Adaptive Delayed-Update Cyclic Algorithm for Variational Inequalities

Yi Wei*
University of Washington
ywei32@cs.washington.edu

Xufeng Cai
University of Wisconsin–Madison
xcai74@wisc.edu

Jelena Diakonikolas
University of Wisconsin–Madison
jelena@cs.wisc.edu

Abstract

Cyclic block coordinate methods are widely used in practice for their simplicity and strong empirical performance. Yet, their theoretical behavior is challenging to explain, and setting their step sizes—beyond classical coordinate descent for minimization—requires careful tuning or line-search machinery. In this work, we develop ADUCA (Adaptive Delayed-Update Cyclic Algorithm), a cyclic algorithm addressing a broad class of Minty variational inequalities with monotone Lipschitz operators. ADUCA is parameter-free and locally adaptive: it requires no global or block-wise Lipschitz constants, it uses no per-epoch line search, and it adapts to local problem geometry. A key feature of the algorithm is using operator information delayed by a full cycle, which makes the algorithm compatible with parallel and distributed implementations, and attractive due to weakened synchronization requirements across blocks. We prove that ADUCA attains (near) optimal global oracle complexity as a function of target error $\epsilon > 0$, scaling with $1/\epsilon$ for monotone operators, or with $\log^2(1/\epsilon)$ for operators that are strongly monotone.

1 Introduction

Cyclic block coordinate methods are a classical family of first-order algorithms that sweep through and update each coordinate block once per cycle, using a deterministic (or possibly reshuffled) order. Their simplicity and strong empirical performance have led to wide adoption, from Kaczmarz- and Osborne-type procedures to modern sparse learning solvers [19, 22, 34, 39]. Yet their theory lags behind that of randomized coordinate methods and full-vector updates: even in convex minimization, cyclic methods are difficult to analyze and typically admit weaker worst-case guarantees [4, 44, 47]. Effective step-size selection further demands careful tuning or line search [4, 7, 21, 30, 44], and the situation worsens for min-max problems and variational inequalities (VIs), where exact cyclic descent fails to converge in general; see, e.g., the discussion in [44].

We study cyclic algorithms for generalized Minty variational inequalities (GMVIs) with monotone, locally block-wise Lipschitz operators and block-separable proximal terms; the formal problem statement appears in Section 2. Our goal is to retain the practical simplicity of cyclic block updates while avoiding global step size tuning or repeated line search. To this end, we propose ADUCA, a *parameter-free* and *locally adaptive* cyclic method that, after a one-time line search at initialization, requires neither per-epoch line search nor any knowledge of Lipschitz constants. In the terminology of recent work, ADUCA is *auto-conditioned* [26]; unlike existing auto-conditioned methods, which

*Much of this work was completed while the author was an undergraduate student at the University of Wisconsin–Madison.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

use full-vector updates or specialized primal-dual/ADMM splitting limited to two or three blocks [1, 26, 28, 29, 31, 49], ADUCA operates in a general cyclic block coordinate regime.

Contributions Our main contributions are summarized as follows.

Locally adaptive, parameter-free cyclic updates. ADUCA is the first cyclic block coordinate method for GMVIs that is both *parameter-free* and *locally adaptive*: it requires no global or block-wise Lipschitz constants and no per-epoch line search beyond a one-time initialization. Even when specialized to convex minimization, ADUCA is the first general block coordinate method—cyclic or randomized—with both properties, resolving a question that had remained open in the auto-conditioned literature (barring the trivial case of exact cyclic descent for minimization).

Delayed updates and parallelization. ADUCA uses one-cycle-delayed operator information, enabling fully explicit cycle-wise step sizes and reducing cross-block synchronization. All blocks within a cycle can be updated in parallel using information from the previous cycle, while operators for the next cycle can be assembled via pipelined synchronization, allowing communication and computation to overlap and avoiding the global synchronization barrier of standard parallel schemes.

New adaptive step sizes. We propose a new adaptive step-size rule based on two computable local Lipschitz surrogates that capture consecutive-iterate and within-cycle variation, controlling the errors induced by cyclic updates. The rule is, to our knowledge, novel in the parameter-free literature.

Local oracle-complexity guarantees. Under monotonicity and local block-wise Lipschitzness, we prove an $\mathcal{O}(1/\epsilon)$ ergodic gap bound, matching the optimal rate [35, 40]. The bound depends on local Lipschitz constants over a compact set containing the algorithmic trajectory, rather than on global Lipschitz constants. Under strong convexity or restricted strong monotonicity (a.k.a. "sharpness"), we further establish near-linear convergence in $\mathcal{O}(\log^2(1/\epsilon))$ iterations, with linear dependence on the local condition number—an improvement over the (at least) quadratic dependence in the comparable auto-conditioned GMVI setting of [31].

Related work Our work sits at the intersection of three lines of research, and we compare with representative algorithms for VIs with respect to oracle complexity in Appendix A. For monotone VIs, classical extragradient and Mirror-Prox methods achieve $\mathcal{O}(1/\epsilon)$ ergodic oracle complexity [24, 35, 37], with linear convergence under strong monotonicity [38]; these rates are known to be optimal by oracle lower bounds [40]. Recent VI research has further developed last-iterate guarantees, stationarity analyses, and structured nonmonotone extensions [11, 13, 14, 20].

For cyclic block coordinate methods, by contrast, most theoretical results remain focused on minimization [4, 7, 21, 30, 47]. In the monotone-VI regime most relevant here, CODER is, to our knowledge, the only prior cyclic method with optimal convergence rates [44]; our analysis also builds on its Mahalanobis block-Lipschitz modeling framework. However, the parameter-free variant of CODER in [44] relies on line search, and its guarantees are phrased in terms of global problem properties.

A separate line of work develops *parameter-free* first-order methods that avoid global Lipschitz information by estimating local Lipschitzness/curvature between algorithm iterates. In the VI setting, [31] introduced the Golden Ratio Algorithm (GRAAL): a method employing local Lipschitz estimation to set step sizes adaptively, without any per-iteration line search. The method achieves the optimal sublinear rate under monotonicity alone, but the provided analysis leads to suboptimal scaling with the condition number in the strongly monotone regime (see also Table 1 in the appendix). Later work [1] refined and extended this approach. In particular, they showed that the explicit cap on the step size used in [31] can be removed, and established sublinear convergence under monotonicity as well as convergence guarantees for structured nonmonotone operators defined in [13]; however, no convergence results were provided for the strongly monotone case.

Recent literature has also studied local-curvature adaptivity in other related optimization settings. For example, for smooth convex minimization problems, [32] developed an adaptive gradient descent scheme, and subsequent work extended similar ideas to proximal methods [27, 33]. Accelerated variants have been developed in [5, 29, 46]. In the saddle-point setting, [49] provided a primal-dual first-order method adaptive to local smoothness, while [26] further extended such auto-conditioning properties to other classes of splitting algorithms.

2 Preliminaries

Notation We consider the standard d-dimensional Euclidean vector space $(\mathbb{R}^d, \|\cdot\|)$, where $\|\cdot\|$ denotes the ℓ_2 norm and $\langle\cdot,\cdot\rangle$ denotes the inner product of vectors. For any positive integer n, we denote the set $\{1, 2, \ldots, n\}$ by $[n]$. We reserve the notation m for the number of blocks, and assume a disjoint partition of the coordinates $[d]$ into nonempty sets $\{\mathcal{S}^i\}_{i\in[m]}$ of sizes $|\mathcal{S}^i| = d_i$, where $\sum_{i=1}^m d_i = d$. Without loss of generality, we assume that the partition is ordered, i.e., for $1 \le i < i' \le m$, $\max_{j\in\mathcal{S}^i} j < \min_{j'\in\mathcal{S}^{i'}} j'$, as permutations of coordinates do not affect our results. We let $\mathbf{u}^i$ and $\boldsymbol{F}^i(\cdot)$ be the subvectors of $\mathbf{u}$ and $\boldsymbol{F}(\cdot)$ indexed by the coordinates in $\mathcal{S}^i$, for $i \in [m]$. Given a matrix $\boldsymbol{A}$, we let $\|\boldsymbol{A}\| = \sup\{\|\boldsymbol{A}\mathbf{u}\| : \mathbf{u} \in \mathbb{R}^d, \|\mathbf{u}\| \le 1\}$ be the standard operator norm. For a positive (semi)definite matrix $\boldsymbol{A}$, $\|\cdot\|_{\boldsymbol{A}}$ denotes the Mahalanobis (semi)norm defined by $\|\mathbf{u}\|_{\boldsymbol{A}} = \sqrt{\langle \boldsymbol{A}\mathbf{u}, \mathbf{u}\rangle}$. Given positive semidefinite $d \times d$ matrices $\{\boldsymbol{Q}^i\}_{i=1}^m$, we define $\widehat{\boldsymbol{Q}}^i$ by $(\widehat{\boldsymbol{Q}}^i)_{j,k} := (\boldsymbol{Q}^i)_{j,k}$ if $\min\{j,k\} > \sum_{\ell=1}^{i-1} d_\ell$, otherwise 0, for $i \in [m]$. Throughout the paper, we use the convention $0/0 = 0$.

Problem setup We consider generalized Minty variational inequality (GMVI) problems:

$$\text{find } \mathbf{u}^* \in \mathbb{R}^d \quad \text{such that} \quad \big\langle \boldsymbol{F}(\mathbf{u}), \mathbf{u} - \mathbf{u}^*\big\rangle + g(\mathbf{u}) - g(\mathbf{u}^*) \ge 0, \quad \forall\, \mathbf{u} \in \mathbb{R}^d. \tag{P}$$

Given an error parameter $\epsilon > 0$, the goal is to find an ϵ-approximate solution $\mathbf{u}_\epsilon^*$ to (P), defined via:

$$\mathrm{Gap}(\mathbf{u}_\epsilon^*; \mathbf{u}) := \big\langle \boldsymbol{F}(\mathbf{u}), \mathbf{u}_\epsilon^* - \mathbf{u}\big\rangle + g(\mathbf{u}_\epsilon^*) - g(\mathbf{u}) \le \epsilon, \quad \forall\, \mathbf{u} \in \mathbb{R}^d. \tag{$\mathrm{P}_{\mathrm{approx}}$}$$

$\mathrm{Gap}(\mathbf{u}_\epsilon^*; \mathbf{u})$ is a gap-like function commonly used in the analysis of methods for variational inequalities [37]. In full generality, $\mathrm{Gap}(\mathbf{u}_\epsilon^*; \mathbf{u})$ is not guaranteed to be bounded uniformly over all $\mathbf{u}$, unless additional assumptions such as strong convexity or compact-domain restrictions are imposed on g. By Corollary 1 in Section 4, all iterates generated by our algorithm lie in a compact set. Our oracle complexity result is therefore stated on those compact sets.

Assumptions We make the following standard assumptions.

Assumption 1. *The operator $\boldsymbol{F} : \mathbb{R}^d \to \mathbb{R}^d$ is monotone, i.e., for any $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$*

$$\big\langle \boldsymbol{F}(\mathbf{u}) - \boldsymbol{F}(\mathbf{v}), \mathbf{u} - \mathbf{v}\big\rangle \ge 0.$$

The solution set of (P) *is nonempty, i.e., there exists at least one $\mathbf{u}^*$ that solves* (P)*.*

Assumption 2. *The function $g : \mathbb{R}^d \to (-\infty, \infty]$ is proper, lower semicontinuous, and block-separable, i.e., $g(\mathbf{u}) = \sum_{i=1}^m g^i(\mathbf{u}^i)$. Given positive diagonal matrices $\{\boldsymbol{\Lambda}_i\}_{i\in[m]}$, where $\boldsymbol{\Lambda}_i \in \mathbb{R}^{d_i \times d_i}$, each g^i is μ-(strongly) convex with a* possibly unknown *convexity modulus $\mu \ge 0$, i.e., for any $\mathbf{u}^i, \mathbf{v}^i$ and any subgradient $\boldsymbol{\zeta}^i \in \partial g^i(\mathbf{v}^i)$*

$$g^i(\mathbf{u}^i) \ge g^i(\mathbf{v}^i) + \big\langle \boldsymbol{\zeta}^i, \mathbf{u}^i - \mathbf{v}^i\big\rangle + \frac{\mu}{2}\big\|\mathbf{u}^i - \mathbf{v}^i\big\|_{\boldsymbol{\Lambda}_i}^2, \quad \text{where } \mu \ge 0,$$

and admits an efficiently computable proximal operator with parameter $\eta > 0$, i.e.,

$$\mathrm{prox}_{\eta g^i}(\mathbf{v}^i; \boldsymbol{\Lambda}_i) := \operatorname*{argmin}_{\mathbf{u}^i} \Big\{ g^i(\mathbf{u}^i) + \frac{1}{2\eta}\|\mathbf{u}^i - \mathbf{v}^i\|_{\boldsymbol{\Lambda}_i}^2 \Big\}.$$

In Assumption 2, for notational convenience, we use $\mu = 0$ for problems in which g is only convex and $\boldsymbol{F}$ is only monotone. At the end of Section 4, we discuss sharpness-type relaxations.

The role of matrices $\boldsymbol{\Lambda}_i$ will become clear from the definition of local block-wise Lipschitzness of $\boldsymbol{F}$; namely, these matrices allow for rescaling of the problem along the coordinate blocks. If matrices $\boldsymbol{\Lambda}_i$ are all identity matrices of appropriate sizes, then there is no rescaling and each norm $\|\cdot\|_{\boldsymbol{\Lambda}_i}$ is the simple ℓ_2 norm. The (strong) convexity and "prox-friendly" assumptions are stated with respect to $\|\cdot\|_{\boldsymbol{\Lambda}_i}$, which are equivalent to their standard ℓ_2 counterparts up to the scaling of μ and η. Throughout, we use $\boldsymbol{\Lambda}$ to denote the diagonal matrix obtained by concatenating the diagonals of $\boldsymbol{\Lambda}_1, \ldots, \boldsymbol{\Lambda}_m$ on the main diagonal, so that for any $\mathbf{u} \in \mathbb{R}^d$, we have $\|\mathbf{u}\|_{\boldsymbol{\Lambda}}^2 = \sum_{i=1}^m \|\mathbf{u}^i\|_{\boldsymbol{\Lambda}_i}^2$.

Assumption 3. *Given positive diagonal matrices $\{\boldsymbol{\Lambda}_i\}_{i\in[m]}$, for every compact set $\mathcal{C} \subseteq \mathbb{R}^d$, there exist positive semidefinite matrices $\{\boldsymbol{Q}_{\mathcal{C}}^i\}_{i\in[m]}$ such that each $\boldsymbol{F}^i(\cdot)$ is 1-Lipschitz continuous with respect to $\|\cdot\|_{\boldsymbol{Q}_{\mathcal{C}}^i}$, i.e., for any $\mathbf{u}, \mathbf{v} \in \mathcal{C}$*

$$\|\boldsymbol{F}^i(\mathbf{u}) - \boldsymbol{F}^i(\mathbf{v})\|_{\boldsymbol{\Lambda}_i^{-1}} \le \|\mathbf{u} - \mathbf{v}\|_{\boldsymbol{Q}_{\mathcal{C}}^i}. \tag{1}$$

Algorithm 1 ADUCA: Adaptive Delayed-Update Cyclic Algorithm for VI

1: **Input:** m, K, $\mathbf{u}_0 = \mathbf{v}_0$, $\beta \in (\frac{\sqrt{5}-1}{2}, 1)$, $\gamma \in \big(0, 1 - \frac{1}{\beta(1+\beta)}\big)$, $\rho \in \big(1, \frac{1}{\beta}\big)$, $\boldsymbol{\Lambda}$ (default $\boldsymbol{I}$), $\mu \geq 0$
2: **Initialize:** $\theta_0 = \omega_0 = 1$. Initialize $\tilde{\boldsymbol{F}}_0 = \boldsymbol{F}(\mathbf{u}_0)$, and $a_0 = a_{-1}$ (using the local backtracking initialization in Algorithm 2)
3: $\mathbf{u}_1 = \operatorname{argmin}_{\mathbf{u}} \big\{ a_0 \langle \boldsymbol{F}(\mathbf{u}_0), \mathbf{u} \rangle + a_0 g(\mathbf{u}) + \frac{1}{2}\|\mathbf{u} - \mathbf{v}_0\|^2_{\boldsymbol{\Lambda}} \big\}$
4: $\tilde{\boldsymbol{F}}_1^i = \boldsymbol{F}^i(\mathbf{u}_1^1, \ldots, \mathbf{u}_1^{i-1}, \mathbf{u}_0^i, \ldots, \mathbf{u}_0^m)$ $(i \in [m])$
5: **for** $k = 1$ **to** K **do**
6: Set step size a_k using the simplified condition (3) (or one of the complete conditions (11), (12), (13) in Appendices C.2 and C.3)
7: **for** $i = 1$ **to** m **do**
8: $\bar{\boldsymbol{F}}_k^i = \tilde{\boldsymbol{F}}_k^i + \frac{a_{k-1}\omega_{k-1}}{a_k}\big(\boldsymbol{F}^i(\mathbf{u}_{k-1}) - \tilde{\boldsymbol{F}}_{k-1}^i\big)$
9: $\mathbf{v}_k^i = (1-\beta)\mathbf{u}_k^i + \beta \mathbf{v}_{k-1}^i$
10: $\mathbf{u}_{k+1}^i = \operatorname{argmin}_{\mathbf{u}^i} \big\{ a_k \langle \bar{\boldsymbol{F}}_k^i, \mathbf{u}^i \rangle + a_k g^i(\mathbf{u}^i) + \frac{1}{2}\|\mathbf{u}^i - \mathbf{v}_k^i\|^2_{\boldsymbol{\Lambda}_i} \big\}$
11: $\tilde{\boldsymbol{F}}_{k+1}^i = \boldsymbol{F}^i(\mathbf{u}_{k+1}^1, \ldots, \mathbf{u}_{k+1}^{i-1}, \mathbf{u}_k^i, \ldots, \mathbf{u}_k^m)$
12: $\theta_k = \theta_{k-1}/\omega_{k-1}$, $\omega_k = \frac{1+\rho\beta\mu a_k}{1+\mu a_k}$
13: **return** $\hat{\mathbf{u}}_K = \sum_{k=1}^K \theta_k a_k \mathbf{u}_k / \sum_{k=1}^K \theta_k a_k$ and $\mathbf{v}_{K+1}$

Assumption 3 is a *local* block-wise Lipschitz condition that was previously used as a *global* condition to study fine-grained convergence properties of cyclic methods [7, 30, 44]. We emphasize here that matrices $\boldsymbol{Q}_{\mathcal{C}}^i$ from Assumption 3 need not be known to the algorithm—instead, the algorithm's convergence guarantee is with respect to the best possible such choice of matrices $\boldsymbol{Q}_{\mathcal{C}}^i$ based on all iterates. Further, existence of such matrices is guaranteed given local Lipschitzness of $\boldsymbol{F}$, since we can take $\boldsymbol{\Lambda}_i = \boldsymbol{I}_{d_i}$ and $\boldsymbol{Q}_{\mathcal{C}}^i = L_{\mathcal{C}}^2 \boldsymbol{I}_d$ for the local Lipschitz constant $L_{\mathcal{C}} > 0$ of the full operator $\boldsymbol{F}$. Similarly, this condition is implied (and most general) among block-wise Lipschitz conditions for variational inequalities studied in the literature. In particular, it is implied by the (local versions of) block-wise Lipschitz assumptions of the form $\|\boldsymbol{F}^i(\mathbf{u}) - \boldsymbol{F}^i(\mathbf{v})\| \leq L_{\mathcal{C}}^i \|\mathbf{u} - \mathbf{v}\|$ from prior work on block coordinate methods for VIs [12, 25], by taking $\boldsymbol{\Lambda}_i = \boldsymbol{I}_{d_i}$ and $\boldsymbol{Q}_{\mathcal{C}}^i = (L_{\mathcal{C}}^i)^2 \boldsymbol{I}_d$ for some $L_{\mathcal{C}}^i > 0$. Finally, we remark that block-wise Lipschitz assumptions that restrict attention to vector pairs differing only on block i are common in the literature on block coordinate methods for minimization problems [4, 36]; however, they are incompatible with general VIs without additional assumptions on $\boldsymbol{F}$—such as block cocoercivity [9] or uniform boundedness of block operator norms $\|\boldsymbol{F}^i\|$ [51]—as remarked in [12, 44].

3 ADUCA: Adaptive Delayed-Update Cyclic Algorithm

Our algorithm is formally stated in Algorithm 1, and its components—the update scheme and the adaptive step size scheme—are introduced in detail below.

Update scheme Our proposed algorithm utilizes cyclic block coordinate updates with extrapolation. For the ith inner iteration of the kth cycle, let $\tilde{\boldsymbol{F}}_k^i := \boldsymbol{F}^i(\mathbf{u}_k^1, \ldots, \mathbf{u}_k^{i-1}, \mathbf{u}_{k-1}^i, \ldots, \mathbf{u}_{k-1}^m)$ denote the operator of the partially updated and one cycle-delayed iterate $\mathbf{u}_{k-1,i} := (\mathbf{u}_k^1, \ldots, \mathbf{u}_k^{i-1}, \mathbf{u}_{k-1}^i, \ldots, \mathbf{u}_{k-1}^m)^\top$. Then, ADUCA computes the new block iterate $\mathbf{u}_{k+1}^i$ by first extrapolating the operator in Line 8 and updating the prox-center $\mathbf{v}_k^i$ in Line 9, followed by the block prox-type update in Line 10 of Algorithm 1. In the algorithm, $\{a_k\}_{k\geq 0}$ are the adaptive step sizes to be discussed below, and $\{\omega_k\}_{k\geq 0}$ are introduced to account for possible strong convexity and achieve faster near-linear convergence when it applies. Note that the minimization step in Line 10 is performed with respect to $\|\cdot\|_{\boldsymbol{\Lambda}_i}$, allowing block-wise scaling for the step sizes and complementing cycle-wise adaptiveness.

Additionally, different from previous block coordinate methods, the algorithm uses the operator values $\tilde{\boldsymbol{F}}_k$ rather than $\tilde{\boldsymbol{F}}_{k+1}$, which are one cycle *delayed* (hence '*Delayed*' in ADUCA), to update $\mathbf{u}_{k+1}$ in Line 10 of Algorithm 1. This arises from the need to adaptively determine the cycle-wise step sizes at the beginning of each cycle. This delayed scheme also yields an appealing feature of ADUCA: despite the inherently sequential nature of cyclic updates, the updates can in fact be implemented in parallel. As remarked earlier, this delay has further implications on practical implementations, as

strict synchronization of updates is not required. In Sections 4 and 6, we show that the algorithm exhibits efficient convergence, both theoretically and empirically, despite its delayed nature.

Adaptive step size Before introducing the step size scheme, we firstly define the hyperparameters for brevity. We discuss possible specific choices later in this section. We define $\beta \in \left(\frac{\sqrt{5}-1}{2}, 1\right)$, $\gamma \in \left(0, 1 - \frac{1}{\beta(1+\beta)}\right)$, $\rho \in \left(1, \frac{1}{\beta}\right)$. Here, β denotes the extrapolation weight in Line 9, Algorithm 1, while γ and ρ, dependent on β, bound the step size growth. We further define auxiliary notation $\eta := \sqrt{\frac{\gamma(1+\beta)}{1+\beta^2}}$. In the following, we provide the simplest set of conditions for the step size update, without any tunable hyperparameters. The complete set is provided in Appendix C, which can be employed to obtain potentially larger step sizes when the strong convexity parameter μ is known (e.g., when g is used as an explicit strongly convex regularizer). The complete set of conditions is also more convenient for carrying out the analysis, as it was, in fact, obtained from the analysis.

Our step size conditions depend on *local* Lipschitz constants, defined by

$$\hat{L}_k := \frac{\|\boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k\|_{\boldsymbol{\Lambda}^{-1}}}{\|\mathbf{u}_k - \mathbf{u}_{k-1}\|_{\boldsymbol{\Lambda}}}, \quad L_k := \frac{\|\boldsymbol{F}(\mathbf{u}_k) - \boldsymbol{F}(\mathbf{u}_{k-1})\|_{\boldsymbol{\Lambda}^{-1}}}{\|\mathbf{u}_k - \mathbf{u}_{k-1}\|_{\boldsymbol{\Lambda}}}. \tag{2}$$

The boundedness of L_k follows directly from the standard definition of local Lipschitz continuity, with additional flexibility from block-wise scaling via $\boldsymbol{\Lambda}$. For $\hat{L}_k$, by Assumption 3, for any compact set $\mathcal{C}$, there exist $\{\boldsymbol{Q}_{\mathcal{C}}^i\}_{i\in[m]}$ and corresponding $\{\widehat{\boldsymbol{Q}}_{\mathcal{C}}^i\}_{i\in[m]}$ such that

$$\|\boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k\|_{\boldsymbol{\Lambda}^{-1}}^2 \le \sum_{i=1}^m \|\mathbf{u}_k - \mathbf{u}_{k-1,i}\|_{\boldsymbol{Q}_{\mathcal{C}}^i}^2 \le \big\|\boldsymbol{\Lambda}^{-1/2}\big(\sum_{i=1}^m \widehat{\boldsymbol{Q}}_{\mathcal{C}}^i\big)\boldsymbol{\Lambda}^{-1/2}\big\|\|\mathbf{u}_k - \mathbf{u}_{k-1}\|_{\boldsymbol{\Lambda}}^2.$$

Here, $\mathcal{C}$ can be chosen as a compact set containing the algorithm iterates; the existence of such a set is established in Corollary 1 of Section 4. In each cycle, $\hat{L}_k$ locally estimates $\hat{L} := \|\boldsymbol{\Lambda}^{-1/2}(\sum_{i=1}^m \widehat{\boldsymbol{Q}}_{\mathcal{C}}^i)\boldsymbol{\Lambda}^{-1/2}\|^{1/2}$, a fine-grained Lipschitz constant characterizing the aggregated progress of cyclic updates [6, 7, 44, 45]. For further details, we refer to the discussion in [44].

Now we present our simplest set of step size conditions. Let $\rho_0 := \min\{\rho, \beta(1+\beta)(1-\gamma)\}$, $\tau = \frac{3\rho_0^2(1+\rho\beta)}{2(\rho\beta)^2+3\rho_0^2(1+\rho\beta)} \in (0,1)$, and recall that $\eta = \sqrt{\frac{\gamma(1+\beta)}{1+\beta^2}}$. Our adaptive step size conditions are:

$$a_k = \min\Big\{\rho_0 a_{k-1},\ \frac{\eta}{2\sqrt{\beta}}\min\Big\{\frac{\sqrt{\tau}\,\rho\beta}{\sqrt{3}L_k\sqrt{1+\rho\beta}},\ \frac{\sqrt{(1-\tau)\rho\beta}}{\sqrt{2}\hat{L}_k}\Big\}\sqrt{\frac{a_{k-1}}{a_{k-2}}}\Big\}. \tag{3}$$

For example, one could choose $\rho = 1.2$, $\beta = 0.8$, $\gamma = 0.2$ for the algorithm parameters and obtain a fully specified step size update that does not require any further parameter selection. Such a choice leads to the following simple and fully specified step size rule:

$$a_k = \min\Big\{1.15\,a_{k-1},\ \min\Big\{\frac{0.093}{L_k},\ \frac{0.079}{\hat{L}_k}\Big\}\sqrt{\frac{a_{k-1}}{a_{k-2}}}\Big\}. \tag{4}$$

We remark that even though the step sizes can be made fully independent of any global problem parameters (including μ; see (3) and (4)), the algorithm itself, as specified, requires knowledge of μ to update ω_k, which appears in the update for the extrapolated operator $\bar{\boldsymbol{F}}_k$. This might appear to undermine the adaptivity of ADUCA when $\mu > 0$. However, $\{\omega_k\}_{k\ge 0}$ was introduced only to facilitate the theoretical near-linear convergence-rate analysis and is not essential to the practical implementation. In general, since μ is used solely to certify a near-linear rate, it can be replaced throughout the analysis (and hence in the definition of ω_k) by any conservative lower bound $\underline{\mu} \in (0, \mu]$, in which case the strong convexity assumption Assumption 2 and Fact B.3 remain valid with $\underline{\mu}$. Using such a lower bound may weaken the theoretical rate constant, but the effect is mild because the factor $\frac{1+\rho\beta\mu a_k}{1+\mu a_k}$ always satisfies $\rho\beta < \frac{1+\rho\beta\mu a_k}{1+\mu a_k} < 1$ for $\mu a_k > 0$, and in typical parameter choices $\rho\beta$ is close to 1; for instance, the values used in our experiments, $\rho = 1.2$ and $\beta = 0.8$, yield $\rho\beta = 0.96$. The empirical impact of using a conservative lower bound is further examined in the ablation study in Appendix F.3, which shows that the algorithm performance is insensitive to the choice of μ; even setting $\mu = 0$ still leads to convergence that is indistinguishable from runs with higher values of μ. We conjecture that this is not a coincidence and that the algorithm can, in fact, be proven to converge linearly even if μ is set to zero. This would, however, increase the complexity of an already technical analysis, so we defer such considerations to future work.

Due to space constraints, additional remarks on the proposed adaptive scheme, including comparisons with related work, are deferred to Appendix C.1.

4 Convergence Analysis

In this section, we carry out the convergence analysis of ADUCA by bounding the merit functions

$$h_k(\mathbf{u}) := a_k\big(\langle \boldsymbol{F}(\mathbf{u}_k), \mathbf{u}_k - \mathbf{u}\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\big) \tag{5}$$

for $\{\mathbf{u}_k\}_{k\geq 1}$, as is standard in the analysis of VI methods; see, e.g., [35]. We also use the weights $\theta_k = \theta_{k-1}/\omega_{k-1}$ ($k \geq 1$, $\theta_0 = 1$), where $\omega_k = \frac{1+\rho\beta\mu a_k}{1+\mu a_k} \in (0, 1]$. Intuitively, $h_k(\mathbf{u})$ measures the one-step VI error at iterate $\mathbf{u}_k$, while the choice of θ_k is tailored so that the delayed-update correction terms telescope across iterations.

In the following, we provide an overview of the complete technical argument, while, due to space constraints, proofs are deferred to Appendix D. To begin, the following lemma explains why controlling the weighted average of the merit functions is enough to control the restricted gap.

Lemma 1 (Weighted merit bound implies a restricted-gap bound)**.** *Suppose Assumptions 1 and 2 hold. Given $K \geq 1$, any sequence $\{\mathbf{u}_k\}_{k=1}^K \subset \mathbb{R}^d$, and any positive sequences $\{a_k\}_{k=1}^K$ and $\{\theta_k\}_{k=1}^K$, define $A_K := \sum_{k=1}^K \theta_k a_k$, $\hat{\mathbf{u}}_K := \sum_{k=1}^K \theta_k a_k \mathbf{u}_k / A_K$. Then, for every $\mathbf{u} \in \mathrm{dom}(g)$, $\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \frac{\sum_{k=1}^K \theta_k h_k(\mathbf{u})}{A_K}$. Consequently, if $\sum_{k=1}^K \theta_k h_k(\mathbf{u})/A_K \leq \epsilon$, then $\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \epsilon$.*

While bounding similar "gap-like" functions is standard, the specific decomposition below is tailored to the delayed cyclic structure of ADUCA. The first step is to decompose the merit function as

$$h_k(\mathbf{u}) = a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \bar{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle + a_k\big(\big\langle \bar{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\big). \tag{6}$$

The decomposition is based on replacing $\boldsymbol{F}(\mathbf{u}_k)$ with the extrapolated term $\bar{\boldsymbol{F}}_k$ and controlling the introduced "error" $h_k^{(1)}(\mathbf{u}) := a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \bar{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle$. In particular, similar to prior work utilizing extrapolated operators [12, 25, 44], we split $h_k^{(1)}(\mathbf{u})$ into telescoping inner product terms and an error term that we later cancel out (Lemma 2). The remaining term $h_k^{(2)}(\mathbf{u}) := a_k\big(\big\langle \bar{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\big)$ on the right-hand side of (6) is controlled by the algorithm update and we show it contains negative quadratic distance terms scaling with $\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2$, which are used in controlling any introduced "error" terms (Lemma 3).

Before delving into the technical details, we describe the rationale behind the different algorithmic choices in our algorithm. First, the extrapolation $\bar{\boldsymbol{F}}_k$ is used to ensure the updates to $\mathbf{u}_{k+1}$ indeed correspond to cyclic block updates. The delayed nature of $\bar{\boldsymbol{F}}_k$ is essential to ensure we can select step size parameters a_k *adaptively* to local geometry; we comment on this more as we carry out the analysis. The update $\mathbf{u}_{k+1}$ resembles the GRAAL update from [31], but with the crucial difference that the full operator $\boldsymbol{F}(\mathbf{u}_k)$ is replaced by the extrapolated one with cyclic block updates—$\bar{\boldsymbol{F}}_k$. While this may appear as a small difference, it nevertheless has a substantial impact on the analysis, leading to more challenges in setting the conditions on the step size update and requiring a new proof strategy, one that controls the squared distance terms $\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2$ not only over two consecutive iterations (as commonly happens for similar algorithms [12, 25, 31, 44]) but over *four* consecutive iterations.

Lemma 2. *Let $h_k^{(1)}(\mathbf{u}) := a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \bar{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle$, where $\bar{\boldsymbol{F}}_k = \tilde{\boldsymbol{F}}_k + \frac{a_{k-1}\omega_{k-1}}{a_k}(\boldsymbol{F}(\mathbf{u}_{k-1}) - \tilde{\boldsymbol{F}}_{k-1})$. Then we obtain*

$$\begin{aligned} h_k^{(1)}(\mathbf{u}) = {} & a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_{k+1} - \mathbf{u}\big\rangle - a_{k-1}\omega_{k-1}\big\langle \boldsymbol{F}(\mathbf{u}_{k-1}) - \tilde{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}\big\rangle \\ & + a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}_{k+1}\big\rangle. \end{aligned}$$

Observe here that when $h_k^{(1)}(\mathbf{u})$ is multiplied by θ_k, the first two terms on the right-hand side in Lemma 2 telescope, as $\theta_k\omega_{k-1} = \theta_{k-1}$, by the definition of θ_k. Thus, only the third inner product term $a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}_{k+1}\big\rangle$ will need to be controlled by the algorithm update. Here, it is crucial that $\tilde{\boldsymbol{F}}_k$ does not depend on the information from the current cycle, since it ensures that $\|\boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k\|_{\boldsymbol{\Lambda}^{-1}}$ is a function of local Lipschitzness between points from the previous cycle and so it is available to the algorithm at the time the step size a_k is being set.

A more technical part of the proof concerns appropriately bounding $h_k^{(2)}(\mathbf{u}) := a_k\big(\big\langle \bar{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\big)$. Here, the inspiration comes from GRAAL-type algorithms [1, 31], which leverage

a conveniently chosen sequence of prox-centers $\mathbf{v}_k$ and the property of Euclidean norms stated in Fact B.2. Our analysis is, however, different, as it directly bounds the merit function and is adapted to the operator extrapolation specific to our algorithm.

Lemma 3. *Consider the iterates $\mathbf{u}_k$, $k \geq 1$, of Algorithm 1. Then, for any $\mathbf{u} \in \mathrm{dom}(g)$, we have that*

$$\begin{aligned} h_k^{(2)}(\mathbf{u}) \leq \frac{1}{2(1-\beta)}\Big((1+\beta\mu a_k)\|\mathbf{u}-\mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 - (1+\mu a_k)\|\mathbf{u}-\mathbf{v}_{k+1}\|_{\boldsymbol{\Lambda}}^2\Big) \\ + \frac{\gamma\beta^2(1+\beta)}{2(1+\beta^2)}\big(\omega_{k-1}\|\mathbf{u}_k-\mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2 - \|\mathbf{u}_{k+1}-\mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2\big) \\ + a_k\big\langle \bar{\boldsymbol{F}}_k - \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\big\rangle \\ - \Big(\frac{\gamma\omega_{k-1}(1+\beta)}{2(1+\omega_{k-1})(1+\beta^2)} + \frac{a_k(1+\beta\mu a_{k-1})}{2\beta a_{k-1}}\Big)\|\mathbf{u}_{k+1}-\mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2. \end{aligned}$$

The bound on $h_k^{(2)}(\mathbf{u})$ in Lemma 3 may be difficult to parse, so we explain here what it states. The terms in the first two lines of the bound telescope when multiplied by θ_k, which is ensured by $\theta_k = \theta_{k-1}/\omega_{k-1}$ and our conditions on step sizes a_k. The remaining two lines contain an "error" that we show how to control in the sequel. What we have achieved thus far is to show that we can bound the merit function $h_k(\mathbf{u})$ by terms that telescope when multiplied by θ_k and terms scaling with squared distances $\|\mathbf{u}_{k+1}-\mathbf{u}_k\|^2$. In particular, Lemmas 2 and 3 imply

$$\theta_k h_k(\mathbf{u}) \leq [\text{telescoping terms}]_k + \theta_k \mathcal{E}_k, \tag{7}$$

where $\mathcal{E}_k := a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}_{k+1}\big\rangle + a_k\big\langle \bar{\boldsymbol{F}}_k - \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\big\rangle - \big(\frac{\gamma\omega_{k-1}(1+\beta)}{2(1+\omega_{k-1})(1+\beta^2)} + \frac{a_k(1+\beta\mu a_{k-1})}{2\beta a_{k-1}}\big)\|\mathbf{u}_{k+1}-\mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2$, with the convention $\mathbf{u}_i = \mathbf{u}_0$ and $\tilde{\boldsymbol{F}}_i = \boldsymbol{F}(\mathbf{u}_0)$ for $i < 0$. The key next step is to show that $\mathcal{E}_k$ can be effectively controlled, using local Lipschitz assumptions on $\boldsymbol{F}$ and by appropriately selecting the (adaptive) step size a_k.

Lemma 4. *Consider the iterates $\mathbf{u}_k$, $k \geq 1$, of Algorithm 1 and $\mathcal{E}_k$ defined in the preceding paragraph. Then, under Assumption 3, for all $k \geq 1$,*

$$\begin{aligned} \theta_k\mathcal{E}_k \leq -\frac{c_k}{2}\|\mathbf{u}_{k+1}-\mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2 \\ + \frac{c_{k-1}}{8}\|\mathbf{u}_k-\mathbf{u}_{k-1}\|_{\boldsymbol{\Lambda}}^2 + \frac{c_{k-2}}{4}\|\mathbf{u}_{k-1}-\mathbf{u}_{k-2}\|_{\boldsymbol{\Lambda}}^2 + \frac{c_{k-3}}{8}\|\mathbf{u}_{k-2}-\mathbf{u}_{k-3}\|_{\boldsymbol{\Lambda}}^2, \end{aligned}$$

where $c_k := \frac{\theta_k a_k(\beta\mu a_{k-1}+1)}{\beta a_{k-1}}$ for $k \geq 0$ and $c_i = 0$ for $i < 0$.

We are now ready to present our main convergence theorem, derived by combining the preceding lemmas and carefully telescoping and absorbing the error terms induced by the update scheme.

Theorem 1. *Suppose Assumptions 1, 2, and 3 hold. Let Algorithm 1 be run with any $a_0 > 0$, and suppose that the step sizes $\{a_k\}_{k\geq 1}$ satisfy the complete step-size conditions in (11) of Appendix C.2 or any sufficient simplified version of them (e.g., (3)). Define $\mathcal{B}_0(\mathbf{u}) := \frac{1+\mu a_0}{2(1-\beta)}\|\mathbf{u}-\mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2 + \Big(\frac{\eta^2\beta^2}{2} + \frac{1+\beta\mu a_0}{2\beta}\Big)\|\mathbf{u}_1-\mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2$. Then, for every $K \geq 2$ and every $\mathbf{u} \in \mathrm{dom}(g)$,*

$$A_K\,\mathrm{Gap}(\hat{\mathbf{u}}_K;\mathbf{u}) + \frac{\theta_K(1+\mu a_K)}{4(1-\beta)}\|\mathbf{u}-\mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2 \leq \mathcal{B}_0(\mathbf{u}), \tag{8}$$

where $A_K = \sum_{k=1}^K \theta_k a_k$ and $\hat{\mathbf{u}}_K = \frac{\sum_{k=1}^K \theta_k a_k \mathbf{u}_k}{A_K}$. Consequently, $\mathrm{Gap}(\hat{\mathbf{u}}_K;\mathbf{u}) \leq \frac{\mathcal{B}_0(\mathbf{u})}{A_K}$. Moreover, for any solution $\mathbf{u}^*$ of (P),

$$\|\mathbf{u}^*-\mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2 \leq \frac{4(1-\beta)\mathcal{B}_0(\mathbf{u}^*)}{\theta_K}.$$

Corollary 1. *Suppose the assumptions of Theorem 1 hold and $\mathbf{u}_0 \in \mathrm{dom}(\partial g)$. Let $\mathbf{u}^*$ be any solution of (P), and let $G_0 := \min_{\xi_0 \in \partial g(\mathbf{u}_0)} \|\boldsymbol{F}(\mathbf{u}_0)+\xi_0\|_{\boldsymbol{\Lambda}^{-1}}$. For a fixed $a_{\max} > 0$ and any run of ADUCA using $a_0 \in (0, a_{\max}]$, there exists a compact set $\mathcal{C} \subset \mathbb{R}^d$, depending only on $\mathbf{u}_0$, $a_{\max}$, G_0, and $\mathbf{u}^*$, such that the algorithm iterates lie in $\mathcal{C}$, i.e., $\{\mathbf{u}_k, \mathbf{v}_k \mid k \geq 0\} \cup \{\mathbf{u}_{k-1,i} \mid k \geq 1,\ i \in [m]\} \subseteq \mathcal{C}$, where $\mathbf{u}_{k-1,i} := \big(\mathbf{u}_k^1, \ldots, \mathbf{u}_k^{i-1}, \mathbf{u}_{k-1}^i, \ldots, \mathbf{u}_{k-1}^m\big)$.*

Algorithm 2 Backtracking Initialization for ADUCA

Input: $m, \mathbf{u}_0, \mathbf{\Lambda}, C, \hat{C}$
Initialize: $i = 0$
$\mathbf{u}_{\text{init}} = \text{argmin}_{\mathbf{u}}\{\langle \boldsymbol{F}(\mathbf{u}_0), \mathbf{u}\rangle + g(\mathbf{u}) + \frac{1}{2}\|\mathbf{u} - \mathbf{u}_0\|^2_{\mathbf{\Lambda}}\}$
$\tilde{\boldsymbol{F}}^j_{\text{init}} = \boldsymbol{F}^j(\mathbf{u}^1_{\text{init}}, \ldots, \mathbf{u}^{j-1}_{\text{init}}, \mathbf{u}^j_0, \ldots, \mathbf{u}^m_0), j \in [m]$
$L_{\text{init}} = \frac{\|\boldsymbol{F}(\mathbf{u}_{\text{init}}) - \boldsymbol{F}(\mathbf{u}_0)\|_{\mathbf{\Lambda}^{-1}}}{\|\mathbf{u}_{\text{init}} - \mathbf{u}_0\|_{\mathbf{\Lambda}}}$, $\hat{L}_{\text{init}} = \frac{\|\boldsymbol{F}(\mathbf{u}_{\text{init}}) - \tilde{\boldsymbol{F}}_{\text{init}}\|_{\mathbf{\Lambda}^{-1}}}{\|\mathbf{u}_{\text{init}} - \mathbf{u}_0\|_{\mathbf{\Lambda}}}$
$a_{\text{init}} = \min\{C/L_{\text{init}}, \hat{C}/\hat{L}_{\text{init}}\}$
repeat
  $a = a_{\text{init}} 2^{-i}, \quad i \leftarrow i + 1$
  $\mathbf{u}_1(a) = \text{argmin}_{\mathbf{u}}\{a\langle \boldsymbol{F}(\mathbf{u}_0), \mathbf{u}\rangle + ag(\mathbf{u}) + \frac{1}{2}\|\mathbf{u} - \mathbf{u}_0\|^2_{\mathbf{\Lambda}}\}$
  $\tilde{\boldsymbol{F}}^j_1(a) = \boldsymbol{F}^j(\mathbf{u}^1_1(a), \ldots, \mathbf{u}^{j-1}_1(a), \mathbf{u}^j_0, \ldots, \mathbf{u}^m_0), j \in [m]$
  $L_1(a) = \frac{\|\boldsymbol{F}(\mathbf{u}_1(a)) - \boldsymbol{F}(\mathbf{u}_0)\|_{\mathbf{\Lambda}^{-1}}}{\|\mathbf{u}_1(a) - \mathbf{u}_0\|_{\mathbf{\Lambda}}}$, $\hat{L}_1(a) = \frac{\|\boldsymbol{F}(\mathbf{u}_1(a)) - \tilde{\boldsymbol{F}}_1(a)\|_{\mathbf{\Lambda}^{-1}}}{\|\mathbf{u}_1(a) - \mathbf{u}_0\|_{\mathbf{\Lambda}}}$
until $aL_1(a) \leq C$ **and** $a\hat{L}_1(a) \leq \hat{C}$
return $a_0 = a, a_{-1} = a$

Further implications under relaxed conditions. We briefly discuss here how our results apply even under more relaxed assumptions about the problem. In particular, convergence of $\|\mathbf{v}_k - \mathbf{u}^*\|^2_{\mathbf{\Lambda}}$ at rate $1/\theta_K$ (argued to be near-linear in the next section) can be guaranteed without assuming strong convexity for g. What suffices is that $\boldsymbol{F}$ satisfies a "restricted strong monotonicity" condition with a parameter $\mu > 0$, meaning that there is a solution $\mathbf{u}^*$ such that for all $\mathbf{u} \in \text{dom}(g)$:

$$\langle \boldsymbol{F}(\mathbf{u}) - \boldsymbol{F}(\mathbf{u}^*), \mathbf{u} - \mathbf{u}^*\rangle \geq \mu\|\mathbf{u} - \mathbf{u}^*\|^2_{\mathbf{\Lambda}}. \tag{9}$$

The reason is as follows. The monotonicity of $\boldsymbol{F}$ is used in Lemma 1 to compare the merit term with the restricted gap. Under (9), the same comparison holds with an additional quadratic decrease when the reference point is $\mathbf{u}^*$. Equivalently, one may apply the analysis to the shifted operator $\widehat{\boldsymbol{F}}(\mathbf{u}) = \boldsymbol{F}(\mathbf{u}) - \mu\mathbf{\Lambda}(\mathbf{u} - \mathbf{u}^*)$ and shifted regularizer $\widehat{g}(\mathbf{u}) = g(\mathbf{u}) + \frac{\mu}{2}\|\mathbf{u} - \mathbf{u}^*\|^2_{\mathbf{\Lambda}}$, which generate the same iterates as the original problem while making the regularizer strongly convex. Thus θ_k grows geometrically and Theorem 1 yields near-linear convergence.

5 Initialization and Global Oracle Complexity Bound

In this section, we provide oracle complexity bounds for ADUCA by characterizing the growth of the step-size sequence under a proper initialization with logarithmic cost. We analyze the adaptive step size scheme (3), rewritten as $a_k = \min\{\rho_0 a_{k-1}, s_k\sqrt{a_{k-1}/a_{k-2}}\}$, where $s_k := \min\{C/L_k, \hat{C}/\hat{L}_k\}$, $C := (\eta\rho/2)\sqrt{\tau\beta/[3(1+\rho\beta)]}$, and $\hat{C} := (\eta/2)\sqrt{(1-\tau)\rho/2}$. For fixed β, γ, ρ, both C and $\hat{C}$ are absolute numerical constants independent of any problem parameters.

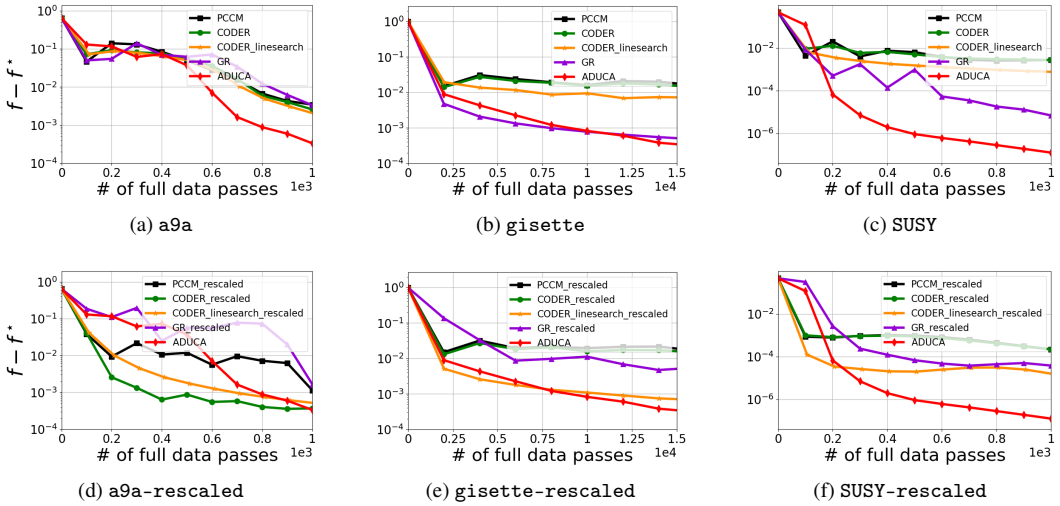
Backtracking initialization. We state our backtracking initialization scheme in Algorithm 2. Similar initialization schemes have been used in auto-conditioned first-order methods [1, 26, 29]. Note that for the $a_{\text{init}} > 0$ from Algorithm 2, there exists a compact set $\mathcal{C}$ containing all ADUCA iterates by Corollary 1. This leads to the existence of $\{\boldsymbol{Q}^i_{\mathcal{C}}\}_{i=1}^m$, and local Lipschitz constants $L_{\mathcal{C}} := \|\mathbf{\Lambda}^{-1/2}(\sum_{i=1}^m \boldsymbol{Q}^i_{\mathcal{C}})\mathbf{\Lambda}^{-1/2}\|^{1/2}$ and $\hat{L}_{\mathcal{C}} := \|\mathbf{\Lambda}^{-1/2}(\sum_{i=1}^m \widehat{\boldsymbol{Q}}^i_{\mathcal{C}})\mathbf{\Lambda}^{-1/2}\|^{1/2}$ under Assumption 3. As discussed in Sec. 3, $L_k \leq L_{\mathcal{C}}$ and $\hat{L}_k \leq \hat{L}_{\mathcal{C}}$. Let $s_{\mathcal{C}} := \min\{C/L_{\mathcal{C}}, \hat{C}/\hat{L}_{\mathcal{C}}\}$, with $s_{\mathcal{C}} \leq s_k$ ($\forall k \geq 1$).

The following lemma shows that Algorithm 2 terminates after logarithmically many backtracking steps and returns an initial step size a_0 at the correct scale for convergence, with proofs in Appendix E.

Lemma 5. *Suppose Assumptions 1–3 hold, and* $\mathbf{u}_0 \in \text{dom}(\partial g)$. *Then the repeat loop in Algorithm 2 terminates after at most* $\lceil \log_2(a_{\text{init}}/s_{\mathcal{C}})\rceil$ *steps and outputs* a_0 *satisfying*

$$s_{\mathcal{C}}/2 \leq a_0 \leq s_1(a_0),$$

where $s_1(a) := \min\{C/L_1(a), \hat{C}/\hat{L}_1(a)\}$, *and* $L_1(a)$, $\hat{L}_1(a)$ *are defined in Line 11 of Algorithm 2 for any* $a > 0$.



(a) a9a (b) gisette (c) SUSY

(d) a9a-rescaled (e) gisette-rescaled (f) SUSY-rescaled

Figure 1: Primal objective suboptimality $f(\mathbf{x}) - f^\star$ for the SVM problem versus the number of full data passes on a9a, gisette, and SUSY. Plots (a)–(c) use the original formulation, while plots (d)–(f) use the diagonally rescaled formulation for every algorithm, induced by the same diagonal matrix $\mathbf{\Lambda}$ defined in (45)–(46). In these figures, ADUCA uses $\mathbf{\Lambda}$ throughout (a)–(f).

Global complexity bound. The main part of the argument consists in proving that, in the aggregate, the algorithm must lead to a large cumulative sum of the step sizes. Notably, this does not prevent some of the step sizes from becoming small; instead, what we argue is that there cannot be any long sequence of small step sizes that is not compensated by large step sizes. This argument is technical and it is deferred to Appendix E. The main result is summarized in the theorem below.

Theorem 2. *Suppose Assumptions 1–3 hold, $\mathbf{u}_0 \in \mathrm{dom}(\partial g)$, and ADUCA is initialized by Algorithm 2. For any compact set $\mathcal{U} \subset \mathrm{dom}(g)$, define $\mathcal{B}_{\mathcal{U}} := \sup_{\mathbf{u}\in\mathcal{U}} \mathcal{B}_0(\mathbf{u})$. Let $\mathcal{L}_{\mathcal{C}} := 1/s_{\mathcal{C}}$ and $R_* := \big(2(1+\mu a_{\mathrm{init}})(\|\mathbf{u}_0 - \mathbf{u}^*\|_{\mathbf{\Lambda}} + a_{\mathrm{init}}G_0)^2 + 2(1-\beta)\big(\eta^2\beta^2 + \frac{1+\beta\mu a_{\mathrm{init}}}{\beta}\big)a_{\mathrm{init}}^2 G_0^2\big)^{1/2}$ with G_0 defined in Corollary 1. Then, for any $\epsilon > 0$, after at most K iterations, where*

$$K = \mathcal{O}\Big(\min\Big\{\frac{\mathcal{L}_{\mathcal{C}}\mathcal{B}_{\mathcal{U}}}{\epsilon}, \frac{\mathcal{L}_{\mathcal{C}}}{\mu}\log\big(\frac{R_*}{\epsilon}\big)\log\big(\frac{\mathcal{L}_{\mathcal{C}}\mathcal{B}_{\mathcal{U}}}{\epsilon}\big)\Big\}\Big),$$

the output of ADUCA guarantees either $\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \le \epsilon$ for all $\mathbf{u} \in \mathcal{U}$, or $\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}} \le \epsilon$.

6 Numerical Experiment

In this section, we evaluate ADUCA and compare it with several representative comparable methods: Proximal Cyclic Coordinate Method (PCCM), Cyclic cOordinate Dual avEraging with extRapolation (CODER) [44], a parameter-free variant of CODER based on per-epoch line-search (CODER-LineSearch) [44], and the Golden RAtio ALgorithm (GRAAL) [31]. PCCM is the straightforward cyclic proximal update for (P), but it is not guaranteed to converge for general monotone variational inequalities. CODER is, to our knowledge, the first cyclic coordinate method with provable convergence guarantees for (P), while CODER-LineSearch makes CODER adaptive by performing a line search in every epoch, at the cost of additional operator evaluations. GRAAL is a full-operator method that is locally adaptive and does not require prior knowledge of Lipschitz constants.

We provide reproducibility details for the experiments in Appendix F.1. We also present additional results showing ADUCA takes larger steps via local Lipschitz-estimate comparisons (Appendix F.2) and evaluate its sensitivity to the hyperparameter $\mu \ge 0$ (Appendix F.3). Additional experiments for the Traffic Network Equilibrium problem are in Appendix F.4.

Support vector machine. We consider the convex elastic net regularized support vector machine, formulated as $\min_{\mathbf{x}\in\mathbb{R}^d} \max_{\mathbf{y}\in\mathbb{R}^n} \frac{1}{n}\sum_{i=1}^n y_i(-1 + b_i \boldsymbol{A}_i^\top \mathbf{x}) + \lambda_1\|\mathbf{x}\|_1 + \frac{\lambda_2}{2}\|\mathbf{x}\|_2^2 - \sum_{j=1}^n \iota_{[-1,0]}(y_j)$, where $\iota_{[-1,0]}(\cdot)$ is the convex indicator function of the interval $[-1, 0]$. Let $\bar{\boldsymbol{A}} = [b_1\boldsymbol{A}_1, \ldots, b_n\boldsymbol{A}_n]$.

This problem is an instance of (P) with $\boldsymbol{F}(\mathbf{x}, \mathbf{y}) = \frac{1}{n}[\bar{\boldsymbol{A}}\mathbf{y}, \mathbf{1} - \bar{\boldsymbol{A}}^\top \mathbf{x}] \in \mathbb{R}^{d+n}$, and $g(\mathbf{x}, \mathbf{y}) = \lambda_1 \|\mathbf{x}\|_1 + \frac{\lambda_2}{2}\|\mathbf{x}\|_2^2 + \sum_{j=1}^n \iota_{[-1,0]}(y_j)$. We evaluate all methods on the LIBSVM datasets [8].

Figure 1 plots $f(\mathbf{x}) - f^\star$ versus the number of full data passes. Across the three datasets, `ADUCA` consistently achieves either the fastest or competitive-with-the-fastest reduction in suboptimality. This advantage can be attributed to its cyclic updates combined with locally estimated adaptive step sizes that leverage both full-operator and block coordinate operator information, enabling `ADUCA` to exploit block structure while avoiding the per-epoch line-search overhead of `CODER-LineSearch`. The first row of plots in Figure 1 compares `ADUCA` to other methods implemented using standard ℓ_2 geometry; the second row of plots is for the implementation of all methods using the Euclidean norms rescaled by $\boldsymbol{\Lambda}$, $\|\cdot\|_{\boldsymbol{\Lambda}}$, to evaluate whether the performance improvements of `ADUCA` are purely due to the row/column rescaling by $\boldsymbol{\Lambda}$. We see that this is not the case, since a similar rescaling does not consistently benefit other algorithms and can, in fact, harm their performance (consider, for instance, the performance of `GRAAL` on the gisette dataset, in the middle column of Figure 1).

7 Conclusion

We proposed `ADUCA`, a cyclic block coordinate method for generalized monotone variational inequalities that combines locally adaptive step sizes with a one-cycle delayed-update mechanism, reducing synchronization across blocks and enabling parallel or distributed implementations. Our analysis provides global oracle-complexity bounds in terms of local geometry on the algorithmic trajectory, without knowledge of global Lipschitz constants. Future directions include asynchronous implementations, data-dependent block partitions, and extensions to weak Minty variational inequalities and stochastic settings.

Acknowledgments

This work was supported in part by the Air Force Office of Scientific Research under award number FA9550-24-1-0076 and by the U.S. Office of Naval Research under contract number N00014-22-1-2348. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Department of Defense.

References

[1] Ahmet Alacaoglu, Axel Böhm, and Yura Malitsky. Beyond the golden ratio for variational inequality algorithms. *Journal of Machine Learning Research*, 24(172):1–33, 2023.

[2] ASU Trans+AI Lab. DTALite: An open-source analysis, modeling, and simulation library. `https://github.com/asu-trans-ai-lab/DTALite`, 2026. Accessed: 2026-05-07.

[3] Francis Bach and Kfir Y Levy. A universal algorithm for variational inequalities adaptive to smoothness and noise. In *Proc. COLT'19*, 2019.

[4] Amir Beck and Luba Tetruashvili. On the convergence of block coordinate descent type methods. *SIAM Journal on Optimization*, 23(4):2037–2060, 2013.

[5] Ekaterina Borodich and Dmitry Kovalev. Nesterov finds GRAAL: Optimal and adaptive gradient method for convex optimization. In *Proc. ICLR'26*, 2026.

[6] Xufeng Cai, Cheuk Yin Lin, and Jelena Diakonikolas. Tighter convergence bounds for shuffled SGD via primal-dual perspective. In *Proc. NeurIPS'24*, 2024.

[7] Xufeng Cai, Chaobing Song, Stephen Wright, and Jelena Diakonikolas. Cyclic block coordinate descent with variance reduction for composite nonconvex optimization. In *Proc. ICML'23*, 2023.

[8] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, 2011.

[9] Yat Tin Chow, Tianyu Wu, and Wotao Yin. Cyclic coordinate-update algorithms for fixed-point problems: Analysis and applications. *SIAM Journal on Scientific Computing*, 39(4):A1280–A1300, 2017.

[10] Stella Dafermos. Traffic equilibrium and variational inequalities. *Transportation Science*, 14(1):42–54, 1980.

[11] Jelena Diakonikolas. Halpern iteration for near-optimal and parameter-free monotone inclusion and strong solutions to variational inequalities. In *Proc. COLT'20*, 2020.

[12] Jelena Diakonikolas. A block coordinate and variance-reduced method for generalized variational inequalities of Minty type. *ACM/IMS Journal of Data Science*, 2(2):1–30, 2025.

[13] Jelena Diakonikolas, Constantinos Daskalakis, and Michael I Jordan. Efficient methods for structured nonconvex-nonconcave min-max optimization. In *Proc. AISTATS'21*, 2021.

[14] Jelena Diakonikolas and Puqian Wang. Potential function-based framework for minimizing gradients in convex and min-max optimization. *SIAM Journal on Optimization*, 32(3):1668–1697, 2022.

[15] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(7), 2011.

[16] Alina Ene, Huy L. Nguyen, and Adrian Vladu. Adaptive gradient methods for constrained convex optimization and variational inequalities. In *Proc. AAAI'21*, 2021.

[17] Alina Ene and Huy Le Nguyen. Adaptive and universal algorithms for variational inequalities with optimal convergence. In *Proc. AAAI'22*, 2022.

[18] Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems*. Springer, 2003.

[19] Jerome H Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33:1–22, 2010.

[20] Noah Golowich, Sarath Pattathil, Constantinos Daskalakis, and Asuman Ozdaglar. Last iterate is slower than averaged iterate in smooth convex-concave saddle point problems. In *Proc. COLT'20*, 2020.

[21] Mert Gürbüzbalaban, Asuman Ozdaglar, Pablo A Parrilo, and N Denizcan Vanli. When cyclic coordinate descent outperforms randomized coordinate descent. In *Proc. NeurIPS'17*, 2017.

[22] Stefan Kaczmarz. Angenäherte Auflösung von systemen linearer Gleichungen. *Bull. Int. Acad. Pol. Sci. Let., Cl. Sci. Math. Nat.*, pages 355–357, 1937.

[23] David Kinderlehrer and Guido Stampacchia. *An introduction to variational inequalities and their applications*. SIAM, 2000.

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

[25] Georgios Kotsalis, Guanghui Lan, and Tianjiao Li. Simple and optimal methods for stochastic variational inequalities, I: Operator extrapolation. *SIAM Journal on Optimization*, 32(3):2041–2073, 2022.

[26] Guanghui Lan and Tianjiao Li. Auto-conditioned primal-dual hybrid gradient method and alternating direction method of multipliers. *arXiv preprint arXiv:2410.01979*, 2024.

[27] Puya Latafat, Andreas Themelis, Lorenzo Stella, and Panagiotis Patrinos. Adaptive proximal algorithms for convex optimization under local Lipschitz continuity of the gradient. *Mathematical Programming*, 213(1):433–471, 2025.

[28] Puya Latafat, Andreas Themelis, Silvia Villa, and Panagiotis Patrinos. On the convergence of proximal gradient methods for convex simple bilevel optimization. *Journal of Optimization Theory and Applications*, 204(3):51, 2025.

[29] Tianjiao Li and Guanghui Lan. A simple uniformly optimal method without line search for convex optimization. *Mathematical Programming*, 2025.

[30] Cheuk Yin Lin, Chaobing Song, and Jelena Diakonikolas. Accelerated cyclic coordinate dual averaging with extrapolation for composite convex optimization. In *Proc. ICML'23*, 2023.

[31] Yura Malitsky. Golden ratio algorithms for variational inequalities. *Mathematical Programming*, 184(1):383–410, 2020.

[32] Yura Malitsky and Konstantin Mishchenko. Adaptive gradient descent without descent. In *Proc. ICML'20*, 2020.

[33] Yura Malitsky and Konstantin Mishchenko. Adaptive proximal gradient method for convex optimization. In *Proc. NeurIPS'24*, 2024.

[34] Rahul Mazumder, Jerome H Friedman, and Trevor Hastie. SparseNet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, 106(495):1125–1138, 2011.

[35] Arkadi Nemirovski. Prox-method with rate of convergence $O(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

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

[37] Yurii Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109(2):319–344, 2007.

[38] Yurii Nesterov and Laura Scrimali. Solving strongly monotone variational and quasi-variational inequalities. *Discrete and Continuous Dynamical Systems*, 31(4):1383–1396, 2011.

[39] E. E. Osborne. On pre-conditioning of matrices. *Journal of the ACM*, 7(4):338–345, 1960.

[40] Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, 185(1-2):1–35, 2021.

[41] Path4GMNS Contributors. Path4GMNS: An open-source path engine for networks encoded in GMNS. https://github.com/jdlph/Path4GMNS, 2026. Accessed: 2026-05-07.

[42] Reza Rahimi Baghbadorani, Peyman Mohajerin Esfahani, and Sergio Grammatico. A Frank-Wolfe algorithm for strongly monotone variational inequalities. *Operations Research Letters*, 65:107388, 2026.

[43] Scott Smith, Ian Berg, and Ci Yang. General modeling network specification: Documentation, software, and data. John A. Volpe National Transportation Systems Center. https://doi.org/10.21949/1518740, 2020. Apache License 2.0.

[44] Chaobing Song and Jelena Diakonikolas. Cyclic coordinate dual averaging with extrapolation. *SIAM Journal on Optimization*, 33(4):2935–2961, 2023.

[45] Chaobing Song, Cheuk Yin Lin, Stephen Wright, and Jelena Diakonikolas. Coordinate linear variance reduction for generalized linear programming. In *Proc. NeurIPS'22*, 2022.

[46] Jaewook J Suh and Shiqian Ma. An adaptive and parameter-free Nesterov's accelerated gradient method for convex optimization. *arXiv preprint arXiv:2505.11670*, 2025.

[47] Ruoyu Sun and Yinyu Ye. Worst-case complexity of cyclic coordinate descent: $O(n^2)$ gap with randomized version. *Mathematical Programming*, 185(1):487–520, 2021.

[48] Transportation Networks for Research Core Team. Transportation networks for research. https://github.com/bstabler/TransportationNetworks, 2026. Accessed: 2026-05-07.

[49] Maria-Luiza Vladarean, Yura Malitsky, and Volkan Cevher. A first-order primal-dual method with adaptivity to local smoothness. In *Proc. NeurIPS'21*, 2021.

[50] xyluo25, ASU Trans+AI Lab, Xuesong Zhou, and Dustin Carlino. xyluo25/grid2demand: v0.5.1. Zenodo. https://doi.org/10.5281/zenodo.12591678, 2024.

[51] Farzad Yousefian, Angelia Nedić, and Uday V Shanbhag. On stochastic mirror-prox algorithms for stochastic Cartesian variational inequalities: Randomized block coordinate and optimal averaging schemes. *Set-Valued and Variational Analysis*, 26:789–819, 2018.

[52] Xuesong Zhou and Jeffrey Taylor. DTALite: A queue-based mesoscopic traffic simulator for fast model evaluation and calibration. *Cogent Engineering*, 1(1):961345, 2014.

A Complexity comparison with representative algorithms for variational inequalities

Table 1: Comparison of representative methods for monotone variational inequalities. "Auto-conditioned" means that the update rule does not require global Lipschitz or block-Lipschitz constants and does not use repeated per-iteration or per-cycle line-search.

Method	Monotone	Strongly monotone	Block-coordinate	Auto-conditioned
Extragradient / Mirror-Prox / Dual Extrapolation [24, 35, 37, 38]	$\mathcal{O}\Big(\frac{LD^2}{\epsilon}\Big)$	$\mathcal{O}\Big(\frac{L}{\mu}\log\frac{D}{\epsilon}\Big)$	$\times$	$\times$
`GRAAL` [1, 31]	$\mathcal{O}\Big(\frac{L_{\mathcal{C}}D^2}{\epsilon}\Big)$	$\mathcal{O}\Big(\frac{L_{\mathcal{C}}^2}{\mu^2}\log\frac{D}{\epsilon}\Big)$	$\times$	✓
`CODER` [44]	$\mathcal{O}\Big(\frac{\widehat{L}D^2}{\epsilon}\Big)$	$\mathcal{O}\Big(\frac{\widehat{L}}{\mu}\log\frac{\widehat{L}D}{\epsilon}\Big)$	✓	$\times$
`ADUCA` (ours)	$\mathcal{O}\Big(\frac{\mathcal{L}_{\mathcal{C}}\mathcal{B}_{\mathcal{U}}}{\epsilon}\Big)$	$\mathcal{O}\Big(\frac{\mathcal{L}_{\mathcal{C}}}{\mu}\log\frac{R_*}{\epsilon}\log\frac{\mathcal{L}_{\mathcal{C}}\mathcal{B}_{\mathcal{U}}}{\epsilon}\Big)$	✓	✓

Notation. For classical full-vector methods, L denotes a global Lipschitz constant and D denotes the relevant diameter or initial-distance term. For `GRAAL`, $L_{\mathcal{C}}$ denotes a trajectory-local full-operator Lipschitz constant on a compact set $\mathcal{C}$. For `CODER`, $\widehat{L}$ denotes the global Mahalanobis cyclic block-Lipschitz aggregate. For `ADUCA`, $\mathcal{C}$ is a compact set containing all points generated by the bounded trajectory. Given local block-Lipschitz matrices $\{\boldsymbol{Q}_{\mathcal{C}}^i\}_{i=1}^m$ on $\mathcal{C}$ and their cyclic lower-triangular truncations $\{\widehat{\boldsymbol{Q}}_{\mathcal{C}}^i\}_{i=1}^m$, define

$$L_{\mathcal{C}} := \|\boldsymbol{\Lambda}^{-1/2}(\sum_{i=1}^m \boldsymbol{Q}_{\mathcal{C}}^i)\boldsymbol{\Lambda}^{-1/2}\|^{1/2}, \qquad \widehat{L}_{\mathcal{C}} := \|\boldsymbol{\Lambda}^{-1/2}(\sum_{i=1}^m \widehat{\boldsymbol{Q}}_{\mathcal{C}}^i)\boldsymbol{\Lambda}^{-1/2}\|^{1/2},$$

and $\mathcal{L}_{\mathcal{C}} := \max\{L_{\mathcal{C}}/C, \widehat{L}_{\mathcal{C}}/\hat{C}\}$, where $C := \frac{\eta\rho}{2}\sqrt{\frac{\tau\beta}{3(1+\rho\beta)}}$, $\hat{C} := \frac{\eta}{2}\sqrt{\frac{(1-\tau)\rho}{2}}$.

For a compact test set $\mathcal{U} \subset \mathrm{dom}(g)$, let $\mathcal{B}_{\mathcal{U}} := \sup_{\mathbf{u}\in\mathcal{U}} \mathcal{B}_0(\mathbf{u})$, where $\mathcal{B}_0(\mathbf{u}) := \frac{1+\mu a_0}{2(1-\beta)}\|\mathbf{u}-\mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2 + (\frac{\eta^2\beta^2}{2} + \frac{1+\beta\mu a_0}{2\beta})\|\mathbf{u}_1-\mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2$. Finally, R_* is the explicit trajectory-radius quantity from the proof of Corollary 1: with $G_0 := \min_{\xi_0\in\partial g(\mathbf{u}_0)}\|\boldsymbol{F}(\mathbf{u}_0)+\xi_0\|_{\boldsymbol{\Lambda}^{-1}}$ and $a_{\max}$,

$$R_* := \Big(2(1+\mu a_{\max})\big(\|\mathbf{u}_0-\mathbf{u}^*\|_{\boldsymbol{\Lambda}} + a_{\max}G_0\big)^2 + 2(1-\beta)\Big(\eta^2\beta^2 + \frac{1+\beta\mu a_{\max}}{\beta}\Big)a_{\max}^2 G_0^2\Big)^{1/2}.$$

If `ADUCA` is initialized with Algorithm 2, here $a_{\max} = a_{\mathrm{init}}$, where a_{init} is defined in Line 6 of Algorithm 2.

Complexity explanation. The strong convexity of g and strong monotonicity of $\boldsymbol{F}$ regimes in these comparisons should be interpreted on the same footing. Indeed, in a composite VI, these two conditions can be exchanged through the standard quadratic shift. This equivalence is discussed in more detail after Corollary 1. Therefore, whether a prior result states its faster rate under strong convexity of g or under strong monotonicity of $\boldsymbol{F}$, the comparison is made under equivalent curvature assumptions and is thus directly comparable.

Classical extragradient, Mirror-Prox, and dual extrapolation methods achieve the optimal monotone order $O(1/\epsilon)$, but their standard guarantees are phrased using a global Lipschitz constant and full-vector updates [24, 35, 37]. Under strong monotonicity, the strongly monotone extrapolation framework has the usual condition-number dependence

$$\mathcal{O}\Big(\frac{L}{\mu}\log\frac{D}{\epsilon}\Big)$$

[38]. The monotone $O(1/\epsilon)$ order is optimal in the standard first-order oracle model, due to known lower bounds that apply even to bilinear problems [40].

GRAAL is adaptive to local full-operator smoothness and does not require a global Lipschitz constant in its update rule [1, 31]. However, it is a full-vector method and does not exploit cyclic block-coordinate structure. In the monotone case, the complexity statement used in Table 1 should be attributed to the improved analysis in [1], which removes the artificial maximum step-size parameter from the adaptive GRAAL analysis. Thus the monotone entry should be read as a local, trajectory-dependent $O(1/\epsilon)$ guarantee, rather than as a guarantee requiring a globally imposed step-size cap. The faster-rate entry for GRAAL should be interpreted through the error-bound result in the original paper [31], not as the standard strongly-monotone extragradient rate. For the natural residual

$$r(z,\lambda) := z - \mathrm{prox}_{\lambda g}(z - \lambda F(z)),$$

and a fixed $\lambda > 0$, [31] assumes the local error bound

$$\mathrm{dist}(z, S) \le \mu_{\mathrm{EB}} \|r(z,\lambda)\|.$$

The linear-rate statement in [31] yields

$$K = \mathcal{O}\Big(\mu_{\mathrm{EB}}^2 \log \frac{D}{\epsilon}\Big).$$

When F is μ-strongly monotone and L-Lipschitz, the above error-bound parameter is itself of condition-number order. Indeed, we have

$$\mu_{\mathrm{EB}}(\lambda) \le \frac{1+\lambda L}{\lambda \mu},$$

which is the standard natural-residual error bound for strongly monotone and Lipschitz VIs; see, e.g., [18, Theorem 2.3.3]. Since all natural residuals $r(\cdot,\lambda)$ are equivalent for any $\lambda > 0$ (see [18, Proposition 10.3.6]), taking $\lambda = \Theta(1/L)$ gives

$$\mu_{\mathrm{EB}} = \mathcal{O}\Big(\frac{L}{\mu}\Big).$$

Consequently, the GRAAL linear-rate entry becomes

$$K = \mathcal{O}\Big(\frac{L^2}{\mu^2} \log \frac{D}{\epsilon}\Big).$$

This is why the GRAAL strongly-monotone entry is stated with the squared condition-number dependence L^2/μ^2.

CODER is the closest cyclic block-coordinate predecessor. It obtains the optimal monotone $O(1/\epsilon)$ order and a linear rate when g is strongly convex, but its main guarantees depend on a global block-Lipschitz aggregate $\widehat{L}$. Its line-search variant can reduce tuning but uses repeated per-cycle line-search [44].

ADUCA combines the two advantages: it is cyclic block-coordinate like CODER and locally adaptive like GRAAL. Its oracle complexity depends only on local constants $L_{\mathcal{C}}$ and $\widehat{L}_{\mathcal{C}}$ on the compact algorithmic trajectory, not on global smoothness constants. The monotone bound is

$$K = \mathcal{O}\Big(\frac{\mathcal{L}_{\mathcal{C}} \mathcal{B}_{\mathcal{U}}}{\epsilon}\Big),$$

and under strong convexity or restricted strong monotonicity the near-linear alternative is

$$K = \mathcal{O}\Big(\frac{\mathcal{L}_{\mathcal{C}}}{\mu} \log \frac{R_*}{\epsilon} \log \frac{\mathcal{L}_{\mathcal{C}} \mathcal{B}_{\mathcal{U}}}{\epsilon}\Big).$$

B Review of standard facts

The short facts collected here are used only inside the proofs. For convenience, we also recall the merit functions

$$h_k(\mathbf{u}) := a_k\big(\langle \boldsymbol{F}(\mathbf{u}_k), \mathbf{u}_k - \mathbf{u}\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\big)$$

and the weights

$$\theta_k = 1/\prod_{i=0}^{k-1} \omega_i \ (k \geq 1), \qquad \theta_k = \theta_{k-1}/\omega_{k-1} \ (k \geq 1,\, \theta_0 = 1).$$

The restricted-gap averaging fact built on these definitions is restated and proved in Section 4, since it is used directly in the main convergence theorem.

Fact B.1. *For any $x, y \in \mathbb{R}$ and any $\alpha > 0$,*

$$(x+y)^2 \geq \frac{\alpha}{1+\alpha} x^2 - \alpha y^2.$$

Proof. This is a direct consequence of Young's inequality:

$$(a+b)^2 \leq \Big(1 + \frac{1}{\alpha}\Big) a^2 + (1+\alpha) b^2.$$

Apply it to $a = x + y$ and $b = -y$ and rearrange. □

Fact B.2. *For any Euclidean norm $\|\cdot\|$, any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, and any $\alpha > 0$,*

$$\big\|\alpha\mathbf{x} + (1-\alpha)\mathbf{y}\big\|^2 = \alpha\|\mathbf{x}\|^2 + (1-\alpha)\|\mathbf{y}\|^2 - \alpha(1-\alpha)\|\mathbf{x}-\mathbf{y}\|^2.$$

Proof. Expand the square and use bilinearity of the inner product:

$$\begin{aligned}\big\|\alpha\mathbf{x} + (1-\alpha)\mathbf{y}\big\|^2 &= \alpha^2\|\mathbf{x}\|^2 + (1-\alpha)^2\|\mathbf{y}\|^2 + 2\alpha(1-\alpha)\langle\mathbf{x},\mathbf{y}\rangle \\ &= \alpha\|\mathbf{x}\|^2 + (1-\alpha)\|\mathbf{y}\|^2 - \alpha(1-\alpha)\big(\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - 2\langle\mathbf{x},\mathbf{y}\rangle\big) \\ &= \alpha\|\mathbf{x}\|^2 + (1-\alpha)\|\mathbf{y}\|^2 - \alpha(1-\alpha)\|\mathbf{x}-\mathbf{y}\|^2.\end{aligned}$$

□

Fact B.3. *Let $g : \mathbb{R}^d \to \mathbb{R}$ be μ-strongly convex with respect to $\|\cdot\|_{\mathbf{\Lambda}}$, let $a > 0$, and define*

$$\mathbf{u}^+ = \operatorname*{argmin}_{\mathbf{v}\in\mathbb{R}^d} \Big\{ a\big\langle \bar{\boldsymbol{F}}, \mathbf{v}\big\rangle + a g(\mathbf{v}) + \frac{1}{2}\|\mathbf{v}-\mathbf{u}\|_{\mathbf{\Lambda}}^2 \Big\}.$$

Then, for any $\mathbf{v} \in \mathbb{R}^d$,

$$a\big(\big\langle \bar{\boldsymbol{F}}, \mathbf{u}^+ - \mathbf{v}\big\rangle + g(\mathbf{u}^+) - g(\mathbf{v})\big) \leq \frac{1}{2}\|\mathbf{v}-\mathbf{u}\|_{\mathbf{\Lambda}}^2 - \frac{1}{2}\|\mathbf{u}^+-\mathbf{u}\|_{\mathbf{\Lambda}}^2 - \frac{a\mu+1}{2}\|\mathbf{v}-\mathbf{u}^+\|_{\mathbf{\Lambda}}^2.$$

Proof. Let

$$M(\mathbf{v}) := a\big\langle \bar{\boldsymbol{F}}, \mathbf{v}\big\rangle + a g(\mathbf{v}) + \frac{1}{2}\|\mathbf{v}-\mathbf{u}\|_{\mathbf{\Lambda}}^2.$$

Since M is $(a\mu+1)$-strongly convex and minimized at $\mathbf{u}^+$, for any $\mathbf{v}$,

$$M(\mathbf{v}) \geq M(\mathbf{u}^+) + \frac{a\mu+1}{2}\|\mathbf{v}-\mathbf{u}^+\|_{\mathbf{\Lambda}}^2.$$

Expanding the definition of M and rearranging yields the claim. □

C Additional remarks and adaptive step-size conditions

C.1 Additional Remarks

While ADUCA is conceptually inspired by two prior lines of work—a cyclic coordinate algorithm CODER [44] and the adaptive full-operator algorithm GRAAL [1, 31]—the resulting algorithm is different and does not lead to either of these two algorithms as a special case. Specifically, when $m = 1$, ignoring the cyclic updates and comparing to GRAAL, ADUCA uses operator information delayed by one iteration. On the other hand, compared to CODER, ADUCA includes an additional iterate extrapolation step of Line 9 from Algorithm 1 and does not rely on dual averaging.

Regarding the locally adaptive step sizes, compared to GRAAL, our scheme introduces an additional upper bound with respect to $\hat{L}_k$ in every epoch in the step-size selection condition to account for cyclic updates, distinguishing it from [31]. In the strongly convex case, our scheme (11) and the corresponding analysis differ from [31], and directly lead to near-linear convergence without requiring a lower bound on the step sizes as in [31]. Finally, to control errors from the cyclic updates, the possible geometric growth of step-sizes here is slightly smaller than the ones in [1, 32], with a smaller ρ_0 imposed on it.

Finally, there is a separate class of adaptive step-size methods based on AdaGrad, which was originally developed for minimization problems [15]. In the context of monotone VIs considered in our work, relevant work includes [3, 16, 17]. The primary limitation of AdaGrad-type adaptivity is that the adaptive step size is non-increasing by design. As a result, such algorithms cannot exploit regions where larger step sizes may be admissible. In deterministic or low-noise regimes, this monotone decay can be overly conservative and lead to slower empirical convergence relative to methods that adapt to *local* geometry via explicit local Lipschitz estimation and allowing increasing step-sizes, as in our work by $\rho_0 > 1$.

C.2 Complete set of step size conditions

For $k \geq 0$, define

$$\omega_k := \frac{1+\rho\beta\mu a_k}{1+\mu a_k} \quad \text{and} \quad \phi_k := \sqrt{\frac{a_k(\beta\mu a_{k-1}+1)}{\beta a_{k-1}}}. \tag{10}$$

Let $\tau \in (0,1)$. The complete set of step size conditions that arise from our analysis and are together sufficient for guaranteeing the claimed algorithm convergence is:

$$\begin{aligned}
a_k &\leq \min\Big\{\rho a_{k-1},\ \frac{\beta(1+\beta)(1-\gamma)a_{k-1}}{[1-(1-\gamma)\beta^2\mu a_{k-1}]_+},\\
&\qquad \frac{\eta}{2}\min\Big\{\frac{\sqrt{\tau}\,\omega_{k-1}}{\sqrt{3}L_k\sqrt{1+\omega_{k-1}}},\ \frac{\sqrt{(1-\tau)\omega_{k-1}/2}}{\hat{L}_k}\Big\}\phi_{k-1}\Big\}, && \text{for } k \geq 1,\\
a_k &\leq \frac{\eta\sqrt{\tau}}{2\sqrt{3}}\frac{\omega_{k-1}\sqrt{\omega_{k-2}}}{\hat{L}_{k-1}\sqrt{1+\omega_{k-1}}}\phi_{k-2}, && \text{for } k \geq 2,\\
a_k &\leq \frac{\eta\sqrt{\tau}}{2\sqrt{3}}\frac{\omega_{k-1}}{\hat{L}_{k-2}\sqrt{1+\omega_{k-1}}}\sqrt{\frac{\omega_{k-3}}{\omega_{k-2}}}\frac{a_{k-1}}{a_{k-2}}\phi_{k-3}, && \text{for } k \geq 3.
\end{aligned} \tag{11}$$

C.3 Simplified step size conditions

The step sizes in (11) can be simplified by (i) defining $\rho_0 := \min\{\rho, \beta(1+\beta)(1-\gamma)\}$ so the first two conditions are implied by $a_k \leq \rho_0 a_{k-1}$ and (ii) arguing that the conditions from the last two inequalities (for $k \geq 2$ and $k \geq 3$) are implied by the first set of inequalities for an appropriate choice of $\tau = \tau(\rho_0)$. We summarize such simplified step sizes in the lemma below.

Lemma C.1 (Simplified step sizes)**.** *Let $\rho_0 := \min\{\rho, \beta(1+\beta)(1-\gamma)\}$ and $\tau = \frac{3\rho_0^2(1+\rho\beta)}{2(\rho\beta)^2+3\rho_0^2(1+\rho\beta)} \in (0,1)$. Then, each of the two step size conditions for $k \geq 1$ stated in* (12) *and* (13) *implies the complete step size conditions in* (11)*.*

$$a_k \leq \min\Big\{\rho_0 a_{k-1},\ \frac{\eta}{2}\min\Big\{\frac{\sqrt{\tau}\,\omega_{k-1}}{\sqrt{3}L_k\sqrt{1+\omega_{k-1}}},\ \frac{\sqrt{(1-\tau)\omega_{k-1}}}{\sqrt{2}\hat{L}_k}\Big\}\phi_{k-1}\Big\}, \tag{12}$$

$$a_k \leq \min\Big\{\rho_0 a_{k-1},\ \frac{\eta}{2\sqrt{\beta}}\min\Big\{\frac{\sqrt{\tau}\,\rho\beta}{\sqrt{3}L_k\sqrt{1+\rho\beta}},\ \frac{\sqrt{(1-\tau)\rho\beta}}{\sqrt{2}\hat{L}_k}\Big\}\sqrt{\frac{a_{k-1}}{a_{k-2}}}\Big\}. \tag{13}$$

Proof. Observe first that the last condition for $k \geq 1$ in (11) implies a lower bound on ϕ_{k-1}. Specifically, as $a_k \leq \frac{\eta}{2}\frac{\sqrt{(1-\tau)\omega_{k-1}/2}}{\hat{L}_k}\phi_{k-1}$, we have

$$\phi_{k-1} \geq a_k\hat{L}_k\frac{2\sqrt{2}}{\eta\sqrt{(1-\tau)\omega_{k-1}}}, \quad \forall k \geq 1. \tag{14}$$

We now argue, using (14), that the right-hand sides of inequalities in the last two lines of (11) are bounded below by one of the first two conditions in (11). First,

$$\begin{aligned}\frac{\eta\sqrt{\tau}}{2\sqrt{3}}\frac{\omega_{k-1}\sqrt{\omega_{k-2}}}{\hat{L}_{k-1}\sqrt{1+\omega_{k-1}}}\phi_{k-2} &\geq \sqrt{\frac{2}{3}}\cdot\sqrt{\frac{\tau}{1-\tau}}\cdot\frac{\omega_{k-1}}{\sqrt{1+\omega_{k-1}}}a_{k-1}\\ &\geq \sqrt{\frac{2}{3}}\cdot\sqrt{\frac{\tau}{1-\tau}}\cdot\frac{\rho\beta}{\sqrt{1+\rho\beta}}a_{k-1},\end{aligned}$$

where the last inequality comes from $\omega_{k-1} \in (\rho\beta, 1]$. The last inequality is bounded below by $\rho_0 a_{k-1}$ whenever $\sqrt{\frac{2}{3}}\cdot\sqrt{\frac{\tau}{1-\tau}}\cdot\frac{\rho\beta}{\sqrt{1+\rho\beta}} \geq \rho_0$, which, solving for τ, is equivalent to the condition $\tau \geq \frac{3\rho_0^2(1+\rho\beta)}{2(\rho\beta)^2+3\rho_0^2(1+\rho\beta)}$.

For the remaining condition in (11), we similarly argue that under an appropriate choice of τ, the right-hand side of the last inequality in (11) is bounded below by $\rho_0 a_{k-1}$, so this condition must be satisfied as $a_k \leq \rho_0 a_{k-1}$ by the first condition in (11). To see this, again using (14), we have

$$\begin{aligned}&\frac{\eta\sqrt{\tau}}{2\sqrt{3}}\frac{\omega_{k-1}}{\hat{L}_{k-2}\sqrt{1+\omega_{k-1}}}\sqrt{\frac{\omega_{k-3}}{\omega_{k-2}}}\frac{a_{k-1}}{a_{k-2}}\phi_{k-3}\\ &\geq \sqrt{\frac{2}{3}}\cdot\sqrt{\frac{\tau}{1-\tau}}\cdot\frac{\omega_{k-1}}{\sqrt{\omega_{k-2}(1+\omega_{k-1})}}a_{k-1}\\ &\geq \sqrt{\frac{2}{3}}\cdot\sqrt{\frac{\tau}{1-\tau}}\cdot\frac{\rho\beta}{\sqrt{1+\rho\beta}}a_{k-1},\end{aligned}$$

where we used that $\omega_{k-2} \leq 1$ and $\frac{\omega_{k-1}}{\sqrt{1+\omega_{k-1}}} \geq \frac{\rho\beta}{\sqrt{1+\rho\beta}}$. As already argued, the last inequality is further bounded below by $\rho_0 a_{k-1}$ for $\tau \geq \frac{3\rho_0^2(1+\rho\beta)}{2(\rho\beta)^2+3\rho_0^2(1+\rho\beta)}$.

Setting $\tau = \frac{3\rho_0^2(1+\rho\beta)}{2(\rho\beta)^2+3\rho_0^2(1+\rho\beta)} < 1$, (11) simplifies to requiring, for $k \geq 1$, that

$$a_k \leq \min\Big\{\rho_0 a_{k-1}, \frac{\eta}{2}\min\Big\{\frac{\sqrt{\tau}\,\omega_{k-1}}{\sqrt{3}L_k\sqrt{1+\omega_{k-1}}}, \frac{\sqrt{(1-\tau)\omega_{k-1}}}{\sqrt{2}\hat{L}_k}\Big\}\phi_{k-1}\Big\}.$$

Notice that because both ω_{k-1} and ϕ_{k-1} depend on μ, this choice of step sizes is valid only for the known value of μ. When μ is not known, we can simply use $\mu \geq 0$ and $\omega_{k-1} \in (\rho\beta, 1]$ to bound below ω_{k-1}, $\frac{\omega_{k-1}}{\sqrt{1+\omega_{k-1}}}$, and ϕ_{k-1}, and obtain the sufficient conditions for the step size stated in (3). □

D Omitted proofs from Section 4

Proof of Lemma 1. By convexity of g and monotonicity of $\boldsymbol{F}$,

$$\begin{aligned}\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) &= \big\langle \boldsymbol{F}(\mathbf{u}), \hat{\mathbf{u}}_K - \mathbf{u}\big\rangle + g(\hat{\mathbf{u}}_K) - g(\mathbf{u})\\ &\leq \frac{1}{A_K}\sum_{k=1}^{K}\theta_k a_k\Big(\big\langle \boldsymbol{F}(\mathbf{u}), \mathbf{u}_k - \mathbf{u}\big\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\Big)\\ &\leq \frac{1}{A_K}\sum_{k=1}^{K}\theta_k a_k\Big(\big\langle \boldsymbol{F}(\mathbf{u}_k), \mathbf{u}_k - \mathbf{u}\big\rangle + g(\mathbf{u}_k) - g(\mathbf{u})\Big)\\ &= \frac{\sum_{k=1}^{K}\theta_k h_k(\mathbf{u})}{A_K}.\end{aligned}$$

□

Proof of Lemma 2. Plugging in the definition of $\bar{\boldsymbol{F}}_k$ from Line 8 of Algorithm 1,

$$h_k^{(1)}(\mathbf{u}) = a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}\big\rangle - a_{k-1}\omega_{k-1}\big\langle \boldsymbol{F}(\mathbf{u}_{k-1}) - \tilde{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}\big\rangle.$$

Now add and subtract $a_k\big\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}_{k+1}\big\rangle$. □

Proof of Lemma 3. Observe that we can decompose $h_k^{(2)}(\mathbf{u})$ as

$$\begin{aligned} h_k^{(2)}(\mathbf{u}) &= a_k\big(\langle \bar{\boldsymbol{F}}_k, \mathbf{u}_{k+1} - \mathbf{u}\rangle + g(\mathbf{u}_{k+1}) - g(\mathbf{u})\big) \\ &\quad + \frac{a_k}{a_{k-1}} a_{k-1}\big(\langle \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle + g(\mathbf{u}_k) - g(\mathbf{u}_{k+1})\big) \\ &\quad + a_k\langle \bar{\boldsymbol{F}}_k - \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle. \end{aligned} \tag{15}$$

Denote $\mathcal{T}_1 := a_k\big(\langle \bar{\boldsymbol{F}}_k, \mathbf{u}_{k+1} - \mathbf{u}\rangle + g(\mathbf{u}_{k+1}) - g(\mathbf{u})\big)$ and $\mathcal{T}_2 := \frac{a_k}{a_{k-1}} a_{k-1}\big(\langle \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle + g(\mathbf{u}_k) - g(\mathbf{u}_{k+1})\big)$, the terms in the first and the second line of the right-hand side of (15). To prove the lemma, we bound above $\mathcal{T}_1 + \mathcal{T}_2$.

Observe first that by the update rule in Line 10 of Algorithm 1 and Fact B.3, we have

$$\mathcal{T}_1 \le \frac{1}{2}\big(\|\mathbf{v}_k - \mathbf{u}\|_{\boldsymbol{\Lambda}}^2 - \|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 - (\mu a_k + 1)\|\mathbf{u}_{k+1} - \mathbf{u}\|_{\boldsymbol{\Lambda}}^2\big), \tag{16}$$

$$\mathcal{T}_2 \le \frac{a_k}{2a_{k-1}}\big(\|\mathbf{u}_{k+1} - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2 - \|\mathbf{u}_k - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2 - (\mu a_{k-1} + 1)\|\mathbf{u}_k - \mathbf{u}_{k+1}\|_{\boldsymbol{\Lambda}}^2\big). \tag{17}$$

Further, by its definition, $\mathbf{v}_k = (1-\beta)\mathbf{u}_k + \beta\mathbf{v}_{k-1}$. Thus, $\|\mathbf{u} - \mathbf{v}_{k+1}\|_{\boldsymbol{\Lambda}}^2 = \|(1-\beta)(\mathbf{u} - \mathbf{u}_{k+1}) + \beta(\mathbf{u} - \mathbf{v}_k)\|_{\boldsymbol{\Lambda}}^2$. A rearrangement of the identity from Fact B.2 then gives

$$\|\mathbf{u}_{k+1} - \mathbf{u}\|_{\boldsymbol{\Lambda}}^2 = \frac{1}{1-\beta}\|\mathbf{u} - \mathbf{v}_{k+1}\|_{\boldsymbol{\Lambda}}^2 - \frac{\beta}{1-\beta}\|\mathbf{u} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 + \beta\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2. \tag{18}$$

Similarly, since $\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 = \|(1-\beta)(\mathbf{u}_{k+1} - \mathbf{u}_k) + \beta(\mathbf{u}_{k+1} - \mathbf{v}_{k-1})\|_{\boldsymbol{\Lambda}}^2$, we have

$$\|\mathbf{u}_{k+1} - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2 = \frac{1}{\beta}\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 - \frac{1-\beta}{\beta}\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2 + (1-\beta)\|\mathbf{u}_k - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2. \tag{19}$$

Plugging (18) and (19) back into (16) and (17), respectively, and summing them up, we obtain the following bound on $\mathcal{T}_1 + \mathcal{T}_2$:

$$\begin{aligned} &\mathcal{T}_1 + \mathcal{T}_2 \\ &\le \frac{1}{2(1-\beta)}\Big((1 + \beta\mu a_k)\|\mathbf{u} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 - (1 + \mu a_k)\|\mathbf{u} - \mathbf{v}_{k+1}\|_{\boldsymbol{\Lambda}}^2\Big) \\ &+ \Big(\frac{a_k}{2\beta a_{k-1}} - \frac{1 + \beta + \beta\mu a_k}{2}\Big)\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 - \frac{a_k(1 + \beta\mu a_{k-1})}{2\beta a_{k-1}}\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2, \end{aligned} \tag{20}$$

where we dropped the non-positive term $-\frac{\beta a_k}{2a_{k-1}}\|\mathbf{u}_k - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2$ from the right-hand side. It now remains to bound above $\big(\frac{a_k}{2\beta a_{k-1}} - \frac{1+\beta+\beta\mu a_k}{2}\big)\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2$.

Observe first that $\frac{a_k}{2\beta a_{k-1}} \le (1-\gamma)\frac{1+\beta+\beta\mu a_k}{2}$ would ensure $\frac{a_k}{2\beta a_{k-1}} - \frac{1+\beta+\beta\mu a_k}{2} \le -\gamma\frac{1+\beta+\beta\mu a_k}{2} \le -\frac{\gamma(1+\beta)}{2}$. A rearrangement of $\frac{a_k}{2\beta a_{k-1}} \le (1-\gamma)\frac{1+\beta+\beta\mu a_k}{2}$ gives $\big(\frac{1}{1-\gamma} - \beta^2\mu a_{k-1}\big)a_k \le \beta(1+\beta)a_{k-1}$, which is implied by (any of) our step size conditions (see (11)), and thus:

$$\Big(\frac{a_k}{2\beta a_{k-1}} - \frac{1 + \beta + \beta\mu a_k}{2}\Big)\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 \le -\frac{\gamma(1+\beta)}{2}\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2. \tag{21}$$

Now observe that we can write $\mathbf{u}_{k+1} - \mathbf{v}_k = \mathbf{u}_{k+1} - \mathbf{u}_k + \mathbf{u}_k - \mathbf{v}_k = \mathbf{u}_{k+1} - \mathbf{u}_k + \beta(\mathbf{u}_k - \mathbf{v}_{k-1})$. Thus, applying Fact B.1 with $\alpha = \omega_{k-1}$, we have

$$\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 \ge \frac{\omega_{k-1}}{1 + \omega_{k-1}}\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2 - \omega_{k-1}\beta^2\|\mathbf{u}_k - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2.$$

Taking a convex combination of the above inequality and the trivial inequality $\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 \ge \|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2$, with weights $\frac{1}{1+\beta^2}$ and $\frac{\beta^2}{1+\beta^2}$, we get

$$\begin{aligned} \|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 &\ge \frac{\omega_{k-1}}{(1+\beta^2)(1+\omega_{k-1})}\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\boldsymbol{\Lambda}}^2 \\ &\quad + \frac{\beta^2}{1+\beta^2}\big(\|\mathbf{u}_{k+1} - \mathbf{v}_k\|_{\boldsymbol{\Lambda}}^2 - \omega_{k-1}\|\mathbf{u}_k - \mathbf{v}_{k-1}\|_{\boldsymbol{\Lambda}}^2\big). \end{aligned} \tag{22}$$

To complete the proof, it now remains to combine (15), (20), (21), and (22), and group the like terms. □

Proof of Lemma 4. By Line 8 from Algorithm 1, the first two terms of $\mathcal{E}_k$ decompose as

$$\begin{aligned}
&a_k\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle + a_k\langle \bar{\boldsymbol{F}}_k - \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle \\
&= a_k\langle \boldsymbol{F}(\mathbf{u}_k) - \boldsymbol{F}(\mathbf{u}_{k-1}), \mathbf{u}_k - \mathbf{u}_{k+1}\rangle \\
&\quad + a_{k-1}\omega_{k-1}\langle \boldsymbol{F}(\mathbf{u}_{k-1}) - \tilde{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle \\
&\quad + a_k\langle \boldsymbol{F}(\mathbf{u}_{k-1}) - \tilde{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle \\
&\quad + \frac{a_k a_{k-2}\omega_{k-2}}{a_{k-1}}\langle \boldsymbol{F}(\mathbf{u}_{k-2}) - \tilde{\boldsymbol{F}}_{k-2}, \mathbf{u}_{k+1} - \mathbf{u}_k\rangle.
\end{aligned}$$

Recall here that $\mathbf{u}_i = \mathbf{u}_0$ for $i < 0$ and $\tilde{\boldsymbol{F}}_i = \boldsymbol{F}(\mathbf{u}_0)$ for $i \leq 0$.

To bound the non-zero terms on the right-hand side, we apply the Cauchy-Schwarz inequality followed by Young's inequality, which leads to the following statement: for any $\mathbf{u}, \mathbf{v}, \mathbf{u}', \mathbf{v}' \in \mathbb{R}^d$, $\delta > 0$,

$$\langle \mathbf{u}' - \mathbf{v}', \mathbf{u} - \mathbf{v}\rangle \leq \|\boldsymbol{\Lambda}^{-1/2}(\mathbf{u}' - \mathbf{v}')\|\|\boldsymbol{\Lambda}^{1/2}(\mathbf{u} - \mathbf{v})\| \leq \frac{\delta}{2}\|\mathbf{u}' - \mathbf{v}'\|^2_{\boldsymbol{\Lambda}^{-1}} + \frac{1}{2\delta}\|\mathbf{u} - \mathbf{v}\|^2_{\boldsymbol{\Lambda}}.$$

Applying this inequality to each term with parameters $\delta_1, \delta_2, \delta_3, \delta_4 > 0$, and recalling the definitions of L_k and $\hat{L}_k$, we obtain for $k \geq 3$:

$$\begin{aligned}
&a_k\langle \boldsymbol{F}(\mathbf{u}_k) - \tilde{\boldsymbol{F}}_k, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle + a_k\langle \bar{\boldsymbol{F}}_k - \bar{\boldsymbol{F}}_{k-1}, \mathbf{u}_k - \mathbf{u}_{k+1}\rangle \\
&\leq \frac{1}{2}\Big(\frac{a_k}{\delta_1} + \frac{a_k}{\delta_2} + \frac{a_{k-1}\omega_{k-1}}{\delta_3} + \frac{a_k a_{k-2}\omega_{k-2}}{a_{k-1}\delta_4}\Big)\|\mathbf{u}_{k+1} - \mathbf{u}_k\|^2_{\boldsymbol{\Lambda}} \\
&\quad + \frac{L_k^2 a_k \delta_1}{2}\|\mathbf{u}_k - \mathbf{u}_{k-1}\|^2_{\boldsymbol{\Lambda}} \\
&\quad + \frac{\hat{L}_{k-1}^2(a_k\delta_2 + a_{k-1}\omega_{k-1}\delta_3)}{2}\|\mathbf{u}_{k-1} - \mathbf{u}_{k-2}\|^2_{\boldsymbol{\Lambda}} \\
&\quad + \frac{\hat{L}_{k-2}^2 a_k a_{k-2}\omega_{k-2}\delta_4}{2a_{k-1}}\|\mathbf{u}_{k-2} - \mathbf{u}_{k-3}\|^2_{\boldsymbol{\Lambda}}.
\end{aligned}$$

For more general k, a similar expression is obtained, where we note that the terms involving δ_1 only appear for $k \geq 1$, the terms involving δ_2 and δ_3 appear for $k \geq 2$, and the terms involving δ_4 appear for $k \geq 3$ (otherwise these terms should be treated as zero). Thus, we have (with the same note about k):

$$\begin{aligned}
\theta_k\mathcal{E}_k \leq \frac{\theta_k}{2}\Big(&\frac{a_k}{\delta_1} + \frac{a_k}{\delta_2} + \frac{a_{k-1}\omega_{k-1}}{\delta_3} + \frac{a_k a_{k-2}\omega_{k-2}}{a_{k-1}\delta_4} \\
&- \frac{\gamma\omega_{k-1}(1+\beta)}{(1+\omega_{k-1})(1+\beta^2)} - \frac{a_k(1+\beta\mu a_{k-1})}{\beta a_{k-1}}\Big)\|\mathbf{u}_{k+1} - \mathbf{u}_k\|^2_{\boldsymbol{\Lambda}} \\
&+ \frac{\theta_k L_k^2 a_k \delta_1}{2}\|\mathbf{u}_k - \mathbf{u}_{k-1}\|^2_{\boldsymbol{\Lambda}} \\
&+ \frac{\theta_k \hat{L}_{k-1}^2(a_k\delta_2 + a_{k-1}\omega_{k-1}\delta_3)}{2}\|\mathbf{u}_{k-1} - \mathbf{u}_{k-2}\|^2_{\boldsymbol{\Lambda}} \\
&+ \frac{\theta_k \hat{L}_{k-2}^2 a_k a_{k-2}\omega_{k-2}\delta_4}{2a_{k-1}}\|\mathbf{u}_{k-2} - \mathbf{u}_{k-3}\|^2_{\boldsymbol{\Lambda}}.
\end{aligned} \tag{23}$$

For the terms on the right-hand side of (23) to telescope when summed over $k \geq 0$, we need that there exists a sequence of positive numbers c_k, $k \geq 0$, such that the coefficient multiplying $\|\mathbf{u}_{k+1} - \mathbf{u}_k\|^2_{\boldsymbol{\Lambda}}$ in (23) is at most $-c_k/2$, while the coefficients multiplying $\|\mathbf{u}_k - \mathbf{u}_{k-1}\|^2_{\boldsymbol{\Lambda}}$, $\|\mathbf{u}_{k-1} - \mathbf{u}_{k-2}\|^2_{\boldsymbol{\Lambda}}$, $\|\mathbf{u}_{k-2} - \mathbf{u}_{k-3}\|^2_{\boldsymbol{\Lambda}}$ are, respectively, at most $\alpha_1 c_{k-1}/2$, $\alpha_2 c_{k-2}/2$, $(1 - \alpha_1 - \alpha_2)c_{k-3}/2$, for some positive α_1, α_2 such that $\alpha_1 + \alpha_2 < 1$. Here we recall that, by the above discussion, any terms that would nominally correspond to c_i for $i < 0$ would be zero and thus can safely be ignored.

In particular, let us choose $\delta_1, \delta_2, \delta_3, \delta_4$ so the corresponding coefficients multiplying $\|\mathbf{u}_k - \mathbf{u}_{k-1}\|^2_{\boldsymbol{\Lambda}}$, $\|\mathbf{u}_{k-1} - \mathbf{u}_{k-2}\|^2_{\boldsymbol{\Lambda}}$, $\|\mathbf{u}_{k-2} - \mathbf{u}_{k-3}\|^2_{\boldsymbol{\Lambda}}$ are, respectively, $c_{k-1}/8$, $c_{k-2}/4$, and $c_{k-3}/8$. Solving the equations resulting from such conditions, we get the following expressions for $\delta_1(k \geq 1), \delta_2(k \geq$

2), $\delta_3(k \geq 2), \delta_4(k \geq 3)$:

$$\begin{aligned} \delta_1 &= \frac{c_{k-1}}{4\theta_k L_k^2 a_k}, & \delta_2 &= \frac{c_{k-2}}{4\theta_k \hat{L}_{k-1}^2 a_k}, \\ \delta_3 &= \frac{c_{k-2}}{4\theta_k \hat{L}_{k-1}^2 a_{k-1}\omega_{k-1}}, & \delta_4 &= \frac{c_{k-3}a_{k-1}}{4\theta_k \hat{L}_{k-2}^2 a_k a_{k-2}\omega_{k-2}}. \end{aligned} \tag{24}$$

Thus, interpreting any terms containing c_i with $i < 0$ as zero, and plugging the choices of (24) into (23), the goal now becomes to ensure that the coefficient multiplying $\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_{\mathbf{\Lambda}}^2$ is at most $-c_k/2$. Recalling that $\theta_k \omega_{k-1} = \theta_{k-1}$, we want to ensure:

$$\begin{aligned} &\frac{(2\theta_k a_k L_k)^2}{c_{k-1}} + \frac{(2\theta_k a_k \hat{L}_{k-1})^2}{c_{k-2}} + \frac{(2\theta_{k-1} a_{k-1} \hat{L}_{k-1})^2}{c_{k-2}} + \frac{1}{c_{k-3}}\Big(\frac{2\hat{L}_{k-2}\theta_k a_k \theta_{k-2} a_{k-2}}{\theta_{k-1} a_{k-1}}\Big)^2 \\ &- \frac{\theta_k \gamma \omega_{k-1}(1+\beta)}{(1+\omega_{k-1})(1+\beta^2)} - \frac{\theta_k a_k (1+\beta\mu a_{k-1})}{\beta a_{k-1}} \leq -c_k. \end{aligned}$$

To ensure this inequality is satisfied, we define $c_k := \frac{\theta_k a_k(\beta\mu a_{k-1}+1)}{\beta a_{k-1}}$, and use the remaining negative term $-\frac{\theta_k \gamma \omega_{k-1}(1+\beta)}{(1+\omega_{k-1})(1+\beta^2)}$ to cancel out all of the non-negative terms in the above inequality. Fix $\tau \in (0,1)$. Defining $\eta := \sqrt{\frac{\gamma(1+\beta)}{1+\beta^2}}$ to simplify the notation, we reach the following conditions on the step size:

Setting $\frac{(2\theta_k a_k L_k)^2}{c_{k-1}} \leq \frac{\tau}{3}\eta^2 \theta_k \frac{\omega_{k-1}}{1+\omega_{k-1}}$ for $k \geq 1$, plugging in the definition of c_{k-1}, and solving for a_k (where we recall $\theta_k \omega_{k-1} = \theta_{k-1}$), we get that the first condition is

$$a_k \leq \frac{\eta\sqrt{\tau}}{2\sqrt{3}L_k} \frac{\omega_{k-1}}{\sqrt{1+\omega_{k-1}}} \sqrt{\frac{a_{k-1}(\beta\mu a_{k-2}+1)}{\beta a_{k-2}}}, \ k \geq 1. \tag{25}$$

Setting $\frac{(2\theta_k a_k \hat{L}_{k-1})^2}{c_{k-2}} \leq \frac{\tau}{3}\eta^2 \theta_k \frac{\omega_{k-1}}{1+\omega_{k-1}}$ for $k \geq 2$, and solving for a_k, the second condition is

$$a_k \leq \frac{\eta\sqrt{\tau}}{2\sqrt{3}\hat{L}_{k-1}} \frac{\omega_{k-1}\sqrt{\omega_{k-2}}}{\sqrt{1+\omega_{k-1}}} \sqrt{\frac{a_{k-2}(\beta\mu a_{k-3}+1)}{\beta a_{k-3}}}, \ k \geq 2. \tag{26}$$

Setting $\frac{(2\theta_{k-1} a_{k-1} \hat{L}_{k-1})^2}{c_{k-2}} \leq (1-\tau)\eta^2 \theta_k \frac{\omega_{k-1}}{1+\omega_{k-1}}$ for $k \geq 2$, and solving for a_{k-1}, the third condition is

$$a_{k-1} \leq \frac{\eta\sqrt{1-\tau}}{2\hat{L}_{k-1}} \frac{\sqrt{\omega_{k-2}}}{\sqrt{1+\omega_{k-1}}} \sqrt{\frac{a_{k-2}(\beta\mu a_{k-3}+1)}{\beta a_{k-3}}}, \ k \geq 2. \tag{27}$$

Finally, setting $\frac{1}{c_{k-3}}\Big(\frac{2\hat{L}_{k-2}\theta_k a_k \theta_{k-2} a_{k-2}}{\theta_{k-1} a_{k-1}}\Big)^2 \leq \frac{\tau}{3}\eta^2 \theta_k \frac{\omega_{k-1}}{1+\omega_{k-1}}$ for $k \geq 3$, and solving for a_k, the fourth condition is

$$a_k \leq \frac{\eta\sqrt{\tau}}{2\sqrt{3}\hat{L}_{k-2}} \frac{\omega_{k-1}}{\sqrt{1+\omega_{k-1}}} \sqrt{\frac{\omega_{k-3}}{\omega_{k-2}}} \frac{a_{k-1}}{a_{k-2}} \sqrt{\frac{a_{k-3}(\beta\mu a_{k-4}+1)}{\beta a_{k-4}}}, \ k \geq 3. \tag{28}$$

To complete the proof, it remains to verify that the conditions (25)–(28) hold by our choice of the step size, which is immediate upon noting that $\omega_k \in (0,1], \forall k$. □

Proof of Theorem 1. Recall that $h_k(\mathbf{u}) = h_k^{(1)}(\mathbf{u}) + h_k^{(2)}(\mathbf{u})$. Using Lemmas 2 to 4, summing $\theta_k h_k(\mathbf{u})$ and observing that the bound telescopes, we obtain

$$\begin{aligned} \sum_{k=1}^{K} \theta_k h_k(\mathbf{u}) \leq{}& \theta_K a_K \big\langle \boldsymbol{F}(\mathbf{u}_K) - \tilde{\boldsymbol{F}}_K, \mathbf{u}_{K+1} - \mathbf{u} \big\rangle \\ &- \frac{3c_{K-1}}{8}\|\mathbf{u}_K - \mathbf{u}_{K-1}\|_{\mathbf{\Lambda}}^2 + \frac{c_0}{2}\|\mathbf{u}_1 - \mathbf{u}_0\|_{\mathbf{\Lambda}}^2 \\ &+ \frac{1}{2(1-\beta)}\Big(\theta_1(1+\beta\mu a_1)\|\mathbf{u} - \mathbf{v}_1\|_{\mathbf{\Lambda}}^2 - \theta_K(1+\mu a_K)\|\mathbf{u} - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}}^2\Big) \\ &+ \frac{\eta^2\beta^2}{2}\big(\theta_0\|\mathbf{u}_1 - \mathbf{v}_0\|_{\mathbf{\Lambda}}^2 - \theta_K\|\mathbf{u}_{K+1} - \mathbf{v}_K\|_{\mathbf{\Lambda}}^2\big), \end{aligned} \tag{29}$$

where $\eta^2 = \frac{\gamma(1+\beta)}{1+\beta^2}$, $\tilde{\boldsymbol{F}}_0 = \boldsymbol{F}(\mathbf{u}_0)$, and we drop the negative terms of $\|\mathbf{u}_{K+1} - \mathbf{u}_K\|_{\boldsymbol{\Lambda}}^2$ and $\|\mathbf{u}_{K-1} - \mathbf{u}_{K-2}\|_{\boldsymbol{\Lambda}}^2$ (if $K \geq 3$).

Using the generalized Cauchy–Schwarz, Young's inequality, and the definition of $\hat{L}_K$, for any $\delta_1, \delta_2 > 0$,

$$\begin{aligned}
&\theta_K a_K \big\langle \boldsymbol{F}(\mathbf{u}_K) - \tilde{\boldsymbol{F}}_K, \mathbf{u}_{K+1} - \mathbf{u} \big\rangle \\
&= \theta_K a_K \big\langle \boldsymbol{F}(\mathbf{u}_K) - \tilde{\boldsymbol{F}}_K, \mathbf{u}_{K+1} - \mathbf{v}_{K+1} \big\rangle + \theta_K a_K \big\langle \boldsymbol{F}(\mathbf{u}_K) - \tilde{\boldsymbol{F}}_K, \mathbf{v}_{K+1} - \mathbf{u} \big\rangle \\
&\leq \frac{(\theta_K a_K \hat{L}_K)^2(\delta_1 + \delta_2)}{2} \|\mathbf{u}_K - \mathbf{u}_{K-1}\|_{\boldsymbol{\Lambda}}^2 + \frac{1}{2\delta_1} \|\mathbf{u} - \mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2 + \frac{1}{2\delta_2} \|\mathbf{v}_{K+1} - \mathbf{u}_{K+1}\|_{\boldsymbol{\Lambda}}^2 \\
&= \frac{(\theta_K a_K \hat{L}_K)^2(\delta_1 + \delta_2)}{2} \|\mathbf{u}_K - \mathbf{u}_{K-1}\|_{\boldsymbol{\Lambda}}^2 + \frac{1}{2\delta_1} \|\mathbf{u} - \mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2 + \frac{\beta^2}{2\delta_2} \|\mathbf{v}_K - \mathbf{u}_{K+1}\|_{\boldsymbol{\Lambda}}^2,
\end{aligned} \tag{30}$$

where the last equality uses $\mathbf{v}_{K+1} = (1-\beta)\mathbf{u}_{K+1} + \beta \mathbf{v}_K$.

Set

$$\delta_1 = \frac{2(1-\beta)}{\theta_K(1+\mu a_K)}, \qquad \delta_2 = \frac{1}{\eta^2 \theta_K}. \tag{31}$$

Under Equation (11), together with Equations (27) and (28), these values ensure

$$\frac{(\theta_K a_K \hat{L}_K)^2(\delta_1 + \delta_2)}{2} \leq \frac{3c_{K-1}}{8}.$$

Hence Equation (29) simplifies to

$$\begin{aligned}
\sum_{k=1}^{K} \theta_k h_k(\mathbf{u}) \leq{}& \frac{1+\mu a_0}{2(1-\beta)} \|\mathbf{u} - \mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2 + \Big(\frac{\eta^2\beta^2}{2} + \frac{c_0}{2}\Big) \|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2 \\
&- \frac{\theta_K(1+\mu a_K)}{4(1-\beta)} \|\mathbf{u} - \mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2.
\end{aligned} \tag{32}$$

By Lemma 1 and (32), recalling $c_0 = \frac{1+\beta\mu a_0}{\beta}$, we obtain

$$\begin{aligned}
&A_K \operatorname{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) + \frac{\theta_K(1+\mu a_K)}{4(1-\beta)} \|\mathbf{u} - \mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2 \\
&\leq \frac{1+\mu a_0}{2(1-\beta)} \|\mathbf{u} - \mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2 + \Big(\frac{\eta^2\beta^2}{2} + \frac{1+\beta\mu a_0}{2\beta}\Big) \|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2 \\
&= \mathcal{B}_0(\mathbf{u}).
\end{aligned} \tag{33}$$

This proves (8). Dropping the nonnegative second term on the left gives

$$\operatorname{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \frac{\mathcal{B}_0(\mathbf{u})}{A_K}.$$

Finally, let $\mathbf{u}^*$ be any solution of (P), and we have

$$\operatorname{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}^*) \geq 0.$$

Substituting $\mathbf{u} = \mathbf{u}^*$ in (33) and dropping the nonnegative gap term gives

$$\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\boldsymbol{\Lambda}}^2 \leq \frac{4(1-\beta)\mathcal{B}_0(\mathbf{u}^*)}{\theta_K(1+\mu a_K)} \leq \frac{4(1-\beta)\mathcal{B}_0(\mathbf{u}^*)}{\theta_K}.$$

□

Proof of Corollary 1. We first introduce the constants used in this proof. Recall

$$\mathcal{B}_0(\mathbf{u}^*) = \frac{1+\mu a_0}{2(1-\beta)} \|\mathbf{u}^* - \mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2 + \left(\frac{\eta^2\beta^2}{2} + \frac{c_0}{2}\right) \|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2,$$

where $c_0 = \frac{1+\beta\mu a_0}{\beta}$. $G_0 := \min_{\xi_0 \in \partial g(\mathbf{u}_0)} \|\boldsymbol{F}(\mathbf{u}_0) + \xi_0\|_{\boldsymbol{\Lambda}^{-1}}$. Let

$$D_1 := \|\mathbf{u}_0 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} + a_{\max} G_0, \qquad c_{\max} := \frac{1+\beta\mu a_{\max}}{\beta}.$$

Set

$$\begin{aligned}
R_* &:= \Big(2(1+\mu a_{\max})D_1^2 + 2(1-\beta)\big(\eta^2\beta^2 + c_{\max}\big)a_{\max}^2 G_0^2\Big)^{1/2}, \\
R_1 &:= \Big(\frac{\eta^2(1-\tau)(1-\beta)}{4\beta}(1+\beta\mu a_{\max})(1+\mu a_{\max})\Big(2(1-\beta) + \frac{1}{\eta^2}\Big)\Big)^{1/2} a_{\max} G_0, \\
R_{\mathbf{v}} &:= \max\Big\{D_1, \sqrt{R_*^2 + R_1^2}\Big\}, \\
R_{\mathbf{u}} &:= \frac{1+\beta}{1-\beta} R_{\mathbf{v}}.
\end{aligned}$$

We prove that

$$\|\mathbf{v}_k - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} \le R_{\mathbf{v}}, \qquad \|\mathbf{u}_k - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} \le R_{\mathbf{u}}, \qquad \forall k \ge 0.$$

(a) Bounds for $\mathbf{u}_1$, $\mathbf{u}_0$ and $\mathbf{v}_1$, $\mathbf{v}_0$. Fix any $\xi_0 \in \partial g(\mathbf{u}_0)$. By the optimality condition for the first proximal step, there exists $\xi_1 \in \partial g(\mathbf{u}_1)$ such that

$$\boldsymbol{\Lambda}(\mathbf{u}_1 - \mathbf{u}_0) + a_0\big(\boldsymbol{F}(\mathbf{u}_0) + \xi_1\big) = 0.$$

Taking the inner product with $\mathbf{u}_1 - \mathbf{u}_0$ and using monotonicity of ∂g gives

$$\begin{aligned}
\|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2 &= -a_0\big\langle \boldsymbol{F}(\mathbf{u}_0) + \xi_1, \mathbf{u}_1 - \mathbf{u}_0\big\rangle \\
&= -a_0\big\langle \boldsymbol{F}(\mathbf{u}_0) + \xi_0, \mathbf{u}_1 - \mathbf{u}_0\big\rangle - a_0\big\langle \xi_1 - \xi_0, \mathbf{u}_1 - \mathbf{u}_0\big\rangle \\
&\le a_0 \|\boldsymbol{F}(\mathbf{u}_0) + \xi_0\|_{\boldsymbol{\Lambda}^{-1}} \|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}}.
\end{aligned}$$

Taking the minimum over $\xi_0 \in \partial g(\mathbf{u}_0)$ yields

$$\|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}} \le a_0 G_0 \le a_{\max} G_0. \tag{34}$$

Therefore,

$$\begin{aligned}
\|\mathbf{u}_1 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} &\le \|\mathbf{u}_0 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} + a_{\max} G_0 \\
&= D_1 \\
&\le R_{\mathbf{v}} \le R_{\mathbf{u}}.
\end{aligned}$$

Moreover, since $\mathbf{v}_1 = (1-\beta)\mathbf{u}_1 + \beta\mathbf{u}_0$,

$$\begin{aligned}
\|\mathbf{v}_1 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} &\le (1-\beta)\|\mathbf{u}_1 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} + \beta\|\mathbf{u}_0 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} \\
&\le (1-\beta)D_1 + \beta\|\mathbf{u}_0 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} \\
&\le D_1 \le R_{\mathbf{v}}.
\end{aligned}$$

The bounds for $\mathbf{u}_0 = \mathbf{v}_0$ are immediate from $\|\mathbf{u}_0 - \mathbf{u}^*\|_{\boldsymbol{\Lambda}} \le D_1 \le R_{\mathbf{v}} \le R_{\mathbf{u}}$.

(b) Bounds for $\mathbf{u}_2$ and $\mathbf{v}_2$. Although Theorem 1 is stated for $K \ge 2$, the same telescoping estimate (32) in its proof can be applied at $K = 1$. By (32), setting $\mathbf{u} = \mathbf{u}^*$ gives

$$h_1(\mathbf{u}^*) \ge 0.$$

The $K = 1$ version of (32) leads to

$$\begin{aligned}
0 \le \theta_1 h_1(\mathbf{u}^*) \le \theta_1 a_1\big\langle \boldsymbol{F}(\mathbf{u}_1) - \tilde{\boldsymbol{F}}_1, \mathbf{u}_2 - \mathbf{u}^*\big\rangle + \frac{c_0}{2}\|\mathbf{u}_1 - \mathbf{u}_0\|_{\boldsymbol{\Lambda}}^2 \\
+ \frac{1}{2(1-\beta)}\Big(\theta_1(1+\beta\mu a_1)\|\mathbf{u}^* - \mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2 - \theta_1(1+\mu a_1)\|\mathbf{u}^* - \mathbf{v}_2\|_{\boldsymbol{\Lambda}}^2\Big) \\
+ \frac{\eta^2\beta^2}{2}\big(\theta_0\|\mathbf{u}_1 - \mathbf{v}_0\|_{\boldsymbol{\Lambda}}^2 - \theta_1\|\mathbf{u}_2 - \mathbf{v}_1\|_{\boldsymbol{\Lambda}}^2\big).
\end{aligned}$$

By the same bounding techniques from (30) and parameters setting from (31), we obtain

$$\begin{aligned}\frac{\theta_1(1+\mu a_1)}{4(1-\beta)}\|\mathbf{u}^*-\mathbf{v}_2\|_{\mathbf{\Lambda}}^2 &\le \mathcal{B}_0(\mathbf{u}^*)+\frac{(\theta_1 a_1\hat{L}_1)^2}{2}\Big(\frac{2(1-\beta)}{\theta_1(1+\mu a_1)}+\frac{1}{\eta^2\theta_1}\Big)\|\mathbf{u}_1-\mathbf{u}_0\|_{\mathbf{\Lambda}}^2\\ &\le \mathcal{B}_0(\mathbf{u}^*)+\frac{(\theta_1 a_1\hat{L}_1)^2}{2}\Big(\frac{2(1-\beta)}{\theta_1(1+\mu a_1)}+\frac{1}{\eta^2\theta_1}\Big)a_{\max}^2G_0^2,\\ &\le \mathcal{B}_0(\mathbf{u}^*)+\frac{(\theta_1 a_1\hat{L}_1)^2}{2}\Big(2(1-\beta)+\frac{1}{\eta^2}\Big)a_{\max}^2G_0^2,\end{aligned}$$

where the second last inequality is by (34), and the last inequality is by $\theta_1 \ge 1$ and $\mu \ge 0$. Consequently,

$$\begin{aligned}\|\mathbf{u}^*-\mathbf{v}_2\|_{\mathbf{\Lambda}}^2 &\le \frac{4(1-\beta)\mathcal{B}_0(\mathbf{u}^*)}{\theta_1(1+\mu a_1)}+\frac{2(1-\beta)(\theta_1 a_1\hat{L}_1)^2}{\theta_1(1+\mu a_1)}\Big(2(1-\beta)+\frac{1}{\eta^2}\Big)a_{\max}^2G_0^2\\ &\le 4(1-\beta)\mathcal{B}_0(\mathbf{u}^*)+2(1-\beta)(\theta_1 a_1\hat{L}_1)^2\Big(2(1-\beta)+\frac{1}{\eta^2}\Big)a_{\max}^2G_0^2.\end{aligned}\tag{35}$$

We firstly bound the first term on the right-hand side of (35). By the definition of $\mathcal{B}_0$, using $a_0 \le a_{\max}$, $c_0 \le c_{\max}$, and the bounds (34) from part (a), we obtain

$$\begin{aligned}4(1-\beta)\mathcal{B}_0(\mathbf{u}^*) &= 2(1+\mu a_0)\|\mathbf{u}^*-\mathbf{v}_1\|_{\mathbf{\Lambda}}^2+2(1-\beta)\big(\eta^2\beta^2+c_0\big)\|\mathbf{u}_1-\mathbf{u}_0\|_{\mathbf{\Lambda}}^2\\ &\le 2(1+\mu a_{\max})D_1^2+2(1-\beta)\big(\eta^2\beta^2+c_{\max}\big)a_{\max}^2G_0^2\\ &= R_*^2.\end{aligned}\tag{36}$$

Now we bound the second term on the right-hand side of (35). By step-size update rules (11), (12), or (13), we have

$$\begin{aligned}a_1\hat{L}_1 &\le \frac{\eta}{2}\sqrt{\frac{(1-\tau)\omega_0}{2}}\sqrt{\frac{\beta\mu a_0+1}{\beta}}\\ &= \frac{\eta}{2}\sqrt{\frac{(1-\tau)}{2}\frac{1+\rho\beta\mu a_0}{1+\mu a_0}}\sqrt{\frac{\beta\mu a_0+1}{\beta}},\end{aligned}\tag{37}$$

where the last equality is by $\omega_0 = \frac{1+\rho\beta\mu a_0}{1+\mu a_0}$. Since $\theta_1 = \frac{\theta_0}{\omega_0} = \frac{1+\mu a_0}{1+\rho\beta\mu a_0}$, by (37) we have

$$\begin{aligned}&2(1-\beta)(\theta_1 a_1\hat{L}_1)^2\Big(2(1-\beta)+\frac{1}{\eta^2}\Big)a_{\max}^2G_0^2\\ &\le \frac{\eta^2(1-\tau)(1-\beta)}{4\beta}\frac{(1+\beta\mu a_0)(1+\mu a_0)}{1+\rho\beta\mu a_0}\Big(2(1-\beta)+\frac{1}{\eta^2}\Big)a_{\max}^2G_0^2\\ &\le R_1^2\end{aligned}\tag{38}$$

Combine (36) and (38) for (35) yields

$$\begin{aligned}\|\mathbf{u}^*-\mathbf{v}_2\|_{\mathbf{\Lambda}} &\le \sqrt{R_*^2+R_1^2}\\ &\le R_{\mathbf{v}}.\end{aligned}$$

Using $\mathbf{u}_2 = \frac{1}{1-\beta}\mathbf{v}_2 - \frac{\beta}{1-\beta}\mathbf{v}_1$, and the bound on $\mathbf{v}_1$ from part (a),

$$\begin{aligned}\|\mathbf{u}_2-\mathbf{u}^*\|_{\mathbf{\Lambda}} &\le \frac{1}{1-\beta}\|\mathbf{v}_2-\mathbf{u}^*\|_{\mathbf{\Lambda}}+\frac{\beta}{1-\beta}\|\mathbf{v}_1-\mathbf{u}^*\|_{\mathbf{\Lambda}}\\ &\le \frac{1+\beta}{1-\beta}R_{\mathbf{v}}\\ &= R_{\mathbf{u}}.\end{aligned}$$

(c) Bounds for $\mathbf{u}_k$ and $\mathbf{v}_k$ for $k \geq 3$. Applying Theorem 1 gives, for every $K \geq 2$,

$$\begin{aligned}\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}}^2 &\leq \frac{4(1-\beta)\mathcal{B}_0(\mathbf{u}^*)}{\theta_K(1+\mu a_K)} \\ &\leq 4(1-\beta)\mathcal{B}_0(\mathbf{u}^*) \\ &\leq R_*^2,\end{aligned}$$

where in the last inequality we use (36). Thus,

$$\|\mathbf{v}_k - \mathbf{u}^*\|_{\mathbf{\Lambda}} \leq R_* \leq R_{\mathbf{v}}, \qquad \forall k \geq 3.$$

Using $\mathbf{u}_k = \frac{1}{1-\beta}\mathbf{v}_k - \frac{\beta}{1-\beta}\mathbf{v}_{k-1}$, we obtain, for every $k \geq 3$,

$$\begin{aligned}\|\mathbf{u}_k - \mathbf{u}^*\|_{\mathbf{\Lambda}} &\leq \frac{1}{1-\beta}\|\mathbf{v}_k - \mathbf{u}^*\|_{\mathbf{\Lambda}} + \frac{\beta}{1-\beta}\|\mathbf{v}_{k-1} - \mathbf{u}^*\|_{\mathbf{\Lambda}} \\ &\leq \frac{1+\beta}{1-\beta}R_{\mathbf{v}} = R_{\mathbf{u}}.\end{aligned}$$

Combining parts (a)–(c), we have

$$\|\mathbf{v}_k - \mathbf{u}^*\|_{\mathbf{\Lambda}} \leq R_{\mathbf{v}}, \qquad \|\mathbf{u}_k - \mathbf{u}^*\|_{\mathbf{\Lambda}} \leq R_{\mathbf{u}}, \qquad \forall k \geq 0.$$

(d) Bounds for partially updated cyclic points. For every $k \geq 1$ and $i \in [m]$,

$$\begin{aligned}\|\mathbf{u}_{k-1,i} - \mathbf{u}^*\|_{\mathbf{\Lambda}}^2 &= \sum_{j<i} \|\mathbf{u}_k^j - \mathbf{u}^{*,j}\|_{\mathbf{\Lambda}_j}^2 + \sum_{j\geq i} \|\mathbf{u}_{k-1}^j - \mathbf{u}^{*,j}\|_{\mathbf{\Lambda}_j}^2 \\ &\leq \|\mathbf{u}_k - \mathbf{u}^*\|_{\mathbf{\Lambda}}^2 + \|\mathbf{u}_{k-1} - \mathbf{u}^*\|_{\mathbf{\Lambda}}^2 \\ &\leq 2R_{\mathbf{u}}^2.\end{aligned}$$

Hence all full iterates and all partially updated cyclic points lie in the compact ball

$$\mathcal{C} := \left\{\mathbf{u} \in \mathbb{R}^d : \|\mathbf{u} - \mathbf{u}^*\|_{\mathbf{\Lambda}} \leq \sqrt{2}R_{\mathbf{u}}\right\}.$$

The radius $R_{\mathbf{u}}$ is finite and depends only on $\mathbf{u}_0$, $a_{\max}$, G_0, $\mathbf{u}^*$, and the fixed algorithmic and metric parameters. This proves the claim. □

E Initialization and omitted proofs from Section 5

Proof of Lemma 5. The trial sequence in the repeat loop of Algorithm 2 is defined as

$$a^{(i)} := a_{\text{init}} 2^{-i}, \qquad i = 0, 1, 2, \ldots.$$

Definition of $a_{\text{init}} = \min\left\{\frac{C}{L_{\text{init}}}, \frac{\hat{C}}{\hat{L}_{\text{init}}}\right\}$, $L_{\mathcal{C}}$, and $\hat{L}_{\mathcal{C}}$ implies $a_{\text{init}} \geq s_{\mathcal{C}}$, so the first trial is at least the inverse Lipschitz scale.

(a) Termination For every trial step size a, the definition of $L_{\mathcal{C}}$ gives

$$L_1(a) \leq L_{\mathcal{C}}.$$

So every trial step size $a \leq s_{\mathcal{C}}$ satisfies the stopping condition.

Now define

$$N := \left\lceil \log_2\left(\frac{a_{\text{init}}}{s_{\mathcal{C}}}\right)\right\rceil.$$

Since $a_{\text{init}} \geq s_{\mathcal{C}}$, we have $N \geq 0$, and

$$a^{(N)} = a_{\text{init}} 2^{-N} \leq s_{\mathcal{C}}.$$

By the preceding paragraph, the trial $a^{(N)}$ is accepted. Hence Algorithm 2 terminates after at most N backtracking reductions.

(b) Upper bound Let a_0 be the accepted trial step size. By the stopping condition, we immediately have

$$a_0 \le s_1(a_0).$$

(c) Lower bound If the first trial is accepted, then

$$a_0 = a_{\text{init}} \ge s_{\mathcal{C}} \ge \frac{s_{\mathcal{C}}}{2}.$$

Otherwise, let a_{prev} be the trial step immediately before the accepted one. Then a_{prev} was rejected and

$$a_0 = \frac{a_{\text{prev}}}{2}.$$

Since every trial step of at most $s_{\mathcal{C}}$ is accepted, the rejected step must satisfy

$$a_{\text{prev}} > s_{\mathcal{C}}.$$

Therefore,

$$a_0 = \frac{a_{\text{prev}}}{2} > \frac{s_{\mathcal{C}}}{2}.$$

Combining the two cases gives

$$a_0 \ge \frac{s_{\mathcal{C}}}{2}.$$

This proves the lemma. □

Claim 1. *For any $k \ge 2$, if $a_k = s_k\sqrt{a_{k-1}/a_{k-2}}$, then*

$$a_k + a_{k-2} \ge \frac{2s_k}{\sqrt{\rho_0}}.$$

Proof of Claim 1. From $a_k = s_k\sqrt{a_{k-1}/a_{k-2}}$ and $a_k \le \rho_0 a_{k-1}$,

$$\sqrt{a_k a_{k-2}} = s_k\sqrt{\frac{a_{k-1}}{a_k}} \ge \frac{s_k}{\sqrt{\rho_0}}.$$

Applying the arithmetic–geometric mean inequality to a_k and a_{k-2}, we obtain

$$a_k + a_{k-2} \ge 2\sqrt{a_k a_{k-2}}.$$

The desired result follows. □

Proof of Theorem 2. Recall the simplified step-size rule:

$$a_k = \min\Big\{\rho_0 a_{k-1}, s_k\sqrt{\frac{a_{k-1}}{a_{k-2}}}\Big\}, \qquad s_k := \min\Big\{\frac{C}{L_k}, \frac{\hat{C}}{\hat{L}_k}\Big\},$$

where $\rho_0 := \min\{\rho, \beta(1+\beta)(1-\gamma)\}$, and $1 < \rho_0 < 2$. We call $a_k = \rho_0 a_{k-1}$ as the first branch, and $a_k = s_k\sqrt{\frac{a_{k-1}}{a_{k-2}}}$ as the second branch of the step-size rule for any $k \ge 1$.

By Lemma 5, the initialization returns

$$a_{-1} = a_0 \ge \frac{s_{\mathcal{C}}}{2}.$$

Moreover, by the definitions of $L_{\mathcal{C}}, \widehat{L}_{\mathcal{C}}$, and $s_{\mathcal{C}}$, the local estimates along the trajectory satisfy

$$L_k \le L_{\mathcal{C}}, \qquad \hat{L}_k \le \widehat{L}_{\mathcal{C}},$$

and hence

$$s_k = \min\Big\{\frac{C}{L_k}, \frac{\hat{C}}{\hat{L}_k}\Big\} \ge \min\Big\{\frac{C}{L_{\mathcal{C}}}, \frac{\hat{C}}{\widehat{L}_{\mathcal{C}}}\Big\} = s_{\mathcal{C}}, \qquad k \ge 1.$$

Consequently,

$$a_1 = \min\{\rho_0 a_0, s_1\} \ge \min\Big\{\frac{\rho_0 s_{\mathcal{C}}}{2}, s_{\mathcal{C}}\Big\} = \frac{\rho_0 s_{\mathcal{C}}}{2} > \frac{s_{\mathcal{C}}}{2}.$$

By Theorem 1, for every $\mathbf{u} \in \mathcal{U}$ and every $K \geq 2$,

$$\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \frac{\mathcal{B}_0(\mathbf{u})}{A_K} \leq \frac{\mathcal{B}_{\mathcal{U}}}{A_K}, \qquad A_K := \sum_{k=1}^{K} \theta_k a_k.$$

Also, applying Theorem 1 with $\mathbf{u} = \mathbf{u}^*$ and using the definition of R_* in Corollary 1, we have

$$\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}}^2 \leq \frac{R_*^2}{\theta_K}.$$

Thus it remains to lower bound A_K and, in the strongly convex case, θ_K.

Since $\omega_k \leq 1$, we have $\theta_k \geq 1$ for all k. Therefore,

$$A_K = \sum_{k=1}^{K} \theta_k a_k \geq \sum_{k=1}^{K} a_k.$$

We first prove a deterministic lower bound on the accumulated step sizes. Call an index j *bad* if

$$a_j < \frac{s_{\mathcal{C}}}{2},$$

and *good* otherwise. Since $a_0 \geq s_{\mathcal{C}}/2$ and $a_1 > s_{\mathcal{C}}/2$, any bad interval starts at index at least 2.

Let

$$J = \{p, p+1, \ldots, p+t-1\}, \quad t \geq 1$$

be a maximal bad interval. Then $p \geq 2$ and $a_{p-1} \geq s_{\mathcal{C}}/2$. Since $\rho_0 > 1$, the first bad step a_p cannot be generated by $\rho_0 a_{p-1}$. Hence

$$a_p = s_p \sqrt{\frac{a_{p-1}}{a_{p-2}}}.$$

We next show that, inside the same bad interval, only a_p and possibly a_{p+1} can be generated by the second branch of the step-size rule. Suppose, for contradiction, that $j \in J$, $j \geq p+2$, and

$$a_j = s_j \sqrt{\frac{a_{j-1}}{a_{j-2}}}.$$

By Claim 1 and $s_j \geq s_{\mathcal{C}}$,

$$a_j + a_{j-2} \geq \frac{2s_j}{\sqrt{\rho_0}} \geq \frac{2s_{\mathcal{C}}}{\sqrt{\rho_0}} > s_{\mathcal{C}},$$

where the last inequality uses $\rho_0 < 2$. This contradicts the fact that both j and $j-2$ are bad, which would imply

$$a_j + a_{j-2} < \frac{s_{\mathcal{C}}}{2} + \frac{s_{\mathcal{C}}}{2} = s_{\mathcal{C}}.$$

Therefore, all bad indices from $p+2$ onward are generated by the first branch.

We now show that every bad interval is compensated by a large earlier step. First suppose either $|J| = 1$, or $|J| \geq 2$ and a_{p+1} is generated by the first branch. Then

$$a_{p+i} = \rho_0^i a_p, \qquad i = 0, \ldots, |J| - 1.$$

Since $a_{p+t-1} < s_{\mathcal{C}}/2$, we have

$$a_p < \frac{s_{\mathcal{C}}}{2} \rho_0^{-(|J|-1)}. \tag{39}$$

On the other hand, using $s_p \geq s_{\mathcal{C}}$ and $a_{p-1} \geq s_{\mathcal{C}}/2$,

$$\begin{aligned} a_p = s_p \sqrt{\frac{a_{p-1}}{a_{p-2}}} &\geq s_{\mathcal{C}} \sqrt{\frac{s_{\mathcal{C}}}{2a_{p-2}}} \\ \Rightarrow a_{p-2} &\geq \frac{s_{\mathcal{C}}^3}{2a_p^2}. \end{aligned} \tag{40}$$

Combining (40) and (39) gives

$$a_{p-2} \geq 2s_{\mathcal{C}}\rho_0^{2|J|-2} \geq \frac{s_{\mathcal{C}}}{2}\rho_0^{2|J|},$$

where the last inequality follows from $\rho_0 < 2$.

It remains to consider the case where $|J| \geq 2$ and

$$a_{p+1} = s_{p+1}\sqrt{\frac{a_p}{a_{p-1}}}.$$

Then $a_{p+i} = \rho_0^{i-1}a_{p+1}$ for $i = 1, \ldots, |J| - 1$, so

$$a_{p+1} < \frac{s_{\mathcal{C}}}{2}\rho_0^{-(|J|-2)}. \tag{41}$$

Since $a_p = s_p\sqrt{\frac{a_{p-1}}{a_{p-2}}}$, we get

$$\begin{aligned} a_p a_{p+1} &= s_p s_{p+1}\sqrt{\frac{a_p}{a_{p-2}}} \\ \Rightarrow a_{p-2} &= \frac{s_p^2 s_{p+1}^2}{a_p a_{p+1}^2}. \end{aligned} \tag{42}$$

By $s_p, s_{p+1} \geq s_{\mathcal{C}}$, $a_p < s_{\mathcal{C}}/2$, and (41) it follows that

$$a_{p-2} \geq \frac{2s_{\mathcal{C}}^3}{a_{p+1}^2} \geq 8s_{\mathcal{C}}\rho_0^{2|J|-4} \geq \frac{s_{\mathcal{C}}}{2}\rho_0^{2|J|},$$

again using $\rho_0 < 2$. Hence for every bad interval, it satisfies

$$a_{p-2} \geq \frac{s_{\mathcal{C}}}{2}\rho_0^{2|J|}. \tag{43}$$

The case $p = 2$ cannot occur under our initialization. Indeed, Lemma 5 gives $a_0 \leq s_1(a_0)$. Together with $a_{-1} = a_0$ and $\rho_0 > 1$, this implies

$$a_1 = \min\{\rho_0 a_0, s_1\} \geq a_0.$$

Using $a_0 \geq s_{\mathcal{C}}/2$, $s_2 \geq s_{\mathcal{C}}$, and $1 < \rho_0 < 2$, we therefore obtain

$$\begin{aligned} a_2 &= \min\left\{\rho_0 a_1,\ s_2\sqrt{\frac{a_1}{a_0}}\right\} \\ &\geq \min\{\rho_0 a_1, s_2\} \\ &\geq \min\left\{\frac{\rho_0 s_{\mathcal{C}}}{2},\ s_{\mathcal{C}}\right\} \\ &= \frac{\rho_0 s_{\mathcal{C}}}{2} > \frac{s_{\mathcal{C}}}{2}. \end{aligned}$$

Thus a_2 is good, and every bad interval starts at $p \geq 3$.

Consequently, every bad interval within $\{1, \ldots, K\}$ has its compensating index $p - 2$ in $\{1, \ldots, K\}$, so the compensating step in (43) is included in the sum $\sum_{k=1}^{K} a_k$. Moreover, distinct bad intervals have distinct compensating indices. Moreover,

$$\inf_{t\geq 1}\frac{\rho_0^{2t}}{t} > 0 \quad \Rightarrow \quad \frac{s_{\mathcal{C}}}{2}\rho_0^{2|J|} \geq \frac{s_{\mathcal{C}}}{2}|J|\inf_{t\geq 1}\frac{\rho_0^{2t}}{t} = \Theta(|J|)$$

Thus each bad interval of length $|J|$ can be charged to its compensating step a_{p-2}, whose size is at least a constant multiple of $s_{\mathcal{C}}|J|$. Good indices contribute at least $s_{\mathcal{C}}/2$ each. Therefore, there exists a constant $c_1 > 0$, depending only on ρ_0, such that, up to the fixed initialization transient,

$$A_K \geq \sum_{k=1}^{K} a_k \geq c_1 s_{\mathcal{C}} K. \tag{44}$$

Thus it is sufficient to take

$$K = \mathcal{O}\Big(\frac{\mathcal{B}_{\mathcal{U}}}{s_{\mathcal{C}}\epsilon}\Big) = \mathcal{O}\Big(\frac{\mathcal{L}_{\mathcal{C}}\mathcal{B}_{\mathcal{U}}}{\epsilon}\Big)$$

iterations to ensure

$$A_K \geq \frac{\mathcal{B}_{\mathcal{U}}}{\epsilon}.$$

Consequently,

$$\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \frac{\mathcal{B}_{\mathcal{U}}}{A_K} \leq \epsilon, \qquad \forall \mathbf{u} \in \mathcal{U}.$$

This proves the monotone part.

Now suppose $\mu > 0$. Recall that $\omega_0 = 1$ and, for $j \geq 1$,

$$\omega_j = \frac{1 + \rho\beta\mu a_j}{1 + \mu a_j} = 1 - \frac{(1 - \rho\beta)\mu a_j}{1 + \mu a_j}.$$

Therefore,

$$\frac{1}{\theta_K} = \prod_{j=0}^{K-1} \omega_j = \prod_{j=1}^{K-1} \omega_j.$$

Let I_{good} be the set of post-initialization good indices before K:

$$I_{\mathrm{good}} := \Big\{ j \leq K - 1 : j \text{ is post-initialization and } a_j \geq \frac{s_{\mathcal{C}}}{2} \Big\}.$$

For every $j \in I_{\mathrm{good}}$,

$$\omega_j \leq 1 - \frac{(1 - \rho\beta)\mu s_{\mathcal{C}}/2}{1 + \mu s_{\mathcal{C}}/2}.$$

Hence

$$\frac{1}{\theta_K} \leq \Big(1 - \frac{(1 - \rho\beta)\mu s_{\mathcal{C}}/2}{1 + \mu s_{\mathcal{C}}/2}\Big)^{|I_{\mathrm{good}}|}.$$

Thus, if

$$|I_{\mathrm{good}}| \geq \frac{2(1 + \mu s_{\mathcal{C}}/2)}{(1 - \rho\beta)\mu s_{\mathcal{C}}/2} \log\Big(\frac{R_*}{\epsilon}\Big),$$

then

$$\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}}^2 \leq \frac{R_*^2}{\theta_K} \leq \epsilon^2,$$

and therefore

$$\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}} \leq \epsilon.$$

It remains to handle the complementary case, where the number of good indices is smaller than the preceding threshold. In this case,

$$|I_{\mathrm{good}}| = \mathcal{O}\Big(\frac{1 + \mu s_{\mathcal{C}}}{\mu s_{\mathcal{C}}} \log\Big(\frac{R_*}{\epsilon}\Big)\Big).$$

The bad indices are split by the good indices into at most $|I_{\mathrm{good}}| + 1$ maximal bad intervals. Hence, by the pigeonhole principle, one such bad interval J satisfies

$$|J| = \Omega\Big(\frac{K}{|I_{\mathrm{good}}| + 1}\Big).$$

By (43),

$$A_K \geq \frac{s_{\mathcal{C}}}{2}\rho_0^{2|J|} \geq \frac{s_{\mathcal{C}}}{2}\rho_0^{\Omega\left(K/(|I_{\mathrm{good}}|+1)\right)}.$$

Therefore, it is sufficient to take

$$\begin{aligned} K &= \mathcal{O}\Big(\frac{1 + \mu s_{\mathcal{C}}}{\mu s_{\mathcal{C}}} \log\Big(\frac{R_*}{\epsilon}\Big) \log\Big(\frac{\mathcal{B}_{\mathcal{U}}}{s_{\mathcal{C}}\epsilon}\Big)\Big) \\ &= \mathcal{O}\Big(\Big(\frac{\mathcal{L}_{\mathcal{C}}}{\mu}\Big) \log\Big(\frac{R_*}{\epsilon}\Big) \log\Big(\frac{\mathcal{L}_{\mathcal{C}}\mathcal{B}_{\mathcal{U}}}{\epsilon}\Big)\Big) \end{aligned}$$

to guarantee

$$A_K \geq \frac{\mathcal{B}_{\mathcal{U}}}{\epsilon}.$$

For this choice of K, the gap estimate from Theorem 1 gives

$$\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \frac{\mathcal{B}_{\mathcal{U}}}{A_K} \leq \epsilon, \qquad \forall \mathbf{u} \in \mathcal{U}.$$

Combining the two strongly convex cases proves that, after the stated number of iterations, either

$$\|\mathbf{u}^* - \mathbf{v}_{K+1}\|_{\mathbf{\Lambda}} \leq \epsilon$$

or

$$\mathrm{Gap}(\hat{\mathbf{u}}_K; \mathbf{u}) \leq \epsilon, \qquad \forall \mathbf{u} \in \mathcal{U}.$$

Together with the monotone guarantee, this proves the theorem. □

F Experimental details, additional experiments, and asset information

F.1 Reproducibility details

The experiments compare `ADUCA` against `PCCM`, `CODER`, `CODER-LineSearch`, and `GRAAL`. Outside the ablation study, `ADUCA` uses $(\beta, \rho, \gamma) = (0.8, 1.2, 0.2)$, the fully specified step-size (4), and the initialization by Algorithm 2. The baselines are tuned within the parameter ranges suggested in their original references, and we report the best-performing configuration for each method. All runs are deterministic first-order optimization runs on fixed public datasets and a fixed objective, so the reported curves do not involve random seeds, random data splits, or stochastic sampling; this is why we do not report error bars. All experiments were run in Python on an internal Linux server with dual AMD EPYC 7763 CPUs, 1 TiB RAM, and multi-terabyte local storage.

For the SVM problem, we set $\lambda_1 = 10^{-4}$ and $\lambda_2 = 10^{-4}$. We evaluate all methods on the LIBSVM [8] datasets `a9a` ($d = 123, n = 32561$), `gisette` ($d = 5000, n = 6000$), and `SUSY` ($d = 18, n = 5000000$). The diagonal rescaling matrix used in our experiments is

$$\begin{aligned} \mathbf{\Lambda} &:= \begin{bmatrix} \mathbf{\Lambda}_{\mathbf{x}} & 0 \\ 0 & \mathbf{\Lambda}_{\mathbf{y}} \end{bmatrix} \in \mathbb{R}^{(d+n)\times(d+n)}, \\ \text{where} \quad & \mathbf{\Lambda}_{\mathbf{x}} = \mathrm{Diag}(\lambda_1^{\mathbf{x}}, \ldots, \lambda_d^{\mathbf{x}}),\ \mathbf{\Lambda}_{\mathbf{y}} = \mathrm{Diag}(\lambda_1^{\mathbf{y}}, \ldots, \lambda_n^{\mathbf{y}}). \end{aligned} \tag{45}$$

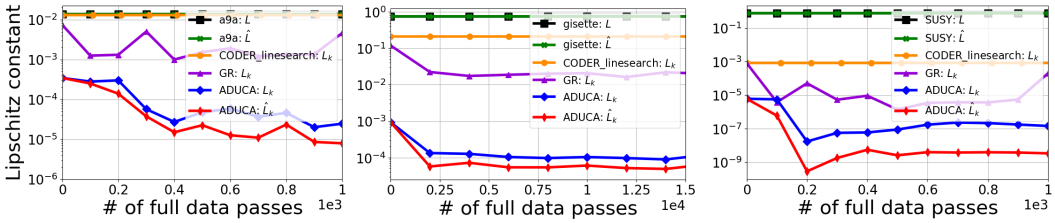
Here, the diagonal entries are set to reciprocal row and column norms of the signed data matrix $\bar{\boldsymbol{A}} = [b_1 \boldsymbol{A}_1, \ldots, b_n \boldsymbol{A}_n] \in \mathbb{R}^{d\times n}$:

$$\begin{aligned} \lambda_j^{\mathbf{x}} &:= \begin{cases} \|\bar{\boldsymbol{A}}_{j,:}\|_2^{-1}, & \text{if } \|\bar{\boldsymbol{A}}_{j,:}\|_2 > 0, \\ 1, & \text{if } \|\bar{\boldsymbol{A}}_{j,:}\|_2 = 0, \end{cases} \quad j = 1, \ldots, d, \\ \lambda_i^{\mathbf{y}} &:= \begin{cases} \|\bar{\boldsymbol{A}}_{:,i}\|_2^{-1}, & \text{if } \|\bar{\boldsymbol{A}}_{:,i}\|_2 > 0, \\ 1, & \text{if } \|\bar{\boldsymbol{A}}_{:,i}\|_2 = 0, \end{cases} \quad i = 1, \ldots, n. \end{aligned} \tag{46}$$

To quantify progress in SVM, we report the primal gap $f(\mathbf{x}) - f^\star$, where f is the primal objective obtained by maximizing over $\mathbf{y}$ in the SVM problem, and $f^\star$ is a reference value taken as the lowest f value attained after a long run, across all methods.

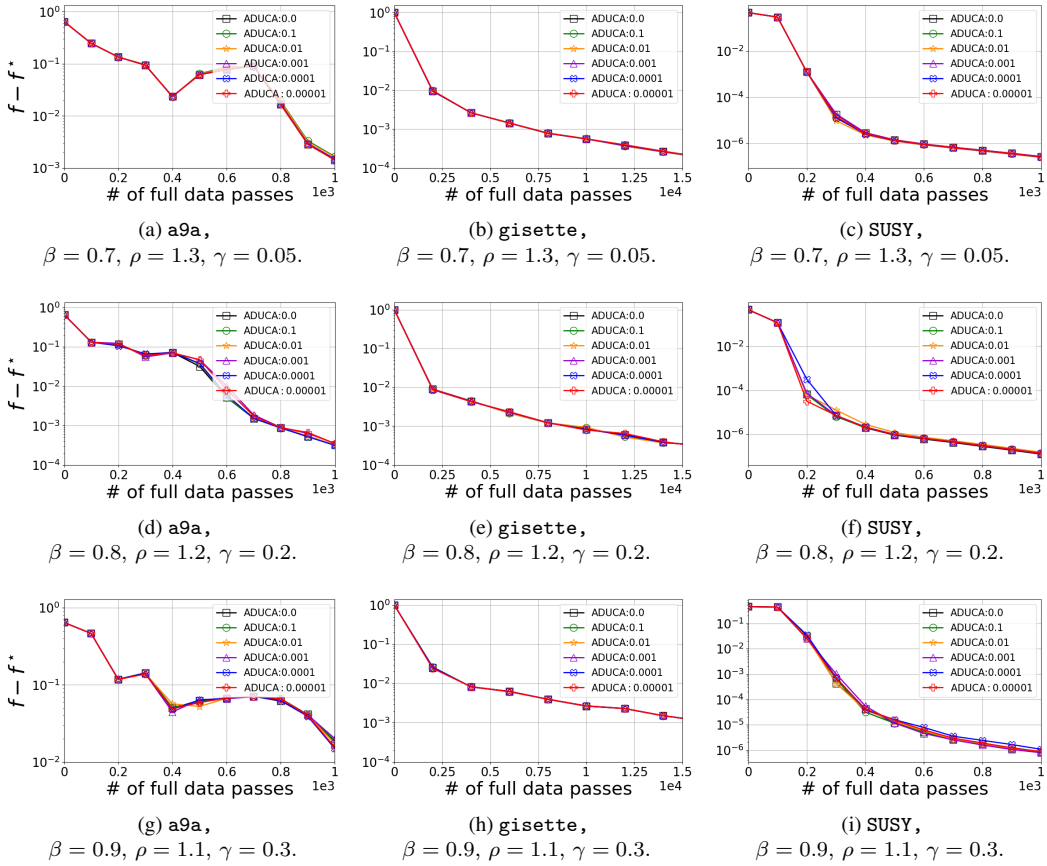
F.2 Local Lipschitz-estimate comparison

We provide illustrative numerical results to explain why the adaptive estimates $\hat{L}_k$ and L_k can lead to faster convergence of `ADUCA` compared to related methods. In Figure 2, we compare different Lipschitz estimates on the regularized support vector machine problem; see Section 6 for details. Here, L is the standard global Lipschitz constant; $\hat{L}$ and $\{L_k\}_{k\geq 1}$ for `CODER-LineSearch` are, respectively, the global block-wise Lipschitz constant with respect to the Mahalanobis norm and the line-search Lipschitz estimates defined in [44]; and $\{L_k\}_{k\geq 1}$ for `GRAAL` is the local Lipschitz estimate with respect to the standard Euclidean norm defined in [31]. On all three datasets, $\hat{L}_k$ and L_k constructed by `ADUCA` provide the tightest estimates.



(a) a9a (b) gisette (c) SUSY

Figure 2: Comparisons of global Lipschitz constants and local Lipschitz estimates used by different algorithms, on the Support Vector Machine problem with datasets a9a, gisette, and SUSY. See Section 6 for further details.



(a) a9a, $\beta = 0.7,\ \rho = 1.3,\ \gamma = 0.05.$ (b) gisette, $\beta = 0.7,\ \rho = 1.3,\ \gamma = 0.05.$ (c) SUSY, $\beta = 0.7,\ \rho = 1.3,\ \gamma = 0.05.$

(d) a9a, $\beta = 0.8,\ \rho = 1.2,\ \gamma = 0.2.$ (e) gisette, $\beta = 0.8,\ \rho = 1.2,\ \gamma = 0.2.$ (f) SUSY, $\beta = 0.8,\ \rho = 1.2,\ \gamma = 0.2.$

(g) a9a, $\beta = 0.9,\ \rho = 1.1,\ \gamma = 0.3.$ (h) gisette, $\beta = 0.9,\ \rho = 1.1,\ \gamma = 0.3.$ (i) SUSY, $\beta = 0.9,\ \rho = 1.1,\ \gamma = 0.3.$

Figure 3: Sensitivity of ADUCA to the hyperparameter μ used in the extrapolation weight ω_k: the numbers in the legend correspond to different choices of μ.

F.3 Ablation analysis for the hyperparameter μ

In this appendix subsection, we study the sensitivity of ADUCA to the hyperparameter μ in Algorithm 1. Recall that μ affects the method only through the extrapolation weight $\omega_k = \frac{1+\rho\beta\mu a_k}{1+\mu a_k}$, so that $\omega_k = 1$ when $\mu = 0$ and $\omega_k \in (\rho\beta, 1)$ when $\mu > 0$. Although μ is required in our theory to obtain near-linear convergence rates under strong convexity, it is typically unknown in practice. Our ablation study shows that, empirically, knowing μ does not meaningfully affect the behavior of ADUCA.

We use the SVM setup from Section 6 and run ADUCA with several choices of μ (reported in the legend), under three different hyperparameter settings (β, ρ, γ) shown in the subplot titles. As shown

in Figure 3, the curves corresponding to different μ values are nearly indistinguishable across datasets and hyperparameter settings. This indicates that the practical performance of ADUCA is largely insensitive to the attenuation introduced by ω_k; consequently, a rough guess for μ, or simply $\mu = 0$, is sufficient, as shown in our experiments.

F.4 Traffic Network Equilibrium with BPR Congestion Costs

Problem formulation. We use traffic assignment as a nonlinear monotone VI benchmark. Traffic equilibrium was one of the classical motivating applications of VI theory [10]; more recently, traffic assignment has also been used as an experimental testbed for first-order algorithms for strongly monotone VIs [42]. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a directed road network with $n_v = |\mathcal{V}|$ nodes and $n_e = |\mathcal{E}|$ directed links. We are given R origin–destination (OD) pairs. For each OD pair $r \in [R]$, let $D_r > 0$ be its demand and let $\mathcal{P}_r$ be a set of admissible paths from the origin to the destination. We write $P_r = |\mathcal{P}_r|$ and $P = \sum_{r=1}^{R} P_r$. The decision variable is the path-flow vector

$$\mathbf{x} = (\mathbf{x}^1, \dots, \mathbf{x}^R) \in \mathbb{R}^P, \quad \mathbf{x}^r \in \mathbb{R}^{P_r},$$

where $\mathbf{x}_p^r$ denotes the amount of OD-r demand assigned to path $p \in \mathcal{P}_r$. The feasible set is the product of OD demand simplexes

$$X = X_1 \times \cdots \times X_R, \quad X_r := \left\{ \mathbf{x}^r \in \mathbb{R}_+^{P_r} : \mathbf{1}_{P_r}^\top \mathbf{x}^r = D_r \right\}.$$

Thus the natural block decomposition has $m = R$ blocks, one for each OD pair.

Let $\boldsymbol{A}_r \in \{0,1\}^{n_e \times P_r}$ be the link–path incidence matrix for OD pair r, and let

$$\boldsymbol{A} := [\boldsymbol{A}_1, \dots, \boldsymbol{A}_R] \in \{0,1\}^{n_e \times P}.$$

The induced link-flow vector is

$$f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} \in \mathbb{R}^{n_e}.$$

Each link $e \in \mathcal{E}$ has a Bureau of Public Roads (BPR) congestion cost

$$t_e(f_e) = t_e^0 \left[1 + \alpha_e \left(\frac{f_e}{c_e} \right)^{q_e} \right],$$

where $t_e^0 > 0$ is the free-flow travel time, $c_e > 0$ is the capacity parameter, $\alpha_e \geq 0$ is the congestion coefficient, and $q_e \geq 1$ is the BPR exponent. Defining $t(f) = (t_e(f_e))_{e \in \mathcal{E}}$, the path-cost operator is

$$F_0(\mathbf{x}) := \boldsymbol{A}^\top t(\boldsymbol{A}\mathbf{x}).$$

The Wardrop equilibrium condition is the Stampacchia Variational Inequality problem:

$$\text{find } \mathbf{x}^\star \in X \quad \text{such that} \quad \langle F_0(\mathbf{x}^\star), \mathbf{x} - \mathbf{x}^\star \rangle \geq 0, \qquad \forall \mathbf{x} \in X.$$

Equivalently, all paths used with positive flow have minimum congested travel time within their OD pair. The operator is monotone because every BPR cost is nondecreasing. Indeed, for any $\mathbf{x}, \boldsymbol{z} \in X$, with $f = \boldsymbol{A}\mathbf{x}$ and $\bar{f} = \boldsymbol{A}\boldsymbol{z}$,

$$\begin{aligned} \langle F_0(\mathbf{x}) - F_0(\boldsymbol{z}), \mathbf{x} - \boldsymbol{z} \rangle &= \langle t(f) - t(\bar{f}), f - \bar{f} \rangle \\ &= \sum_{e \in \mathcal{E}} \left(t_e(f_e) - t_e(\bar{f}_e) \right) \left(f_e - \bar{f}_e \right) \geq 0. \end{aligned}$$

The problem fits our composite GMVI template by taking

$$F_{\mathrm{BPR}}(\mathbf{x}) := F_0(\mathbf{x}) + \lambda_{\mathrm{p}} \mathbf{x}, \quad g_{\mathrm{BPR}}(\mathbf{x}) := \sum_{r=1}^{R} \iota_{X_r}(\mathbf{x}^r).$$

In all reported traffic experiments we use the standard Wardrop model $\lambda_{\mathrm{p}} = 0$. The block prox is therefore only a simplex projection,

$$\mathrm{prox}_{\eta g_r}(\boldsymbol{z}^r) = \Pi_{X_r}(\boldsymbol{z}^r), \qquad g_r = \iota_{X_r}.$$

This makes the instance well matched to cyclic block-coordinate methods: each block update modifies one OD pair's route split while respecting its total demand.

Table 2: Traffic instances used in the BPR experiments. All instances use $R = 150$ OD-pair blocks, $K = 50$ paths per OD pair, and path-flow dimension $d = 7500$. The columns α_e and q_e report the range of BPR congestion parameters used after parsing the network data.

Instance	nodes	links	OD rows	demand	α_e	q_e
Grid2Demand ASU Auto	696	931	4,761	7,503	0.15	4
DTALite Tucson	4,534	12,230	153,509	4,229	0.15	4
TNTP Austin AM	7,388	18,961	1,080,603	15,163	0.83	5.5
Path4GMNS Lima	2,232	6,095	13,000	5,152	0.15	4
TNTP Sydney	33,113	75,379	3,340,619	12,640	0.25–0.67	4
TNTP Berlin Center	12,981	28,376	49,688	7,965	0.15–2.00	4

Experimental setup. We evaluate on six public traffic-assignment networks: three TNTP instances from the TransportationNetworks repository [48], and three GMNS-format networks from the Grid2Demand [50], DTALite [2, 52], and Path4GMNS [41] collections. For every dataset, we select $R = 150$ OD pairs using a high-demand "top-full-blocks" rule: an OD pair is accepted only if at least $K = 50$ feasible paths can be generated. The paths $\mathcal{P}_r$ are the $K = 50$ shortest simple paths under free-flow travel times. Consequently, every instance has

$$m = R = 150, \quad P_r = K = 50, \quad d = P = RK = 7500.$$

All runs use uniform initialization over the paths of each OD pair, demand scale 1.0, no path-flow regularization ($\lambda_{\mathrm{p}} = 0$), and no block rescaling ($\boldsymbol{\Lambda} = \boldsymbol{I}$). We log every full data pass. BPR parameters are read from the data when available; otherwise, we use the fallback values $\alpha_e = 0.15$ and $q_e = 4$.

Wardrop gap. For this experiment, we report the absolute Wardrop gap. Let

$$c(\mathbf{x}) := F_0(\mathbf{x}) = \boldsymbol{A}^\top t(\boldsymbol{A}\mathbf{x}), \quad c^r(\mathbf{x}) := \boldsymbol{A}_r^\top t(\boldsymbol{A}\mathbf{x}) \in \mathbb{R}^{P_r}.$$

The metric is the standard Stampacchia VI gap

$$\mathrm{Gap}_{\mathrm{W}}(\mathbf{x}) := \max_{\boldsymbol{z} \in X} \langle F_0(\mathbf{x}), \mathbf{x} - \boldsymbol{z} \rangle .$$

Because X is a product of OD simplexes, this maximization has the closed form

$$\mathrm{Gap}_{\mathrm{W}}(\mathbf{x}) = \sum_{r=1}^{R} \left[\langle c^r(\mathbf{x}), \mathbf{x}^r \rangle - D_r \min_{p \in \mathcal{P}_r} c_p^r(\mathbf{x}) \right].$$

Thus $\mathrm{Gap}_{\mathrm{W}}(\mathbf{x})$ is the total excess travel cost of the current route assignment relative to assigning each OD demand to a currently cheapest path. It is nonnegative and equals zero precisely when each OD pair sends positive flow only through minimum-cost paths, i.e., when the Wardrop condition is satisfied.

We next relate this metric to the gap notion in our theory. For feasible $\mathbf{x}, \boldsymbol{z} \in X$, the pointwise gap from ($\mathrm{P}_{\mathrm{approx}}$) specializes to

$$\mathrm{Gap}(\mathbf{x}; \boldsymbol{z}) = \langle F_0(\boldsymbol{z}), \mathbf{x} - \boldsymbol{z} \rangle ,$$

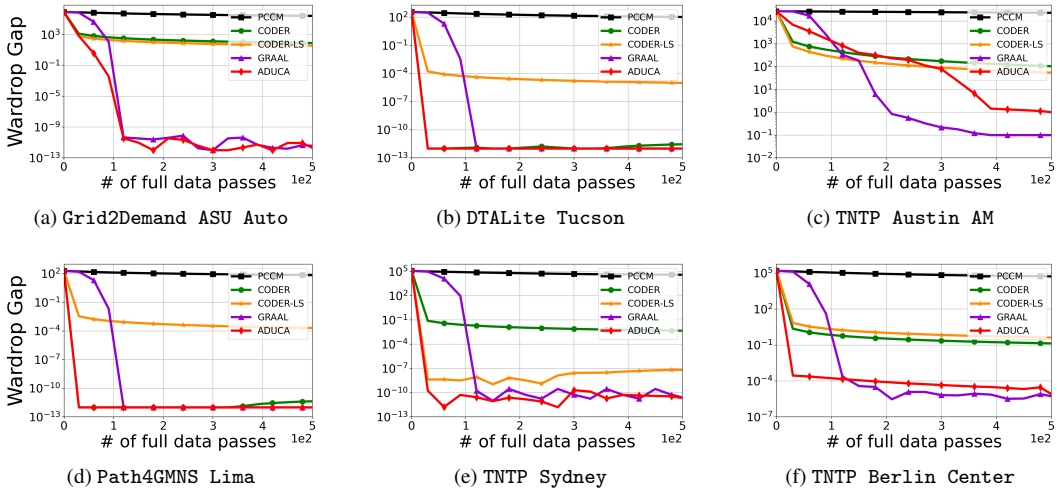
because $g = \iota_X$ and both arguments are feasible. Since F_0 is monotone, for any $\mathbf{x}, \boldsymbol{z} \in X$,

$$\langle F_0(\boldsymbol{z}), \mathbf{x} - \boldsymbol{z} \rangle \le \langle F_0(\mathbf{x}), \mathbf{x} - \boldsymbol{z} \rangle .$$

Consequently,

$$\sup_{\boldsymbol{z} \in X} \mathrm{Gap}(\mathbf{x}; \boldsymbol{z}) \le \mathrm{Gap}_{\mathrm{W}}(\mathbf{x}).$$

Therefore, a small absolute Wardrop gap also controls the restricted worst-case Minty gap used in the theoretical analysis. Moreover, because F_0 is continuous and monotone on the convex feasible set X, the Minty and Stampacchia VI solution sets coincide (see [23]); hence $\mathrm{Gap}_{\mathrm{W}}(\mathbf{x}) = 0$ if and only if $\mathbf{x}$ solves the traffic equilibrium VI.



(a) `Grid2Demand ASU Auto` (b) `DTALite Tucson` (c) `TNTP Austin AM`

(d) `Path4GMNS Lima` (e) `TNTP Sydney` (f) `TNTP Berlin Center`

Figure 4: Absolute Wardrop gap $\mathrm{Gap}_{\mathrm{W}}(\mathbf{x})$ versus the number of full data passes for the BPR traffic equilibrium instances. Lower values are better. Each subplot uses $R = 150$ OD-pair blocks and $K = 50$ generated paths per OD pair, so the path-flow dimension is $d = 7500$.

Performance comparison. Figure 4 shows that `ADUCA` consistently outperforms or remains highly competitive with the other methods across the six nonlinear traffic instances. On Grid2Demand ASU Auto, DTALite Tucson, Path4GMNS Lima, and TNTP Sydney, `ADUCA` reaches zero or numerical-zero absolute Wardrop gap. On TNTP Austin AM, `GRAAL` obtains the smallest final gap, but `ADUCA` is the best cyclic/block-coordinate method by a large margin: its final gap is below 1, while `CODER` and `CODER-LineSearch` remain around 50, and `PCCM` remains above 2×10^4. On TNTP Berlin Center, `ADUCA` and `GRAAL` are essentially tied at small scale, whereas `CODER`, `CODER-LineSearch`, and `PCCM` are visibly worse.

F.5 Assets, licenses, and missing metadata

We use only public benchmark datasets and public transportation-network collections. For the SVM experiments, we use the LIBSVM repository [8]: `a9a`, the LIBSVM preprocessing of the UCI Adult dataset; `gisette`; and `SUSY`. For the traffic experiments in Appendix F.4, we use TNTP/TransportationNetworks instances [48] and GMNS-format networks [43] from Grid2Demand [50], DTALite [2, 52], and Path4GMNS [41]. Specifically, the traffic instances are:

- `tntp_austin`, `tntp_sydney`, and `tntp_berlin_center`, obtained from the TransportationNetworks repository.
- `grid2demand_asu_auto`, obtained from the Grid2Demand collection.
- `dtalite_tucson`, obtained from the DTALite collection.
- `path4gmns_lima_network`, obtained from the Path4GMNS sample networks.

We do not introduce new datasets. The submission package contains only derived figures and experiment logs, not raw third-party data. Dataset licenses and metadata should be checked at the corresponding source repositories before redistribution.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's scope and main claims, and those claims are matched by the theorem statements in Sections 4–5 and the experiments in Section 6.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations and future directions are discussed in the main body and in the appendix. For example, the use of one-time line search is acknowledged in the introduction and technical sections; the assumption about known strong convexity parameter for the purpose of the analysis is acknowledged in Section 3 (with an ablation study provided in Appendix F.3), and asynchronous implementations are proposed as directions for future research in the Conclusion section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Sections 2–5 state the assumptions and main results, while Appendix B–E provide complete proofs and supporting lemmas.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 6 specifies the benchmark problem, datasets, baselines, and evaluation metric, and Appendix F provides the rescaling, hyperparameter, and figure details needed to reproduce the reported comparisons.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The paper describes how to reproduce the experiments using public datasets, but this submission package does not include anonymized code or a runnable supplementary archive.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 6 and Appendix F give the task, datasets, baselines, and main hyperparameters, and Appendix F also provides the diagonal rescaling and additional implementation details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Appendix F explains that the reported curves are deterministic fixed-problem optimization runs with no stochastic sampling or random data splits, so error bars are not applicable here.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We report the compute resources we use in Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The submission is anonymized, uses only public benchmark data, and contains no human-subject experimentation or unsafe release practices that would conflict with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: This is a theoretical work; therefore, no specific societal impacts need to be discussed.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk models, scraped datasets, or other assets that would require special release safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Appendix F.5 credits the LIBSVM repository and the underlying UCI datasets as well as the datasets used in the transportation experiments.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The submission does not introduce or release new datasets, models, or code assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The work does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not an important or non-standard component of the proposed method, the theory, or the experiments.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.