in $\mathcal{O}(\frac{1}{\varepsilon})$ iterations.*

**Theorem 1.4** (Informal version of Theorem 3.2)**.** *Suppose $f$ is $\mu$-strongly-convex-strongly-concave and $L$-smooth. Let $\kappa := L/\mu$ denote the condition number. With appropriate choices of the stepsize $\eta > 0$ and negative momentum $\beta < 0$, the algorithm (1.3) converges in $\mathcal{O}(\kappa \log \frac{1}{\varepsilon})$ iterations.*

Central to both results is the use of alternating updates in $x$ and $y$ rather than simultaneous updates (i.e., algorithm (1.3) rather than (1.2)). As described above, this makes the analysis more challenging as it requires analyzing multi-step progress (at least 2 updates rather than 1). However, as we show, this alternation enables negative momentum to converge in convex-concave settings, and




(a) For convex-concave $f$, alternation enables negative momentum to *converge* (Theorem 1.3).

(b) For strongly-convex-strongly-concave $f$, alternation enables negative momentum to *accelerate* (Theorem 1.4).

FIGURE 1: Negative momentum is more effective with alternating updates (1.3) in $x$ and $y$ than simultaneous updates (1.2). Plotted: 200 iterations on 1-smooth problems $\min_x \max_y f(x,y)$ with unique solution at the origin. Alternating updates are shown for the parameter choices we analyze (stepsize $\eta = 0.2$ and momentum $\beta = -0.5$). Left: trajectories for convex-concave $f(x,y) = xy$. Simultaneous updates shown for $\eta = 0.2$, $\beta = -0.8$; all parameter choices similarly diverge [16]. Right: trajectories for $\mu$-strongly-convex-strongly-concave $f(x,y) = \frac{\mu}{2}x^2 + \sqrt{1-\mu^2}xy - \frac{\mu}{2}y^2$, with $\mu = 0.01$. Simultaneous updates shown for the suggested optimal $\eta = \sqrt{\mu} = 0.1, \beta = \sqrt{\mu} - 1 = -0.9$ from [49].

to accelerate in strongly-convex-strongly-concave settings (overcoming the aforementioned lower bound of [48] which applies only to simultaneous updates). See Fig. 1.

We prove both results via a key progress lemma (Lemma 4.1) which identifies a non-obvious quadratic Lyapunov function under which negative momentum makes significant progress in each full iteration (i.e., after the update of both $x$ and $y$). This lemma applies to both settings.

## 1.2 Related work

**Negative momentum.** As described above, the direct use of (positive) momentum fails to make gradient dynamics converge for min-max optimization. [16] introduced negative momentum as an algorithmic mechanism for min-max problems, proving non-asymptotic global convergence for bilinear objectives and asymptotic local convergence for settings beyond, and showing empirically that this can help in complicated non-convex-non-concave settings like GANs. Their use of negative momentum was inspired by a trend in decreasing momentum values in GAN training. In the years since, a number of works have studied negative momentum, e.g., for strongly-convex-strongly-concave objectives [48, 49], for constrained settings [14], and more broadly as an algorithmic building block [7, 15, 29]. However, despite significant progress over the past decade, basic questions remain about the power of negative momentum, in particular about its generality (Question 1.1) and its speed (Question 1.2). The purpose of this paper is to answer these two questions.

**Asymmetry and alternation.** A key feature of some (but not all) min-max optimization algorithms is asymmetry, meaning updates that distinguish the minimization variable $x$ and maximization variable $y$. This asymmetry is provably beneficial in several contexts, for example accelerating the convergence of GDA in strongly-convex-strongly-concave settings [26, 39, 50], enabling the

convergence of GDA in convex-concave settings [39], and enabling faster algorithms for min-max problems than their associated variational inequalities [38]. In the context of negative momentum, asymmetry is exploited through the alternation of updates of $x$ and $y$ (cf., (1.3) versus (1.2)). This is provably necessary for convergence even for bilinear objectives [16] and therefore also for more general settings such as convex-concave objectives (cf., Question 1.1). For strongly-convex-strongly-concave objectives, alternation is necessary for negative momentum to converge at optimal rates [48], and in this paper we show that it is also sufficient (cf., Question 1.2 and Theorem 1.4).

**Alternative algorithms.** Due to the failure of GDA (1.1) even in simple bilinear settings such as $\min_x \max_y xy$ [23], an extensive literature has been devoted to developing convergent first-order algorithms for min-max optimization. Classic algorithms such as extragradient [23] and optimistic gradient [36] enable GDA to converge for convex-concave objectives. These algorithms augment GDA by moving in the gradient direction at a "lookahead iterate" rather than at the current iterate. While some qualitative connections have been made between optimism and negative momentum [10, 32], connections thus far appear to be primarily syntactic and in particular no formal reductions or relations between theoretical convergence rates are known. Non-asymptotic convergence rates have recently been shown for these optimistic and extragradient algorithms, namely $\mathcal{O}(1/\varepsilon)$ for convex-concave objectives [9, 18, 19, 33] and $\mathcal{O}(\kappa \log 1/\varepsilon)$ for strongly convex-strongly concave objectives [11, 43]. In this paper we show matching convergence rates for negative momentum, thereby putting it on more equal footing with these classical algorithms. For the strongly-convex-strongly-concave setting, this rate is optimal among arbitrary first-order algorithms. For the convex-concave setting, more sophisticated algorithms have recently been shown to accelerate convergence to $\mathcal{O}(1/\sqrt{\varepsilon})$ [27, 42, 46], and it remains an interesting problem if negative momentum can further accelerate on convex-concave objectives. By providing the first global convergence result for negative momentum, Theorem 3.1 opens the door to such refined questions.

**Computer-assisted search for Lyapunov functions.** Our convergence analysis is inspired by the framework of performance estimation problem (PEP), although with several key differences (detailed in § 5). Introduced in [12], PEP casts the problem of finding the worst-case rate of a first-order algorithm over a fixed number of iterations as a semidefinite program. Closest to our work is the variant of PEP which automatically searches for Lyapunov functions certifying descent [28, 40, 44]. These families of ideas enable (partially) automating the design and analysis of optimization algorithms, and have been used to great effect in the past ~15 years in a broad range of optimization settings (see the surveys [13, 41] and references within), including min-max optimization (see for example [18, 19, 24, 25, 26, 37, 39, 46, 47, 50]). We describe how our analysis framework builds upon PEP in detail in §5.

## 2 Preliminaries and notation

We focus on unconstrained convex-concave optimization, a fundamental setting for min-max optimization that is a standard testing ground for algorithms. These problems are of the form

$$\min_{x \in \mathbb{R}^{d_x}} \max_{y \in \mathbb{R}^{d_y}} f(x, y)\,. \tag{2.1}$$

Throughout we assume the existence of a solution, i.e., a point $(x^\star, y^\star)$ satisfying the stationarity condition $\nabla f(x^\star, y^\star) = 0$, or equivalently (since $f$ is convex-concave) satisfying the saddle-point condition $f(x^\star, y) \le f(x^\star, y^\star) \le f(x, y^\star)$ for all $x, y$. For simplicity of exposition, throughout we

consider finite-dimensional spaces of (arbitrary) dimension $d_x$ and $d_y$, although we remark that our results extend to infinite-dimensional Hilbert space $\ell_2$.

Algorithmic results such as fast global convergence are tied to structural assumptions of the objective $f$. We focus on the standard setting of objectives $f$ that are smooth and convex-concave (or strongly-convex-strongly-concave). We briefly recall the definitions of these notions.

**Definition 2.1** (Smooth functions)**.** *A function $g$ is $L$-smooth if its gradient $\nabla g$ is $L$-Lipschitz. That is, $\|\nabla g(a) - \nabla g(b)\| \le L\|a - b\|$ for all $a, b$.*

**Definition 2.2** ((Strongly) convex functions)**.** *A function $g$ is convex if $g(ta + (1-t)b) \le tg(a) + (1-t)g(b)$ for all $t \in [0,1]$ and $a, b$. For $\mu \ge 0$, $g$ is $\mu$-strongly convex if $g(b) - \frac{\mu}{2}\|b\|^2$ is convex.*

**Definition 2.3** ((Strongly) convex-concave functions)**.** *For $\mu \ge 0$, a function $f(x, y)$ is $\mu$-strongly-convex-strongly-concave if $f(\cdot, y)$ is $\mu$-strongly-convex for every $y$, and $f(x, \cdot)$ is $\mu$-strongly-concave for every $x$. If $f$ satisfies this for $\mu = 0$, $f$ is convex-concave.*

Our results do not require the objective $f \in C^2$. However, for intuition, we note that under such an assumption, $f$ being $L$-smooth is equivalent to $\|\nabla^2 f\|_{\mathrm{op}} \le L$, and $f$ being $\mu$-strongly-convex-strongly-concave is equivalent to $\nabla^2_{xx} f \succeq \mu \boldsymbol{I}_{d_x}$ and $-\nabla^2_{yy} f \succeq \mu \boldsymbol{I}_{d_y}$.

In our analysis, the primary way that we exploit these structural properties of smoothness and (strong) convexity-concavity is through the standard "co-coercivity inequality", recalled next.

**Lemma 2.4** (Co-coercivity)**.** *Suppose $g$ is 1-smooth and $\mu$-strongly convex. The co-coercivity*

$$C_g(a, b) := g(a) - g(b) - \nabla g(b)^\top (a - b) - \frac{\mu}{2}\|a - b\|^2 - \frac{1}{2(1-\mu)}\|\nabla g(a) - \nabla g(b) - \mu(a - b)\|^2$$

*satisfies $C_g(a, b) \ge 0$ for any points $a, b$.*

**Notation.** Our notation is standard. We write $\boldsymbol{I}_d$ to denote the identity matrix in dimension $d$, $\otimes$ to denote the Kronecker product, and $\succeq, \preceq$ to denote inequalities in the Löewner order. The notation $\mathcal{O}(\cdot)$ and $\Omega(\cdot)$ refer to upper and lower bounds, respectively, up to universal constants. For shorthand, we often concatenate variables as $z_t := (x_t, y_t)$. All vectors are column vectors.

## 3 Formal statement of main results

In this section, we formally state our convergence results for alternating GDA with negative momentum on convex-concave and strongly-convex-strongly-concave functions.

Our first result answers Question 1.1:

**Theorem 3.1** (Convergence for convex-concave objectives)**.** *Let $f$ be an $L$-smooth, convex-concave function with saddle point $z^* = (x^*, y^*)$. Then for any dimensions $d_x, d_y \in \mathbb{N}$, any initialization $z_0 = (x_0, y_0)$, and any number of iterations $T \in \mathbb{N}$, the iterates of alternating GDA (1.3) with stepsize $\eta = \frac{1}{5L}$ and negative momentum $\beta = -\frac{1}{2}$ satisfy*

$$\frac{1}{T}\sum_{t<T} \|\nabla f(z_t)\|^2 \le \frac{12\|z_0 - z^*\|^2}{\eta^2 T}. \tag{3.1}$$

This result can be rewritten as $\mathbb{E}[\|\nabla f(z_\tau)\|^2] \lesssim \frac{1}{T}$ at a random stopping time $\tau$ chosen uniformly from $0, 1, ..., T-1$. Importantly, this does *not* require taking an average of the iterates $z_t$, a key benefit for practical settings such as training GANs (see the discussions in e.g., [22, 31] ).

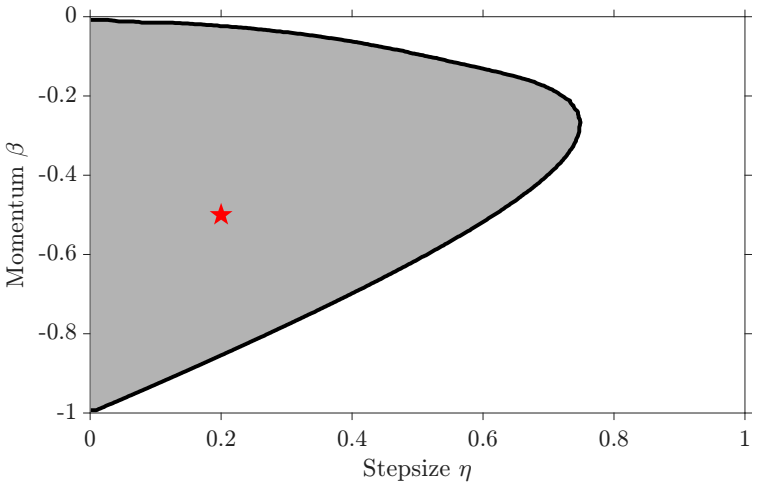



FIGURE 2: The shaded region represents the set of parameters $(\eta, \beta)$ for which it is possible to certify convergence on 1-smooth, convex-concave functions ($\mu = 0$) via the semidefinite programming approach described in §5 (i.e., the parameters for which convergence can be certified by an identity of the form (5.1)). The boundary's roughness is due to numerical instability of SDP solvers (note however that our actual proofs are fully symbolic and rigorous). The red star marks the specific choice $(\eta, \beta) = (1/5, -1/2)$ considered for simplicity in Theorems 3.1 and 3.2.

The rate $\mathcal{O}(1/T)$ in Theorem 3.1 matches popular algorithms such as extragradient [9, 18] and optimistic GDA [9, 19]. An accelerated rate of $\mathcal{O}(1/T^2)$ can be achieved by sophisticated algorithms [46], and it is an interesting question whether negative momentum can similarly lead to acceleration. By providing the first global convergence result for negative momentum, Theorem 3.1 opens the door to such refined questions.

Our second result answers Question 1.2:

**Theorem 3.2** (Convergence for strongly-convex-strongly-concave objectives)**.** *Let $f$ be an $L$-smooth, $\mu$-strongly-convex-strongly-concave function with saddle point $z^\star = (x^\star, y^\star)$. For any dimensions $d_x, d_y \in \mathbb{N}$, any initialization $z_0 = (x_0, y_0)$, and any number of iterations $T$, the iterates of alternating GDA (1.3) with stepsize $\eta = \frac{1}{5L}$ and negative momentum $\beta = -\frac{1}{2}$ satisfy*

$$\|z_T - z^\star\|^2 \le 6(1 - \eta\mu)^T \|z_0 - z^\star\|^2. \tag{3.2}$$

*In particular, $\|z_T - z^\star\|^2 \le \varepsilon \|z_0 - z^\star\|^2$ after $T = \mathcal{O}(\kappa \log \frac{1}{\varepsilon})$ iterations, where $\kappa := L/\mu$ denotes the condition number.*

This rate $\mathcal{O}(\kappa \log \frac{1}{\varepsilon})$ improves over the $\Omega(\kappa^{1.5} \log \frac{1}{\varepsilon})$ lower bound for *simultaneous* GDA with negative momentum [48]. Moreover, this rate is optimal among the class of first-order algorithms [2] and matches classic algorithms such as extragradient and optimistic GDA [33].

**Remark 3.3** (Choice of parameters)**.** *For concreteness and simplicity of exposition, we choose explicit constants for the stepsize $\eta$ and momentum $\beta$ in Theorems 3.1 and 3.2. No effort has been made to optimize constants. The algorithm converges for other parameter choices, see Fig. 2.*

## 4 Progress lemma and its implications

We establish Theorems 3.1 and 3.2 via a progress lemma which identifies a quadratic Lyapunov function $\xi_t^\top \boldsymbol{Q} \xi_t$ under which negative momentum makes significant progress in each iteration. The

state $\xi_t \in \mathbb{R}^{3(d_x+d_y)}$ includes the position, gradient[2], and momentum at time $t$ in both the $x$ and $y$ coordinates:

$$\xi_t := (x_t - x^*,\ \nabla_x f(x_t, y_t),\ v_t,\ y_t - y^*,\ \nabla_y f(x_t, y_t),\ w_t).$$

Here the momentum $(v_t, w_t)$ plays the role of $(x_t - x_{t-1}, y_t - y_{t-1})$. It is initialized to $(v_0, w_0) = (0, 0)$ due to the standard convention of initializing $x_{-1} := x_0$, and $y_{-1} := y_0$.

The matrix $\boldsymbol{Q}$ can be interpreted as a coordinate system which extracts quadratic statistics of the state $\xi_t$ relevant for establishing convergence. We take $\boldsymbol{Q}$ to be block-diagonal of the form

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{Q}_x \otimes \boldsymbol{I}_{d_x} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_y \otimes \boldsymbol{I}_{d_y} \end{bmatrix},$$

where $\boldsymbol{Q}_x, \boldsymbol{Q}_y \in \mathbb{R}^{3\times 3}$ are positive definite.

For simplicity, we state this progress lemma for 1-smooth functions and fixed parameters $\eta = 1/5$ and $\beta = -1/2$. This extends trivially to $L$-smooth functions via rescaling, see Remark 4.2 below.

**Lemma 4.1** (Progress lemma for negative momentum)**.** *Let $\mu \in [0, 1)$. Suppose $f$ is a 1-smooth, $\mu$-strongly-convex-strongly-concave function with saddle point $z^* = (x^*, y^*)$. Consider the state vectors $\xi_t$ generated by running alternating GDA (1.3) with stepsize $\eta = \frac{1}{5}$ and negative momentum $\beta = -\frac{1}{2}$. There exists $\boldsymbol{Q}$ such that for all iterations $t$, it holds that*

$$(1 - \mu/5)\xi_t^\top \boldsymbol{Q}\xi_t - \xi_{t+1}^\top \boldsymbol{Q}\xi_{t+1} \geq (1 - \mu)\|\nabla f(z_t)\|^2 \tag{4.1}$$

*and*

$$50\|z_t - z^*\|^2 \leq \xi_t^\top \boldsymbol{Q}\xi_t \leq 150\|\xi_t\|^2. \tag{4.2}$$

This lemma is written in a convenient way so that it applies to both the strongly-convex-strongly-concave setting ($\mu > 0$) and the convex-concave setting ($\mu = 0$). The key inequality (4.1) quantifies the decrease of the Lyapunov function $\xi_t^\top \boldsymbol{Q}\xi_t$, implying exponential convergence rates when $\mu > 0$ (due to the multiplicative decrease from the contraction) and polynomial rates when $\mu = 0$ (due to the additive decrease from the gradient term). The second inequality (4.2) ensures that the Lyapunov function can be related to quantities such as $\|z_t - z^*\|^2$, so that our final convergence results can be stated in terms of standard interpretable metrics.

**Remark 4.2** (Rescaling $L = 1$ without loss of generality)**.** *Running the algorithm (1.3) with momentum $\beta$ and stepsize $\eta = 1/(5L)$ on an $L$-smooth, $\mu$-strongly-convex-strongly-concave function $f$ generates exactly the same trajectory as running it with stepsize $1/5$ on the rescaled function $\tilde{f} := f/L$, which is 1-smooth and $\tilde{\mu}$-strongly-convex-strongly-concave for $\tilde{\mu} := \mu/L = 1/\kappa$. This is why we state Lemma 4.1 for $L = 1$ without loss of generality.*

### 4.1 Proof of Theorem 3.1 using the progress lemma

By Remark 4.2, it is equivalent to analyze the trajectory on the rescaled function $\tilde{f} = f/L$ using rescaled stepsize $\tilde{\eta} = 1/5$. Note that $\xi_t$ refers to the state of this rescaled trajectory, and thus includes information about $\nabla\tilde{f}(x_t, y_t)$ rather than $\nabla f(x_t, y_t)$.

Setting $\mu = 0$ and telescoping the key inequality (4.1) over $T$ iterations gives:

$$\sum_{t<T} \|\nabla\tilde{f}(z_t)\|^2 \leq \xi_0^\top \boldsymbol{Q}\xi_0 - \xi_T^\top \boldsymbol{Q}\xi_T.$$

[2]Note that the state $\xi_t$ does not include $\nabla_y f(x_{t+1}, y_t)$ which is used in the second part of the update (1.3). This is for simplicity as this concise state $\xi_t$ captures enough information to show the desired convergence.

We drop the term $\xi_T^\top \boldsymbol{Q} \xi_T \geq 0$ and upper bound $\xi_0^\top \boldsymbol{Q} \xi_0 \leq 150 \|\xi_0\|^2 \leq 300 \|z_0 - z^*\|^2$ using property (4.2) and then the fact that $\|\xi_0\|^2 = \|z_0 - z^*\|^2 + \|\nabla \tilde{f}(z_0)\|^2 \leq 2\|z_0 - z^*\|^2$ (by 1-smoothness of $f$ and the standard initialization $z_{-1} = z_0$ which implies $v_0, w_0 = 0$). Plugging in $\nabla \tilde{f} = \nabla f / L$ gives

$$\frac{1}{L^2} \sum_{t<T} \|\nabla f(z_t)\|^2 \leq 300 \|z_0 - z^*\|^2 .$$

Multiplying both sides by $\frac{L^2}{T}$ and recalling the stepsize choice $\eta = \frac{1}{5L}$ completes the proof.

### 4.2 Proof of Theorem 3.2 using the progress lemma

Again by Remark 4.2 we analyze the trajectory on the rescaled function $\tilde{f} := f/L$. The proof of Theorem 3.2 then follows immediately from Lemma 4.1:

$$\|z_T - z^*\|^2 \leq \frac{1}{50} \xi_T^\top \boldsymbol{Q} \xi_T \leq \frac{1}{50} (1 - \eta\mu)^T \xi_0^\top \boldsymbol{Q} \xi_0 \leq 3(1 - \eta\mu)^T \|\xi_0\|^2 \leq 6(1 - \eta\mu)^T \|z_0 - z^*\|^2 .$$

Above, the first and third steps are by the property (4.2) of $\boldsymbol{Q}$. The second step is by the key inequality (4.1) in Lemma 4.1, which implies the 1-step contraction $\xi_{t+1}^\top \boldsymbol{Q} \xi_{t+1} \leq (1 - \tilde{\mu}/5) \xi_t^\top \boldsymbol{Q} \xi_t = (1 - \eta\mu) \xi_t^\top \boldsymbol{Q} \xi_t$ after dropping the gradient term. The final step is because $\|\xi_0\|^2 \leq 2\|z_0 - z^*\|^2$ as argued already in §4.1.

## 5 Proof of progress lemma

In the previous section, we showed that Lemma 4.1 implies our main results; in this section, we prove this key progress lemma.

We prove Lemma 4.1 by establishing an identity of the form

$$(1 - \mu/5) \xi_t^\top \boldsymbol{Q} \xi_t - \xi_{t+1}^\top \boldsymbol{Q} \xi_{t+1} - (1 - \mu) \|\nabla f(z_t)\|^2 = \sum_\alpha \lambda_\alpha M_\alpha + S \tag{5.1}$$

for an explicit matrix $\boldsymbol{Q}$ satisfying (4.2). The left-hand side of this identity is the quantity that we seek to show is non-negative in order to prove the progress inequality (4.1) in Lemma 4.1. The right-hand side consists of two types of non-negative quantities. The first term is the sum of non-negative multipliers $\lambda_\alpha \geq 0$ (that we construct explicitly) multiplied by certain explicit polynomial quantities $M_\alpha \geq 0$ of the iterates (that are non-negative by the smoothness and convexity-concavity properties of $f$). The second term $S$ is a sum-of-squares quadratic polynomial.

### 5.1 Starting point and technical overview

Our starting point for proving (5.1) is a general numerical approach, based on semidefinite programming (SDP), for searching for the quantities in identities of this form [28, 40, 44]. Intuitively, this search is an SDP feasibility problem because there are linear equality constraints and positive semidefinite (PSD) constraints in the relevant variables $\{\lambda_\alpha\}$, $\boldsymbol{Q}$, and $S$. Indeed, the linear equality constraints arise because both sides of the identity (5.1) are quadratic polynomials (in the state space) whose coefficients depend linearly on the decision variables $\{\lambda_\alpha\}$, $\boldsymbol{Q}$, $S$. The PSD constraints arise because $S$ being a sum-of-squares quadratic is equivalent to it being a PSD quadratic form in the state space, and because $\boldsymbol{Q}$ satisfying the two-sided inequality (4.2) is equivalent to two PSD constraints.

However, while this connection to SDP provides a helpful starting point, there are three overarching challenges that we must address for our problem:

**Challenge 1: choice of valid inequalities.** Establishing the identity (5.1) requires first specifying the non-negative quantities $M_\alpha \geq 0$. This amounts to choosing which "valid inequalities" or "proof system" one uses to prove a convergence rate, or equivalently which structural properties of the objective $f$ to exploit. The most popular approach for choosing $M_\alpha$ in min-max optimization settings is based on the classical fact that if $f(x, y)$ is convex-concave, then $(\nabla_x f, -\nabla_y f)$ is a monotone operator; this monotonicity suggests simple quantities $M_\alpha \geq 0$. See for example [37]. However, a fundamental issue is that those quantities $M_\alpha$ are unavoidably oblivious to the asymmetry between $x$ and $y$ intrinsic to convex-concave functions $f(x, y)$. Without this structure, the proof system is too weak to establish the identity (5.1). We overcome this by using a more powerful proof system $\{M_\alpha\}$ that directly stems from the definition of convex-concavity. Details in §5.2.

**Challenge 2: parametric solution.** The SDP approach numerically certifies the identity (5.1), but only for a fixed numerical choice of algorithm parameters (stepsize $\eta$ and momentum $\beta$) and problem parameters (strong convexity $\mu$ and smoothness $L$). However, this is insufficient due to two important issues. First, it is a priori unclear how to find good algorithm parameters because the search for the stepsize $\eta$ and momentum $\beta$ cannot be tractably included in the SDP—indeed this search is in general non-convex, a notorious challenge in the PEP literature more broadly [41]. Second, Lemma 4.1 requires a (symbolic) proof that applies to any value of the problem parameter $\mu$ (one can eliminate $L$ via rescaling, see Remark 4.2). We overcome this by identifying convenient choices of $\eta$ and $\beta$ that enable SDP solutions and convergence proofs that are simple in two key ways. First, they require only a sparse subset of the multipliers $\{\lambda_\alpha\}$, in fact with only 3 distinct values out of potentially 72, which makes it significantly more tractable to identify symbolic solutions. Second, this enables choices of the multipliers $\lambda_\alpha$ and potential function $\boldsymbol{Q}$ that are *independent* of the problem parameter $\mu \geq 0$, making the symbolic verification (described next) significantly simpler. Details in §5.3.

**Challenge 3: rigorous proof.** SDP solvers provide numerical outputs that may satisfy the feasibility constraints only approximately. However, proving Lemma 4.1 requires rigorously certifying both the linear coefficient-matching constraints defining (5.1) as well as the PSD constraints related to (4.2) and the sum-of-squares property for $S$. Rounding numerical SDP solutions to rigorous proofs is a well-documented challenge in the PEP literature, see e.g., [41]. We overcome this by leveraging techniques from the symbolic computational algebra community, for example we use Descartes' rule of signs and Sturm's method in order to certify PSD constraints (the most difficult of these issues for PEP). Details in §5.4.

**Organization of the remainder of the section.** Below we detail how we address these challenges and prove Lemma 4.1. This analysis is inspired by the numerical output of the aforementioned SDP, but we emphasize that our proof is fully rigorous (not numerical) and can be read entirely by itself (the purpose of the above discussion is primarily to provide insight for how this proof was obtained). Specifically, below in §5.2 we specify the choice of valid inequalities $\{M_\alpha\}$; in §5.3 we explicitly construct all other quantities $\{\lambda_\alpha\}$, $\boldsymbol{Q}$, $S$ in the progress identity (5.1); in §5.4 we prove that $S$ satisfies the desired sum-of-squares property; and finally in §5.5 we combine these ingredients to prove Lemma 4.1.

## 5.2 Valid inequalities

Here we specify the non-negative quantities $\{M_\alpha\}$ that we use in the progress identity (5.1). As explained above, this choice of $\{M_\alpha\}$ can be interpreted as the choice of which structural properties

of $f$ (a.k.a. which "valid inequalities") we exploit in our convergence analysis. We use three types of quantities, defined for $i, j, k, l \in \{t, t+1, *\}$:

- **Smoothness.** Recall that $f$ being 1-smooth amounts to $\nabla f$ being 1-Lipschitz (see Definition 2.1). Thus the following quantities are clearly non-negative:

$$M_{\text{smooth}}(x_i, y_j, x_k, y_l) := \|(x_i, y_j) - (x_k, y_l)\|^2 - \|\nabla f(x_i, y_j) - \nabla f(x_k, y_l)\|^2.$$

- **Convexity.** Since $f(\cdot, \cdot)$ is $\mu$-strongly-convex-strongly-concave and 1-smooth, the restricted function $f(\cdot, y_k)$ is $\mu$-strongly convex and 1-smooth for any $y_k$. By Lemma 2.4, this implies non-negativity of the associated co-coercivities:

$$M_{\text{convex}}(x_i, x_j, y_k) := C_{f(\cdot, y_k)}(x_i, x_j)\,.$$

- **Concavity.** Analogously, the restricted function $f(x_k, \cdot)$ is $\mu$-strongly-concave and 1-smooth, hence the co-coercivities for $-f(x_k, \cdot)$ are non-negative:

$$M_{\text{concave}}(y_i, y_j, x_k) := C_{-f(x_k, \cdot)}(y_i, y_j)\,.$$

Note that all three quantities are polynomials which are linear (or zero) in the function values and quadratic in the iterates and gradients. Indeed, by expanding the definition of the co-coercivities (Lemma 2.4), the latter two quantities can be written explicitly as

$$\begin{aligned}
M_{\text{convex}}(x_i, x_j, y_k) &= f(x_i, y_k) - f(x_j, y_k) - \nabla_x f(x_j, y_k)^\top (x_i - x_j) \\
&\quad - \frac{\mu}{2}\|x_i - x_j\|^2 - \frac{1}{2(1-\mu)}\|\nabla_x f(x_i, y_k) - \nabla_x f(x_j, y_k) - \mu(x_i - x_j)\|^2\,, \\
M_{\text{concave}}(y_i, y_j, x_k) &= f(x_k, y_j) - f(x_k, y_i) + \nabla_y f(x_k, y_j)^\top (y_i - y_j) \\
&\quad - \frac{\mu}{2}\|y_i - y_j\|^2 - \frac{1}{2(1-\mu)}\|\nabla_y f(x_k, y_j) - \nabla_y f(x_k, y_i) - \mu(y_i - y_j)\|^2\,.
\end{aligned}$$

These non-negative quantities $M_\alpha$ are also used to prove convergence rates in [26, 39], albeit for different algorithms and purposes. We highlight two crucial features of this choice:

1. **Convexity-concavity rather than monotonicity.** More standard is not to use $M_{\text{convex}}$ and $M_{\text{concave}}$, instead to use only the simpler $M_{\text{monotone}}(z_i, z_j) := \langle F(z_i) - F(z_j), z_i - z_j\rangle$ where $z_i := (x_i, y_i)$, $z_j := (x_j, y_j)$, and $F := (\nabla_x f, -\nabla_y f)$ is the saddle-point operator associated to $f$. The motivation is that convexity-concavity of $f$ implies monotonicity of the operator $F$, which ensures non-negativity of $M_{\text{monotone}}$. See for example [37, 49]. However, convexity-concavity of $f$ is a strictly stronger property than monotonicity of $F$ when enforced only at a finite collection of points (as done here), which is why $M_{\text{convex}}$ and $M_{\text{concave}}$ are strictly more powerful inequalities than $M_{\text{monotone}}$ and enable proving much faster convergence rates.

2. **Gridded inequalities.** More standard is to use valid inequalities which depend on the iterates only through $(x_i, y_i)$ for $i \in \{t, t+1, *\}$. In contrast, we consider valid inequalities of the above three types for $i, j, k, l \in \{t, t+1, *\}$. This amounts to enforcing these inequalities on a "grid" of (concatenated) points, many of which are not visited by the algorithm yet are helpful to proving convergence analyses. For example, $M_{\text{concave}}(y_t, y_{t+1}, x_t)$ includes information about $f$ at $(x_t, y_{t+1})$, and $M_{\text{concave}}(y^*, y_t, x_{t+1})$ includes information about $f$ at $(x_{t+1}, y^*)$. This gridding enables exploiting global structural properties of $f$ that the aforementioned standard approach cannot.

Both features enable proving stronger rates, and the combination is needed to prove Lemma 4.1.

### 5.3 Construction of quantities in progress identity (5.1)

Here we explicitly construct the quantities $\boldsymbol{Q}$, $\boldsymbol{S}$, and $\{\lambda_\alpha\}$ that define the progress identity (5.1).

Letting $\otimes$ denote the Kronecker product, we define $\boldsymbol{Q}$ as

$$\boldsymbol{Q} := \begin{bmatrix} \boldsymbol{Q}_x \otimes \boldsymbol{I}_{d_x} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_y \otimes \boldsymbol{I}_{d_y} \end{bmatrix}, \quad \boldsymbol{Q}_x := \begin{bmatrix} 120 & -\frac{81}{10} & -40 \\ -\frac{81}{10} & 1 & 4 \\ -40 & 4 & 40 \end{bmatrix}, \quad \boldsymbol{Q}_y := \begin{bmatrix} 120 & 0 & -40 \\ 0 & 1 & -4 \\ -40 & -4 & 40 \end{bmatrix}. \tag{5.2}$$

Note that $\boldsymbol{Q}$ satisfies (4.2) since a straightforward eigenvalue calculation shows that $50\boldsymbol{E}_{11} \preceq \boldsymbol{Q}_x, \boldsymbol{Q}_y \preceq 150\boldsymbol{I}$, where $\boldsymbol{E}_{11}$ denotes the $3 \times 3$ matrix with first entry 1 and all other entries 0.

Next, we define the multipliers $\{\lambda_\alpha\}$. Although the proof system in §5.2 allows for arbitrary non-negative combinations $\sum_\alpha \lambda_\alpha M_\alpha$ of all 72 valid inequalities $M_\alpha$, an appealing aspect of our analysis is that we use only a small subset of these quantities to prove the desired identity. Explicitly, letting $\lambda_1 = 4, \lambda_2 = \frac{79}{5}, \lambda_3 = \frac{81}{5}$, the non-negative combination of valid inequalities we use is

$$\begin{aligned} \sum_\alpha \lambda_\alpha M_\alpha = {} & \lambda_1 M_{\text{smooth}}(x_t, y_t, x_{t+1}, y_{t+1}) \\ & + \lambda_2 \left( M_{\text{convex}}(x_t, x_{t+1}, y_t) + M_{\text{convex}}(x^*, x_t, y_t) + M_{\text{concave}}(y_t, y^*, x^*) \right) \\ & + \lambda_3 \left( M_{\text{convex}}(x^*, x_{t+1}, y_{t+1}) + M_{\text{concave}}(y_{t+1}, y^*, x^*) + M_{\text{concave}}(y_t, y_{t+1}, x_t) \right) \\ & + (\lambda_2 + \lambda_3) \left( M_{\text{convex}}(x_{t+1}, x^*, y^*) + M_{\text{concave}}(y^*, y_t, x_{t+1}) \right) \end{aligned}$$

Finally, we define $S$ as the residual in (5.1) so that the identity holds by construction:

$$S := (1 - \mu/5)\xi_t^\top \boldsymbol{Q} \xi_t - \xi_{t+1}^\top \boldsymbol{Q} \xi_{t+1} - (1-\mu)\|\nabla f(z_t)\|^2 - \sum_\alpha \lambda_\alpha M_\alpha \,. \tag{5.3}$$

It remains to prove that this $S$ satisfies the desired property; we do this next.

### 5.4 Proof that $S$ is sum-of-squares

As described above, in order to use the progress identity (5.1) to prove Lemma 4.1, we need to show that $S$ is non-negative. We accomplish this by showing that $S$ is a sum of squares for any $\mu \in [0, 1)$. We begin by simplifying the definition of $S$ in (5.3). An algebraically tedious but conceptually straightforward calculation shows that $S$ is a quadratic form which can be expressed as

$$S = \frac{1}{2(1-\mu)} \Xi_t^\top \begin{bmatrix} \boldsymbol{S}_x \otimes \boldsymbol{I}_{d_x} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S}_y \otimes \boldsymbol{I}_{d_y} \end{bmatrix} \Xi_t \,. \tag{5.4}$$

The matrices $\boldsymbol{S}_x, \boldsymbol{S}_y$ are complicated but fully explicit (see Appendix A for the closed-form) and have entries which depend quadratically in $\mu$. The vector $\Xi_t$ is an extended state space that incorporates information about $\nabla f$ at certain additional points that are not iterates of the algorithm:

$$\begin{aligned} \Xi_t := {} & (x_t - x^*, \nabla_x f(x^*, y_t), \nabla_x f(x^*, y_{t+1}), \nabla_x f(x_{t+1}, y^*), \nabla_x f(x_t, y_t), \nabla_x f(x_{t+1}, y_t), \nabla_x f(x_{t+1}, y_{t+1}), v_t, \\ & y_t - y^*, \nabla_y f(x^*, y_t), \nabla_y f(x^*, y_{t+1}), \nabla_y f(x_{t+1}, y^*), \nabla_y f(x_t, y_t), \nabla_y f(x_{t+1}, y_t), \nabla_y f(x_{t+1}, y_{t+1}), w_t) \,. \end{aligned}$$

By virtue of the expression (5.4), the following lemma suffices to show that $(1-\mu)S$ is a sum-of-squares polynomial (and therefore $S$ is non-negative) for any fixed value of $\mu \in [0, 1)$.

**Lemma 5.1.** *For any $\mu \in [0, 1)$, it holds that $\boldsymbol{S}_x, \boldsymbol{S}_y \succeq 0$.*

For brevity, we sketch the main idea of the proof here and defer full details to Appendix A. We prove Lemma 5.1 by showing that for all $\mu \in [0, 1)$, the characteristic polynomials of $\boldsymbol{S}_x, \boldsymbol{S}_y$ have only non-negative roots. We establish this via Descartes' rule of signs: it suffices to show that these characteristic polynomials have coefficients which alternate in sign. Each coefficient is itself a polynomial in $\mu$, so we must verify the alternating sign pattern holds for all $\mu \in [0, 1)$. This can be rigorously proved by first checking this pattern for $\mu = 0$ and then confirming, using standard symbolic algebra techniques such as Sturm's method, that the relevant coefficient polynomials have no roots in $(0, 1)$ and thus never change sign.

### 5.5 Concluding the proof of Lemma 4.1

Consider $\{M_\alpha\}$ as defined in §5.2, and consider $\{\lambda_\alpha\}$, $\boldsymbol{Q}$, and $S$ as defined in §5.3. By construction of $S$, the identity (5.1) holds. Next, note that $\lambda_\alpha$, $M_\alpha$, and $S$ are all non-negative (the former two by construction, and the latter by Lemma 5.1). Thus the right-hand side of (5.1) is non-negative, which implies the desired progress bound (4.1). Finally, the property (4.2) of $\boldsymbol{Q}$ is immediate from the construction of $\boldsymbol{Q}$ in §5.3.

# A Deferred details from §5.4

Here we provide the remaining details for §5.4 and in particular prove Lemma 5.1. See §5.4 for an overview of the proof strategy.

## A.1 Explicit expressions

We begin by explicitly stating the coefficient matrices $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$ in the factorization (5.4) of $S$:

$$\boldsymbol{S}_x = \begin{bmatrix} 16\mu+48\mu^2 & \frac{79}{5}\mu & \frac{81}{5}\mu & -32\mu & -\frac{111}{5}\mu-\frac{81}{25}\mu^2 & 0 & -\frac{81}{5}\mu & -\frac{81}{10}\mu-16\mu^2 \\ \frac{79}{5}\mu & \frac{79}{5} & 0 & 0 & -\frac{79}{5} & 0 & 0 & 0 \\ \frac{81}{5}\mu & 0 & \frac{81}{5} & 0 & -\frac{81}{25}\mu & 0 & -\frac{81}{5} & -\frac{81}{10}\mu \\ -32\mu & 0 & 0 & 32 & \frac{32}{5}\mu & 0 & 0 & 16\mu \\ -\frac{111}{5}\mu-\frac{81}{25}\mu^2 & -\frac{79}{5} & -\frac{81}{25}\mu & \frac{32}{5}\mu & \frac{822}{25}-\frac{86}{25}\mu-\frac{8}{5}\mu^2 & -\frac{316}{25} & -\frac{32}{5}+\frac{241}{25}\mu & -\frac{44}{5}+\frac{57}{10}\mu+\frac{8}{5}\mu^2 \\ 0 & 0 & 0 & 0 & -\frac{316}{25} & \frac{79}{5} & 0 & \frac{79}{10} \\ -\frac{81}{5}\mu & 0 & -\frac{81}{5} & 0 & -\frac{32}{5}+\frac{241}{25}\mu & 0 & \frac{111}{5}-6\mu & 4+\frac{41}{10}\mu \\ -\frac{81}{10}\mu-16\mu^2 & 0 & -\frac{81}{10}\mu & 16\mu & -\frac{44}{5}+\frac{57}{10}\mu+\frac{8}{5}\mu^2 & \frac{79}{10} & 4+\frac{41}{10}\mu & 38-38\mu+16\mu^2 \end{bmatrix}$$

$$\boldsymbol{S}_y = \begin{bmatrix} 16\mu+48\mu^2 & \frac{79}{5}\mu & \frac{81}{5}\mu & -32\mu & 0 & \frac{881}{25}\mu & 0 & \frac{79}{10}\mu-16\mu^2 \\ \frac{79}{5}\mu & \frac{79}{5} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{81}{5}\mu & 0 & \frac{81}{5} & 0 & 0 & \frac{81}{25}\mu & 0 & -\frac{81}{10}\mu \\ -32\mu & 0 & 0 & 32 & 0 & -32 & 0 & 0 \\ 0 & 0 & 0 & 0 & 8-\frac{32}{5}\mu-\frac{8}{5}\mu^2 & 0 & -8+8\mu & -8+\frac{48}{5}\mu-\frac{8}{5}\mu^2 \\ \frac{881}{25}\mu & 0 & \frac{81}{25}\mu & -32 & 0 & \frac{1037}{25}+\frac{192}{125}\mu & -\frac{284}{25}-\frac{8}{5}\mu & \frac{84}{5}-\frac{597}{50}\mu \\ 0 & 0 & 0 & 0 & -8+8\mu & -\frac{284}{25}-\frac{8}{5}\mu & \frac{111}{5}-6\mu & -\frac{121}{10}+4\mu \\ \frac{79}{10}\mu-16\mu^2 & 0 & -\frac{81}{10}\mu & 0 & -8+\frac{48}{5}\mu-\frac{8}{5}\mu^2 & \frac{84}{5}-\frac{597}{50}\mu & -\frac{121}{10}+4\mu & 38-\frac{459}{10}\mu+16\mu^2 \end{bmatrix}$$

In the proof of Lemma 5.1 below, we make use of the characteristic polynomials $p_x(\zeta;\mu) := \det(\boldsymbol{S}_x - \zeta\boldsymbol{I})$ and $p_y(\zeta;\mu) := \det(\boldsymbol{S}_y - \zeta\boldsymbol{I})$ of these matrices. Since each entry of $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$ is a polynomial in $\mu$, these characteristic polynomials are bivariate in $\zeta, \mu$ and can be written as

$$p_x(\zeta;\mu) = \sum_{j=0}^{8} c_{x,j}(\mu)\zeta^j, \qquad p_y(\zeta;\mu) = \sum_{j=0}^{8} c_{y,j}(\mu)\zeta^j, \tag{A.1}$$

where $c_{x,j}(\mu)$ and $c_{y,j}(\mu)$ are polynomials of $\mu$. These coefficients can be computed explicitly via an algebraically tedious but conceptually simple calculation. We state these coefficients below in a concatenated (matrix) form to make it easier to visually see the sign changes, which is the key way in which we use these coefficients below. For $i, j \geq 0$, the coefficient of $\mu^i$ in $c_{x,j}(\mu)$ is the $(i+1, j+1)$-th entry of $\boldsymbol{C}_x$. Similarly for $\boldsymbol{C}_y$.

$$\boldsymbol{C}_x = \begin{bmatrix} 0 & -\frac{3016949328}{3125} & \frac{3925293412889}{31250} & -\frac{315920718241}{6250} & \frac{410082261313}{62500} & -\frac{4859926311}{12500} & \frac{29183151}{2500} & -\frac{4322}{25} & 1 \\ \frac{48271189248}{3125} & -\frac{32554037377816}{15625} & \frac{19024674172661}{31250} & -\frac{116858962492}{3125} & -\frac{4944502477}{6250} & \frac{78866357}{625} & -\frac{444266}{125} & \frac{786}{25} & 0 \\ \frac{118119041994672}{390625} & \frac{218825704369549}{31250} & -\frac{1628209078050351}{781250} & \frac{24840599510927}{156250} & -\frac{20682996319}{12500} & -\frac{647480389}{3125} & \frac{4380333}{625} & -\frac{312}{5} & 0 \\ -\frac{570636561473136}{390625} & -\frac{6749359461879353}{781250} & \frac{794651282514111}{390625} & -\frac{14880446763889}{156250} & -\frac{36988061198}{15625} & \frac{48458491}{250} & -\frac{318582}{125} & 0 & 0 \\ \frac{980677852825824}{390625} & \frac{1863197504635273}{390625} & -\frac{144439831610443}{156250} & \frac{5736968309049}{156250} & \frac{26987184513}{31250} & -\frac{315937633}{6250} & \frac{247839}{625} & 0 & 0 \\ -\frac{789913089425376}{390625} & -\frac{95808733158877}{78125} & \frac{226228514254623}{781250} & -\frac{1259177102927}{78125} & \frac{2732915364}{15625} & \frac{1390428}{625} & 0 & 0 & 0 \\ \frac{293540264356464}{390625} & \frac{152705311358489}{781250} & -\frac{42651897368459}{781250} & \frac{609107362377}{156250} & -\frac{3267679759}{31250} & \frac{590096}{625} & 0 & 0 & 0 \\ -\frac{37821406934448}{390625} & -\frac{13840030176837}{781250} & \frac{377986454943}{78125} & -\frac{9381626397}{31250} & \frac{3540576}{625} & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{C}_y = \begin{bmatrix} 0 & -\frac{1288809792}{15625} & \frac{60597876882}{15625} & -\frac{246249337657}{12500} & \frac{246345620709}{62500} & -\frac{754682859}{2500} & \frac{26859791}{2500} & -\frac{4342}{25} & 1 \\ \frac{20620956672}{15625} & -\frac{5264240621856}{78125} & \frac{28547368182704}{78125} & -\frac{6613389982081}{156250} & \frac{4107920378}{15625} & \frac{15227937}{125} & -\frac{2794858}{625} & \frac{10191}{250} & 0 \\ \frac{6278652931968}{78125} & -\frac{240098071662172}{390625} & -\frac{652932152553913}{781250} & \frac{32815526836777}{312500} & -\frac{29665687223}{15625} & -\frac{1048066121}{6250} & \frac{16482201}{2500} & -\frac{312}{5} & 0 \\ -\frac{59961927583872}{390625} & \frac{1112181291312788}{390625} & \frac{449869267325923}{781250} & -\frac{9937608117483}{156250} & -\frac{10725965407}{15625} & \frac{450804572}{3125} & -\frac{1439996}{625} & 0 & 0 \\ -\frac{758510810673408}{1953125} & -\frac{8360973992985416}{1953125} & -\frac{2871838767839}{31250} & \frac{974565108823}{31250} & -\frac{8239681301}{62500} & -\frac{96310742}{3125} & \frac{10176}{25} & 0 & 0 \\ \frac{2827722585238272}{1953125} & \frac{5710819702531544}{1953125} & -\frac{9205535476859}{156250} & -\frac{1972122788511}{156250} & \frac{4290414752}{15625} & \frac{2966208}{3125} & 0 & 0 & 0 \\ -\frac{3367757488007808}{1953125} & -\frac{1801214735526316}{1953125} & \frac{3957829549436}{78125} & \frac{6639541174}{3125} & -\frac{1620729152}{15625} & \frac{23552}{25} & 0 & 0 & 0 \\ \frac{1850723490483072}{1953125} & \frac{235934676676004}{1953125} & -\frac{4122192150876}{390625} & -\frac{4024869248}{78125} & \frac{13142016}{3125} & 0 & 0 & 0 & 0 \\ -\frac{437038196779008}{1953125} & -\frac{14342187792576}{1953125} & \frac{547701173056}{390625} & -\frac{327867392}{15625} & 0 & 0 & 0 & 0 & 0 \\ \frac{5025222955008}{390625} & \frac{225511122816}{390625} & -\frac{927148032}{15625} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

### A.2 Proof of Lemma 5.1

Let $p_x(\zeta;\mu) := \det(\boldsymbol{S}_x - \zeta\boldsymbol{I})$ and $p_y(\zeta;\mu) := \det(\boldsymbol{S}_y - \zeta\boldsymbol{I})$ denote the characteristic polynomials of $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$, respectively. (See (A.1) for their explicit expressions in terms of coefficients $c_{x,j}(\mu)$ and $c_{y,j}(\mu)$.) Since $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$ are symmetric, their eigenvalues are real, hence all roots of $p_x(\cdot;\mu)$ and $p_y(\cdot;\mu)$ are real for all $\mu$. To prove the lemma, we show that these real roots are in fact non-negative. By Descartes' rule of signs[3], it suffices to show that for each $\mu \in [0,1)$, the non-zero coefficients of $p_x(\zeta;\mu)$ and $p_y(\zeta;\mu)$ alternate in sign.

First, for $\mu = 0$, Descartes' rule of signs holds by inspection of $c_{x,j}(0)$ and $c_{y,j}(0)$. See the first row of $\boldsymbol{C}_x$ and $\boldsymbol{C}_y$ in §A.1 above.

Next, to apply Descartes' rule of signs for $\mu > 0$, we first deal with a slight nuance: the coefficients $c_{x,0}$ and $c_{y,0}$ vanish at $\mu = 0$ (see the top-left entries of $\boldsymbol{C}_x$ and $\boldsymbol{C}_y$). To ensure that the alternating sign pattern persists for small positive $\mu$, observe that in both $c_{x,0}$ and $c_{y,0}$, the linear coefficient of $\mu$ is positive (see the (2,1)-th entries of $\boldsymbol{C}_x$ and $\boldsymbol{C}_y$), which in particular is the opposite of the sign of the constant term in $c_{x,1}$ and $c_{y,1}$ (see the (1,2)-th entries). All other coefficients are already nonzero at $\mu = 0$, hence by continuity their signs remain unchanged for sufficiently small $\mu > 0$, and thus the alternating sign pattern persists for sufficiently small $\mu > 0$.

Now we are ready to apply Descartes' rule of signs simultaneously for all $\mu \in (0,1)$. To do this, it suffices to show that $c_{x,j}(\mu)$ and $c_{y,j}(\mu)$ do not change sign on the interval $\mu \in (0,1)$. That is, it suffices to show that these coefficient polynomials $c_{x,j}(\mu)$ and $c_{y,j}(\mu)$ have no roots on the interval. This can be rigorously proven using standard techniques from symbolic computer algebra such as Sturm's Theorem (see e.g. [3, Theorem 2.50]). For brevity of exposition and the convenience of the reader, we provide a short Mathematica script that validates this in a rigorous symbolic manner [1].

## References


[1] Negative momentum for convex-concave optimization – verification of identities computer algebra script. https://jasonaltschuler.github.io/NegativeMomentum.

[2] Waïss Azizian, Ioannis Mitliagkas, Simon Lacoste-Julien, and Gauthier Gidel. A tight and unified analysis of gradient-based methods for a whole spectrum of differentiable games. In *International Conference on Artificial Intelligence and Statistics*, 2020.

[3] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in real algebraic geometry*. Springer, 2006.

[4] Aharon Ben-Tal, Arkadi Nemirovski, and Laurent El Ghaoui. *Robust optimization*. Princeton University Press, 2009.


[3] We recall here the relevant version of Descartes' rule of signs (see e.g. [3, Theorem 2.33]). Let $q(\zeta) = \sum_{k=0}^{n} c_k \zeta^k$ have real coefficients and real roots. Suppose that for some $m \geq 0$, the low-order coefficients $c_k = 0$ for all $k < m$, and the high-order coefficients $c_m, c_{m+1}, \ldots, c_n$ are non-zero and alternate in sign. Then all roots of $q$ are non-negative.

[5] Michele Benzi, Gene H Golub, and Jörg Liesen. Numerical solution of saddle point problems. *Acta Numerica*, 14:1–137, 2005.

[6] Dimitri P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 2014.

[7] Aleksandr Beznosikov, Aibek Alanov, Dmitry Kovalev, Martin Takáč, and Alexander Gasnikov. On scaled methods for saddle point problems. *Preprint at arXiv:2206.08303*, 2022.

[8] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[9] Yang Cai, Argyris Oikonomou, and Weiqiang Zheng. Tight last-iterate convergence of the extragradient and the optimistic gradient descent-ascent algorithm for constrained monotone variational inequalities. *Advances in Neural Information Processing Systems*, 2022.

[10] Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. *Advances in Neural Information Processing Systems*, 2018.

[11] Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with optimism. In *International Conference on Learning Representations*, 2018.

[12] Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, 145(1):451–482, 2014.

[13] Alexandre d'Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. *Foundations and Trends® in Optimization*, 5(1-2):1–245, 2021.

[14] Zijian Fang, Zongkai Liu, Chao Yu, and Chaohao Hu. Rapid learning in constrained minimax games with negative momentum. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 16541–16549, 2025.

[15] Yi Feng, Kaito Fujii, Stratis Skoulakis, Xiao Wang, and Volkan Cevher. Continuous-time analysis of heavy ball momentum in min-max games. In *International Conference on Machine Learning*, pages 16670–16710. PMLR, 2025.

[16] Gauthier Gidel, Reyhane Askari Hemmat, Mohammad Pezeshki, Rémi Le Priol, Gabriel Huang, Simon Lacoste-Julien, and Ioannis Mitliagkas. Negative momentum for improved game dynamics. In *International Conference on Artificial Intelligence and Statistics*, 2019.

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

[18] Eduard Gorbunov, Nicolas Loizou, and Gauthier 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*, 2022.

[19] Eduard Gorbunov, Adrien Taylor, and Gauthier Gidel. Last-iterate convergence of optimistic gradient method for monotone variational inequalities. *Advances in Neural Information Processing Systems*, 35, 2022.

[20] Baptiste Goujaud, Adrien Taylor, and Aymeric Dieuleveut. Provable non-accelerations of the heavy-ball method. *Mathematical Programming*, pages 1–59, 2025.

[21] Martin Hast, Karl Johan Åström, Bo Bernhardsson, and Stephen Boyd. PID design by convex-concave optimization. In *European Control Conference*, pages 4460–4465. IEEE, 2013.

[22] Ya-Ping Hsieh, Panayotis Mertikopoulos, and Volkan Cevher. The limits of min-max optimization algorithms: Convergence to spurious non-critical sets. In *International Conference on Machine Learning*, pages 4337–4348. PMLR, 2021.

[23] Galina M Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

[24] Valery Krivchenko, Alexander Gasnikov, and Dmitry Kovalev. Strengthening the finite characterizations of smooth min-max games. *arXiv preprint arXiv:2603.17053*, 2026.

[25] Valery O Krivchenko, Alexander V Gasnikov, and Dmitry A Kovalev. Convex-concave interpolation and application of pep to the bilinear-coupled saddle point problem. *Russian Journal of Nonlinear Dynamics*, 20(5):875–893, 2024.

[26] Jaewook Lee, Hanseul Cho, and Chulhee Yun. Fundamental benefit of alternating updates in minimax optimization. *Preprint at arXiv:2402.10475*, 2024.

[27] Sucheol Lee and Donghwan Kim. Fast extra gradient methods for smooth structured nonconvex-nonconcave minimax problems. *Advances in Neural Information Processing Systems*, 34:22588–22600, 2021.

[28] Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016.

[29] Jonathan P Lorraine, David Acuna, Paul Vicol, and David Duvenaud. Complex momentum for optimization in games. In *International Conference on Artificial Intelligence and Statistics*, pages 7742–7765. PMLR, 2022.

[30] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[31] Panayotis Mertikopoulos, Bruno Lecouat, Houssam Zenati, Chuan-Sheng Foo, Vijay Chandrasekhar, and Georgios Piliouras. Optimistic mirror descent in saddle-point problems: Going the extra (gradient) mile. In *International Conference on Learning Representations*, pages 1–23, 2019.

[32] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. In *International Conference on Artificial Intelligence and Statistics*, 2020.

[33] Aryan Mokhtari, Asuman E Ozdaglar, and Sarath Pattathil. Convergence rate of $\mathcal{O}(1/k)$ for optimistic gradient and extragradient methods in smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 30(4):3230–3251, 2020.

[34] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. *Doklady Akademii Nauk SSSR*, 269(3):543–547, 1983.

[35] Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[36] 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):845–848, 1980. ISSN 1573-8876.

[37] Ernest K Ryu, Adrien B Taylor, Carolina Bergeling, and Pontus Giselsson. Operator splitting performance estimation: Tight contraction factors and optimal parameter selection. *SIAM Journal on Optimization*, 30(3):2251–2271, 2020.

[38] Henry Shugart and Jason M Altschuler. Min-max optimization is strictly easier than variational inequalities. *Preprint at arXiv:2511.03052*, 2025.

[39] Henry Shugart and Jason M. Altschuler. Negative Stepsizes Make Gradient-Descent-Ascent Converge. *Preprint at arXiv:2505.01423*, 2025.

[40] Adrien Taylor, Bryan Van Scoy, and Laurent Lessard. Lyapunov functions for first-order methods: Tight automated convergence guarantees. In *International Conference on Machine Learning*, pages 4897–4906. PMLR, 2018.

[41] Adrien B. Taylor. Towards principled and systematic approaches to the analysis and design of optimization algorithms. *PSL Research University*, 2024. Habilitation à diriger des recherches.

[42] Quoc Tran-Dinh and Yang Luo. Halpern-type accelerated and splitting algorithms for monotone inclusions. *Preprint at arXiv:2110.08150*, 2021.

[43] Paul Tseng. On linear convergence of iterative methods for the variational inequality problem. *Journal of Computational and Applied Mathematics*, 60(1):237–252, 1995.

[44] Manu Upadhyaya, Sebastian Banert, Adrien B Taylor, and Pontus Giselsson. Automated tight Lyapunov analysis for first-order methods. *Mathematical Programming*, 209(1):133–170, 2025.

[45] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ, 1947.

[46] TaeHo Yoon and Ernest 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*, 2021.

[47] Moslem Zamani, Hadi Abbaszadehpeivasti, and Etienne de Klerk. Convergence rate analysis of the gradient descent–ascent method for convex–concave saddle-point problems. *Optimization Methods and Software*, 39(5):967–989, 2024.

[48] Guodong Zhang and Yuanhao Wang. On the suboptimality of negative momentum for minimax optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 2098–2106, 2021.

[49] Guodong Zhang, Xuchan Bao, Laurent Lessard, and Roger Grosse. A unified analysis of first-order methods for smooth games via integral quadratic constraints. *Journal of Machine Learning Research*, 22(103):1–39, 2021.

[50] Guodong Zhang, Yuanhao Wang, Laurent Lessard, and Roger B. Grosse. Near-optimal local convergence of alternating gradient descent-ascent for minimax optimization. In *International Conference on Artificial Intelligence and Statistics*, 2022.

[51] Junyu Zhang, Mingyi Hong, and Shuzhong Zhang. On lower iteration complexity bounds for the convex concave saddle point problems. *Mathematical Programming*, 194(1–2):901–935, 2022.