Coordinate Linear Variance Reduction for Generalized Linear Programming

Anonymous Author(s)
Affiliation
Address
email

Abstract

We study a class of generalized linear programs (GLP) in a large-scale setting, which includes a simple, possibly nonsmooth convex regularizer and simple convex set constraints. By reformulating (GLP) as an equivalent convex-concave min-max problem, we show that the linear structure in the problem can be used to design an efficient, scalable first-order algorithm, to which we give the name *Coordinate Linear Variance Reduction* (CLVR; pronounced "clever"). CLVR yields improved complexity results for (GLP) that depend on the max row norm of the linear constraint matrix in (GLP) rather than the spectral norm. When the regularization terms and constraints are separable, CLVR admits an efficient lazy update strategy that makes its complexity bounds scale with the number of nonzero elements of the linear constraint matrix in (GLP) rather than the matrix dimensions. Further, for the special case of linear programs and by exploiting sharpness, we propose a restart scheme for CLVR to obtain empirical linear convergence. Finally, we show that Distributionally Robust Optimization (DRO) problems with ambiguity sets based on both f-divergence and Wasserstein metrics can be reformulated as (GLPs) by introducing sparsely connected auxiliary variables. We complement our theoretical guarantees with numerical experiments that verify our algorithm's practical effectiveness in terms of wall-clock time and number of data passes.

1 Introduction

We study the following generalized linear program (GLP):

$$\min_{\boldsymbol{x}} \ \big\{\boldsymbol{c}^T\boldsymbol{x} + r(\boldsymbol{x}) : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \in \mathcal{X}\big\}, \tag{GLP}$$

where $\boldsymbol{x}, \boldsymbol{c} \in \mathbb{R}^d$, $\boldsymbol{A} \in \mathbb{R}^{n\times d}$, $\boldsymbol{b} \in \mathbb{R}^n$, $r : \mathbb{R}^d \to \mathbb{R}$ is a convex regularizer, and $\mathcal{X} \subseteq \mathbb{R}^d$ is a closed convex set, such that a proximal / projection operator involving r and $\mathcal{X}$ can be computed efficiently. When $\mathcal{X}$ is the nonnegative orthant $\{\boldsymbol{x} : x_i \geq 0, i \in [d]\}$ and $r \equiv 0$, (GLP) reduces to the standard form of a linear program (LP). When $\mathcal{X}$ is a convex cone and $r \equiv 0$, (GLP) reduces to a conic linear program. (GLP) has had a significant impact in traditional engineering disciplines such as transportation, energy, telecommunications, and manufacturing. In modern data science, we observe the renaissance of (GLP) due to its modeling power in such areas as reinforcement learning [18], optimal transport [57], and neural network verification [39]. For traditional engineering disciplines with moderate scale or exploitable sparsity, off-the-shelf interior point methods that form and factorize matrices in each iteration are often good choices as practical solvers [25]. In data science applications, however, where the data are often dense or of extreme scale, the amount of computation and/or memory required by matrix factorization is prohibitive. Thus, first-order methods that avoid matrix factorizations are potentially appealing options. In this context, because the presence of the linear equality constraint in (GLP) may complicate projection operations onto the feasible set, we consider

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

an equivalent reformulation of (GLP) as a min-max problem involving the Lagrangian:

$$\min_{\boldsymbol{x}\in\mathcal{X}\subset\mathbb{R}^d} \max_{\boldsymbol{y}\in\mathbb{R}^n} \left\{\mathcal{L}(\boldsymbol{x},\boldsymbol{y}) := \boldsymbol{c}^T\boldsymbol{x} + r(\boldsymbol{x}) + \boldsymbol{y}^T\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}^T\boldsymbol{b}\right\}. \tag{PD-GLP}$$

In data science applications, both n and d can be very large. (PD-GLP) can be viewed as a structured bilinearly coupled min-max problem, where the linearity of $\mathcal{L}(\boldsymbol{x},\boldsymbol{y})$ in the dual variable vector $\boldsymbol{y}$ is vital to our algorithmic development.

1.1 Background

While the literature addressing (PD-GLP) is sparse — some special cases have been considered in [13, 24, 41–43, 60, 62, 63] — there has been significant recent work on first-order methods for general bilinearly coupled convex-concave min-max problems. Deterministic first-order methods include the proximal point method (PPM) [51], the extragradient/mirror-prox method (EGM) [34, 45], the primal-dual hybrid gradient (PDHG) method [14], and the alternating direction method of multipliers (ADMM) [19]. All these methods have per-iteration cost $\Theta(\mathrm{nnz}(\boldsymbol{A}))$ and convergence rate $1/k$, where $\mathrm{nnz}(\boldsymbol{A})$ denotes the number of nonzero elements of $\boldsymbol{A}$ and k is the number of iterations.

For better scalability, stochastic counterparts of these methods have been proposed. [10, 33, 47, 49] have used "vanilla" stochastic gradients to replace the full gradients of their deterministic counterparts. [2, 12, 27] have exploited the finite-sum structure of the interaction term $\langle\boldsymbol{y},\boldsymbol{A}\boldsymbol{x}\rangle$ involving both primal and dual variables to perform variance reduction. With a separability assumption for the dual variables, [3] and [15] have combined incremental coordinate approaches on the dual variables with an implicit variance reduction strategy on the primal variables. Recently, under a separability assumption for dual variables, [55] have proposed a new incremental coordinate method with an initialization step that requires a single access to the full data. This approach, known as *variance reduction via primal-dual accelerated dual averaging* (VRPDA^2), obtains the first theoretical bounds that are better than their deterministic counterparts in the class of incremental coordinate approaches. The VRPDA^2 algorithm serves as the main motivation for our approach.

It is of particular interest to design algorithms that scale with the number of nonzero elements in $\boldsymbol{A}$ for at least two reasons: (i) the data matrix can be sparse; and (ii) when we consider simplified reformulations of certain complicated models, we often need to introduce sparsely connected auxiliary variables. Nevertheless, the randomized coordinate algorithms of [3, 15, 55] have $O(d)$ per-iteration cost regardless of the sparsity of $\boldsymbol{A}$. To address this issue, [23, 35] have proposed incremental primal-dual coordinate methods with per-iteration cost that scales with the number of nonzero elements in the row from $\boldsymbol{A}$ that is used in each iteration, at the price of smaller step size for dense $\boldsymbol{A}$. Moreover, [5] has proposed a random extrapolation approach that admits both low per-iteration cost and large step size. Despite these developments, all these algorithms produce less accurate iterates than the methods with $O(d)$ per-iteration cost, thus degrading their worst-case complexity.

Finally, for the special case of LP, based on the positive Hoffman constant [30], [9] proved that the primal-dual formulation of LP exhibits a sharpness property, which bounds below the growth of a normalized primal-dual gap from the same work. Based on this sharpness property, [9] proposed a restart scheme for the deterministic first-order methods discussed above to obtain linear convergence. [8] further extended this restart strategy using various heuristics to improve practical performance.

1.2 Motivation

We sharpen the focus from general bilinearly coupled convex-concave min-max problems to (GLP) and its primal-dual formulation (PD-GLP), because many complicated models can be reformulated as (GLP) and (GLP) possesses additional structure that can be exploited in algorithm design. Our motivation for focusing on (GLP) is to bridge the large gap between the well-studied stochastic variance reduced first-order methods [6, 32, 54, 55] and the increasingly popular and complicated, yet highly structured large-scale problems arising in distributionally robust optimization (DRO) [20–22, 31, 38, 44, 52, 58, 61]; see also a recent survey by [50] and references therein.

For DRO problems with ambiguity sets defined by f-divergence [31, 37, 44], the original formulation is a nonbilinearly coupled convex-concave min-max problem. Even the well constructed reformulation in [37] does not admit unbiased stochastic gradients, leading to complicated algorithms and analysis.

For DRO problems with ambiguity sets defined by Wasserstein metric [22, 29, 38, 52, 61], the original formulation is in general infinite-dimensional. (Finite-dimensional reformulations [22, 52] exist for special cases of logistic regression and smooth convex losses.) Solvers that have been proposed for DRO with Wasserstein metric are either multiple-loop deterministic ADMM [38] or are designed for general convex-concave problems [61].

By introducing auxiliary variables with sparse connections,[1] we show that DRO with ambiguity sets based on both f-divergence and the Wasserstein metric can be reformulated as (GLP). Thus, complicated DRO problems can be addressed by a simple, efficient, and scalable algorithm for (GLP). Our algorithm for solving (GLP) and the proposed reformulations of DRO are our main contributions.

1.3 Contributions

Algorithm. Motivated by VRPDA^2 [55], we propose a simple, efficient, and scalable algorithm for (PD-GLP). Our algorithm combines an incremental *coordinate* method with exploitation of the *linear* structure for the dual variables in (PD-GLP) and the implicit *variance reduction* effect in the algorithm, so we name it *coordinate linear variance reduction* (CLVR, pronounced "clever"). CLVR is inspired by VRPDA^2 but customized to the particular structure of (PD-GLP). In particular, by exploiting the fact that the max problem is linear and unconstrained in the dual variable vector $\boldsymbol{y} \in \mathbb{R}^n$, we find that the expensive initialization step used in VRPDA^2 is not needed and we can take simpler and larger steps. Further, in the structured case in which $\boldsymbol{A}$ is sparse and the convex constraint set $\mathcal{X}$ and the regularizer $r(\boldsymbol{x})$ are fully separable[2], we show that the dual averaging update in CLVR enables us to design an efficient lazy update strategy for which the per-iteration cost of CLVR scales with the number of nonzero elements of the selected row from $\boldsymbol{A}$ in each iteration, which is potentially much lower than the order-d cost in VRPDA^2. Finally, CLVR uses extrapolation on dual variables rather than on primal variables considered in VRPDA^2, which significantly reduces implementation complexity of our lazy update strategy for structured variants of (PD-GLP). On the technical side, although both CLVR and VRPDA^2 are randomized algorithms that bound the primal-dual gap in expectation, the guarantee provided by CLVR is stronger as it allows bounding the expectation of the supremum gap as opposed to the supremum of expected gap in VRPDA^2.

To state our complexity results, we make the following scaling assumption.

Assumption 1. $L := \|\boldsymbol{A}\|$ *and each row of* $\boldsymbol{A}$ *in* (GLP) *is normalized with Euclidean norm* R.

As a comment, it is known that preprocessing in modern LP solvers [25] often ensures normalized rows/columns for the data matrix. Observe that $R \leq L \leq \sqrt{n}R$. The upper bound is achieved when all elements of $\boldsymbol{A}$ have identical value. Although this is an extreme case, there exist ill-conditioned practical datasets where we can expect significant performance gains if the complexity can be reduced from $O(L)$ to $O(R)$. In Table 1, we give the overall complexity bounds (total number of arithmetic operations) and the per-iteration cost of a representative set of existing algorithms and our CLVR algorithm for solving structured (PD-GLP), i.e., $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$ with $\mathcal{X}_i \in \mathbb{R}\,(i \in [d])$ and $r(\boldsymbol{x}) := \sum_{i=1}^d r(x^i)$. We further assume that for the stochastic algorithms [2, 15, 55] and our CLVR algorithm, we draw one row of $\boldsymbol{A}$ per iteration uniformly at random to make the complexity results comparable. The general convex setting corresponds to $r(\boldsymbol{x})$ being general convex ($\sigma = 0$), while the strongly convex setting corresponds to $r(\boldsymbol{x})$ being σ-strongly convex ($\sigma > 0$).

As shown in Table 1, all the algorithms have the optimal dependence on ϵ [48], while the dependence on the ambient dimensions n, d, the number of nonzero elements of $\boldsymbol{A}$, $\mathrm{nnz}(\boldsymbol{A})$, and the constants L and R are quite different. For both the general convex and strongly convex settings and among coordinate-type methods, CLVR is the first algorithm that reduces the runtime dependence on the input matrix size from nd to $\mathrm{nnz}(\boldsymbol{A})$. Moreover, the complexity of CLVR depends on the max row norm R rather than the spectral norm L, and the per-iteration cost of CLVR only depends on the nonzero elements of the selected row from $\boldsymbol{A}$ in each iteration, which can be far less than d.

By exploiting the linear structure again, we provide explicit guarantees for both the objective value and the constraint satisfaction of (GLP). Further, the analysis of CLVR applies to the more general

[1] "Sparse connections" here means that even though the newly introduced variables may substantially increase the problem dimensions, the number of nonzero entries in the constraint matrix remains of the same order.

[2] We state the results here for the fully separable setting for convenience of comparison; however, our results are also applicable to the block separable setting.

Table 1: Overall complexity and per-iteration cost for solving structured (PD-GLP). ("—" indicates that the corresponding result does not exist or is unknown.)

Algorithm	General Convex (Primal-Dual Gap)	Strongly Convex (Distance to Solution)	Per-Iteration Cost
PDHG CP(2011)	$O\big(\frac{\mathrm{nnz}(\boldsymbol{A})L}{\epsilon}\big)$	$O\big(\frac{(\mathrm{nnz}(\boldsymbol{A})+n+d)L}{\sigma\sqrt{\epsilon}}\big)$	$O(\mathrm{nnz}(\boldsymbol{A}))$
SPDHG CERS(2018)	$O\big(\frac{ndL}{\epsilon}\big)$	$O\big(\frac{ndL}{\sigma\sqrt{\epsilon}}\big)$	$O(d)$
EVR AM (2021)	$O\big(\mathrm{nnz}(\boldsymbol{A}) + \frac{\sqrt{\mathrm{nnz}(\boldsymbol{A})(n+d)nR}}{\epsilon}\big)$	—	$O(n+d)$
VRPDA[2] SWD(2021)	$O(nd\log\min\{\frac{1}{\epsilon}, n\} + \frac{ndR}{\epsilon})$	$O(nd\log\min\{\frac{1}{\epsilon}, n\} + \frac{ndR}{\sigma\sqrt{\epsilon}})$	$O(d)$
CLVR (**This Paper**)	$O(\frac{\mathrm{nnz}(\boldsymbol{A})R}{\epsilon})$	$O(\frac{\mathrm{nnz}(\boldsymbol{A})R}{\sigma\sqrt{\epsilon}})$	$O(\mathrm{nnz}(\mathrm{row}(\boldsymbol{A})))$

block-coordinate update setting, which is better suited to modern parallel computing platforms. Finally, following the restart strategy based on the *normalized duality gap* for LP introduced in [9], we propose a more straightforward strategy to restart our CLVR algorithm (as well as other iterative algorithms for (PD-GLP)): Restart the algorithm every time a widely known metric for LP optimality [7] halves. Compared with the normalized duality gap, the LP metric can be computed more efficiently and in a more straightforward fashion.

DRO Reformulations. When the loss function is convex, DRO problems with ambiguity sets based on f-divergence [44] or Wasserstein metric [22] are convex. However, as both problems either have complicated constraints or are infinite-dimensional, vanilla first-order methods are inapplicable.

For DRO with f-divergence, we show that by using convex conjugates and introducing auxiliary variables, the problem can be reformulated as a (GLP). As a result, we do not have the issue of biased stochastic gradients encountered in [37] and CLVR can be applied. Even though the resulting problem has larger dimensions, due to the sparseness of the introduced auxiliary variables and the lazy update strategy of CLVR, it can be solved with complexity scaling only with the number of nonzero elements of the data matrix. Due to being cast as a (GLP), the DRO problem can be solved with $O(1/\epsilon)$ iteration complexity with CLVR, while existing methods such as [37] have $O(1/\epsilon^2)$ iteration complexity, with higher iteration cost because of the batch of samples needed to reduce bias. This improvement is enabled in part by considering the primal-dual gap (rather than the primal gap considered in [37]) and by allowing the constraints to be approximately satisfied (see Corollary 1).

For DRO with Wasserstein metric, following the reformulation of [52, Theorem 1], we show further that the problem can be cast in the form of (GLP). Compared with the existing reformulations [22, 38, 52, 61], our reformulation can handle both smooth and nonsmooth convex loss functions. In fact, our reformulation can provide a more compact form for nonsmooth piecewise-linear convex loss functions (such as hinge loss). Moreover, compared with algorithms customized to this problem [38] and extragradient methods [34, 45, 61] for general convex-concave min-max problems, our CLVR method attains the best-known iteration complexity and per-iteration cost, as shown in Table 1.

2 Notation and Preliminaries

For any positive integer p, we use $[p]$ to denote $\{1, 2, \ldots, p\}$. We assume that there is a given partition of the set $[n]$ into sets S^j, $j \in [m]$, where $|S^j| = n^j > 0$ and $\sum_{j=1}^m n^j = n$. For $j \in [m]$, we use $\boldsymbol{A}^{S^j}$ to denote the submatrix of $\boldsymbol{A}$ with rows indexed by S^j and $\boldsymbol{y}^{S^j}$ to denote the subvector of $\boldsymbol{y}$ indexed by S^j. We use $\mathbf{0}_d$ and $\mathbf{1}_d$ to denote the vectors with all ones and all zeros in d dimensions, respectively. Unless otherwise specified, we use $\|\cdot\|$ to denote the Euclidean norm for vectors and the spectral norm for matrices. For a given proper convex lower semi-continuous function $f : \mathbb{R} \to \mathbb{R} \cup \{+\infty\}$, we define the convex conjugate in the standard way as $f^*(y) = \sup_{x\in\mathbb{R}}\{yx - f(x)\}$ (so that $f^{**} = f$). For a vector $\boldsymbol{u}$, the inequality $\boldsymbol{u} \geq \mathbf{0}$ is applied entry-wise. For a convex function $r(\boldsymbol{x})$, we use $r'(\boldsymbol{x})$ to denote an element of the subdifferential set $\partial r(\boldsymbol{x})$. The proximal operator of $r(\boldsymbol{x})$ over $\mathcal{X}$ is

$$\mathrm{prox}_r(\hat{\boldsymbol{x}}) = \arg\min_{\boldsymbol{x}\in\mathcal{X}} \Big\{\frac{1}{2}\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2 + r(\boldsymbol{x})\Big\}. \tag{1}$$

Further, we make the following assumptions, which apply throughout the convergence analysis.

Assumption 2. (PD-GLP) *attains at least one primal-dual solution* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$. $\mathcal{W}^*$ *denotes the set of all primal-dual solutions.*

Due to the convex-concave property of (PD-GLP), $\mathcal{W}^*$ is a convex set in $\mathcal{X} \times \mathbb{R}^n$.

Assumption 3. $\hat{L} = \max_{j\in[m]} \|\boldsymbol{A}^{S^j}\|$ *is given at the input, where* $\|\boldsymbol{A}^{S^j}\| = \max_{\|\boldsymbol{x}\|\le 1} \|\boldsymbol{A}^{S^j}\boldsymbol{x}\|$.

Note that $\hat{L}$ can be obtained either via preprocessing of the data or by parameter tuning. By combining Assumptions 1 and 3, it follows that $R \le \hat{L} \le \sqrt{\max_{j\in[m]} |S^j|}R$.

Assumption 4. $r(\boldsymbol{x})$ *is* σ*-strongly convex* $(\sigma \ge 0)$*; that is, for all* $\boldsymbol{x}_1$ *and* $\boldsymbol{x}_2$ *in* $\mathcal{X}$ *and all* $r'(\boldsymbol{x}_2) \in \partial r(\boldsymbol{x}_2)$*, we have* $r(\boldsymbol{x}_1) \ge r(\boldsymbol{x}_2) + \langle r'(\boldsymbol{x}_2), \boldsymbol{x}_1 - \boldsymbol{x}_2\rangle + \frac{\sigma}{2}\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2$.

For convex-concave min-max problems, a common metric for measuring solution quality is the primal-dual gap, which, for a feasible solution $(\boldsymbol{x}, \boldsymbol{y})$ of (PD-GLP), is defined by

$$\sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{X}\times\mathbb{R}^n} \{\mathcal{L}(\boldsymbol{x}, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \boldsymbol{y})\}. \tag{2}$$

However, as the domain of $\boldsymbol{v}$ is unbounded, the primal-dual gap can be infinite, which makes it a poor metric for measuring the progress of algorithms. As a result, for measuring the progress of our algorithm, we consider the following restricted primal-dual gap instead:

$$\sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{W}} \{\mathcal{L}(\boldsymbol{x}, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \boldsymbol{y})\}, \tag{3}$$

where $\mathcal{W} \subset \mathcal{X} \times \mathbb{R}^n$ is a compact (i.e., closed and bounded) convex set. Such a restricted version of primal-dual gap is standard in the existing literature; see, e.g., [14, 46].

3 The CLVR Algorithm

3.1 Algorithm and Analysis for General Formulation

Algorithm 1 specifies CLVR for (PD-GLP) in the general setting. The algorithm alternates the full update for $\boldsymbol{x}_k$ in Step 4 ($O(d)$ cost) with an incremental block coordinate update for $\boldsymbol{y}_k$ in Steps 5 and 6 (with $O(|S^{j_k}|d)$ cost for dense $\boldsymbol{A}$). The auxiliary variables $\boldsymbol{z}_k$ and $\boldsymbol{q}_k$ accumulate the cancellation terms in the estimation sequence and give a pathway to a straightforward development of the lazified CLVR, which appears as Algorithm 2 in the appendix. The cost of updating auxiliary vectors $\boldsymbol{z}_k$ and $\boldsymbol{q}_k$ is $O(|S^{j_k}|d)$ and $O(d)$, respectively. In essence, CLVR is a primal-dual coordinate method that uses a *dual averaging* update for $\boldsymbol{x}_k$, then updates the state variables $\{\boldsymbol{q}_k\}$ by a *linear recursion*, and computes $\boldsymbol{x}_k$ from $\boldsymbol{q}_{k-1}$ via a *proximal step* without direct dependence on $\boldsymbol{x}_{k-1}$. The output $\tilde{\boldsymbol{x}}_K$ is a convex combination of the iterates $\{\boldsymbol{x}_k\}_{k=1}^K$, as is standard for primal-dual methods. However, $\tilde{\boldsymbol{y}}_K$ is only an *affine* (not convex) combination of $\{\boldsymbol{y}_k\}_{k=0}^K$, as it involves the term $-(m-1)\boldsymbol{y}_0$ (whose coefficient is negative) and some of the coefficients $ma_k - (m-1)a_{k+1}$ multiplying $\boldsymbol{y}_k$ for $k \in \{1, \dots K-1\}$ may also be negative. An affine combination still provides valid bounds because the dual variable vector $\boldsymbol{y}$ appears linearly in (PD-GLP). Moreover, in Step 9, the term $ma_k(\boldsymbol{z}_k - \boldsymbol{z}_{k-1})$ serves to cancel certain errors from the randomization of the update w.r.t. $\boldsymbol{y}_k$, thus playing a key role in implicit variance reduction.

Theorem 1 provides the convergence results for Algorithm 1. The proof is provided in Appendix B. In the theorem (as in the algorithm), γ is a positive parameter that can be tuned.

Theorem 1. *Let* $\boldsymbol{x}_k$*,* $\boldsymbol{y}_k$*,* $k \in [K]$*, be the iterates of Algorithm 1 and let* $\tilde{\boldsymbol{x}}_k$*,* $\tilde{\boldsymbol{y}}_k$ *be defined by*

$$\tilde{\boldsymbol{x}}_k = \frac{1}{A_k}\sum_{i=1}^k a_i\boldsymbol{x}_i, \quad \tilde{\boldsymbol{y}}_k = \frac{1}{A_k}\sum_{i=1}^k (a_i\boldsymbol{y}_i + (m-1)a_i(\boldsymbol{y}_i - \boldsymbol{y}_{i-1})), \tag{4}$$

for $k \in [K]$*. Let* $\mathcal{W}_k \subset \mathcal{X} \times \mathbb{R}^n$*,* $k \in [K]$*, be a sequence of compact convex sets such that* $(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{y}}_k) \in \mathcal{W}_k \subset \mathcal{W} \subset \mathcal{X} \times \mathbb{R}^n$*, where* $\mathcal{W}$ *is also convex and compact. Then:*

$$\begin{aligned}&\mathbb{E}\Big[\sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{W}_k} \{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)\}\Big] \\ &\le \frac{1}{A_k}\left(\mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{u}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\hat{\boldsymbol{v}} - \boldsymbol{y}_0\|^2\Big] + \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2\right),\end{aligned} \tag{5}$$

Algorithm 1 Coordinate Linear Variance Reduction (CLVR)

Input: $\boldsymbol{x}_0 \in \mathcal{X}, \boldsymbol{y}_0 \in \mathbb{R}^n, \boldsymbol{z}_0 = \boldsymbol{A}^T\boldsymbol{y}_0, \gamma > 0, \hat{L} > 0, \sigma \geq 0, K, m, \{S^1, S^2, \ldots, S^m\}$.
$a_1 = A_1 = \frac{1}{2\hat{L}m}, \boldsymbol{q}_0 = a_1(\boldsymbol{z}_0 + \boldsymbol{c})$.
for $k = 1, 2, \ldots, K$ **do**
  $\boldsymbol{x}_k = \mathrm{prox}_{\frac{1}{\gamma}A_k r}(\boldsymbol{x}_0 - \frac{1}{\gamma}\boldsymbol{q}_{k-1})$.
  Pick j_k uniformly at random in $[m]$.
  $\boldsymbol{y}_k^{S^i} = \begin{cases} \boldsymbol{y}_{k-1}^{S^i}, & i \neq j_k \\ \boldsymbol{y}_{k-1}^{S^i} + \gamma m a_k\big(\boldsymbol{A}^{S^i}\boldsymbol{x}_k - \boldsymbol{b}^{S^i}\big), & i = j_k \end{cases}$.
  $a_{k+1} = \frac{\sqrt{1+\sigma A_k/\gamma}}{2\hat{L}m}, A_{k+1} = A_k + a_{k+1}$.
  $\boldsymbol{z}_k = \boldsymbol{z}_{k-1} + \boldsymbol{A}^{S^{j_k},T}(\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}})$.
  $\boldsymbol{q}_k = \boldsymbol{q}_{k-1} + a_{k+1}\big(\boldsymbol{z}_k + \boldsymbol{c}\big) + m a_k(\boldsymbol{z}_k - \boldsymbol{z}_{k-1})$.
end for
return $\tilde{\boldsymbol{x}}_K = \frac{1}{A_K}\sum_{k=1}^K a_k\boldsymbol{x}_k, \tilde{\boldsymbol{y}}_K = \frac{1}{A_K}\sum_{k=1}^K (a_k\boldsymbol{y}_k + (m-1)a_k(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}))$.

where $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}) = \arg\sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{W}_k}\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)\}$. *Furthermore,*

$$\mathbb{E}\Big[\frac{\gamma + \sigma A_k}{4}\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}^*\|^2\Big] \leq \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2. \tag{6}$$

Define $K_0 = \lceil \frac{\sigma}{18\hat{L}m\gamma} \rceil$. *Then in the bounds above:*

$$A_k \geq \max\left\{\frac{k}{2\hat{L}m}, \frac{\sigma}{(6\hat{L}m)^2\gamma}\Big(k - K_0 + \max\Big\{3\sqrt{2\hat{L}m\gamma/\sigma}, 1\Big\}\Big)^2\right\}.$$

Observe that $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}})$ in the theorem statement exists because of compactness of $\mathcal{W}_k$ and our assumptions on $r(\cdot)$. The parameter γ can be tuned to balance the relative weights of primal and dual initial quantities $\|\boldsymbol{x}^* - \boldsymbol{x}_0\|$ and $\|\boldsymbol{y}^* - \boldsymbol{y}_0\|$ (or estimates of these quantities), which can significantly influence practical performance of the method.

In addition to the guarantee on the variational form, due to the linear structure, we also provide explicit guarantees for both the objective and the constraints in (GLP), stated in the following corollary.

Corollary 1. *In Algorithm 1, for all* $k \geq 1$, $\tilde{\boldsymbol{x}}_k$ *satisfies*

$$\mathbb{E}[\|\boldsymbol{y}^*\| \cdot \|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|] \leq \frac{\gamma\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2 + \frac{1}{\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2]}{A_k},$$

$$|\mathbb{E}[(\boldsymbol{c}^T\tilde{\boldsymbol{x}}_k + r(\tilde{\boldsymbol{x}}_k)) - (\boldsymbol{c}^T\boldsymbol{x}^* + r(\boldsymbol{x}^*))]| \leq \frac{\gamma\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2 + \frac{1}{\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2]}{A_k},$$

where $\boldsymbol{v} = 2\frac{\|\boldsymbol{y}^*\|}{\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|}(\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b})$.

In CLVR, we allow for arbitrary $(\boldsymbol{x}_0, \boldsymbol{y}_0) \in \mathcal{X} \times \mathbb{R}^n$. Nevertheless, by setting $\boldsymbol{y} = \boldsymbol{0}_n$, we can obtain $\boldsymbol{z}_0 = \boldsymbol{0}_d$ at no cost — a useful strategy for large-scale problems since it avoids the (potentially expensive) single matrix-vector multiplication w.r.t. $\boldsymbol{A}$.

3.2 Lazy Update for Sparse and Structured (PD-GLP)

In Algorithm 1, direct computation of the iterates $(\boldsymbol{x}_k, \boldsymbol{y}_k)$ and the output points $(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{y}}_k)$ can be expensive. However, [17] showed that it is possible to only update the averaged vector in the coordinate block chosen for that iteration. This strategy requires us to record the most recent update for each coordinate block and update it only when it is selected again, which is tricky and needs to be implemented carefully. For sparse and block coordinate-separable instances of (PD-GLP), we show that by introducing auxiliary variables that are sparsely connected, we can significantly simplify CLVR and make its complexity scale independently of the ambient dimension $n \cdot d$, instead scaling with $\mathrm{nnz}(\boldsymbol{A})$. Due to space constraints, we defer technical details, including the lazy version of CLVR and associated proofs, to Appendix A.

3.3 Restart Scheme

We now propose a fixed restart strategy with a fixed number of iterations per each restart epoch and discuss an adaptive restart strategy for the special case of standard-form LP, which corresponds to (GLP) with $r(\boldsymbol{x}) \equiv 0$ and $\mathcal{X} = \{\boldsymbol{x} : x_i \geq 0, i \in [d]\}$. We write

$$\min_{\boldsymbol{x}} \ \boldsymbol{c}^T \boldsymbol{x} \ \text{ s.t. } \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \ \boldsymbol{x} \geq \boldsymbol{0}_d, \tag{LP}$$

and the primal-dual form

$$\min_{\boldsymbol{x} \geq \boldsymbol{0}_d} \max_{\boldsymbol{y} \in \mathbb{R}^n} \Big\{ \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{y}^T \boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}^T \boldsymbol{b} \Big\}. \tag{PD-LP}$$

This problem has a sharpness property that can be used to obtain linear convergence in first-order methods [9]. For convenience, in the following, we define $\boldsymbol{w} = (\boldsymbol{x}, \boldsymbol{y}), \hat{\boldsymbol{w}} = (\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}), \tilde{\boldsymbol{w}} = (\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}})$ and $\boldsymbol{w}^* = (\boldsymbol{x}^*, \boldsymbol{y}^*)$. Meanwhile, for $\gamma > 0$, we denote the weighted norm $\|\boldsymbol{w}\|_{(\gamma)} := \sqrt{\gamma \|\boldsymbol{x} - \boldsymbol{x}^*\|_2^2 + \frac{1}{\gamma}\|\boldsymbol{y} - \boldsymbol{y}^*\|_2^2}$. Further, we use $\mathcal{W}^*$ to denote the optimal solution set of the LP and define the distance to $\mathcal{W}^*$ by $\text{dist}(\boldsymbol{w}, \mathcal{W}^*)_{(\gamma)} = \min_{\boldsymbol{w}^* \in \mathcal{W}^*} \|\boldsymbol{w} - \boldsymbol{w}^*\|_{(\gamma)}$. When $\gamma = 1$, $\|\cdot\|_{(\gamma)}$ is the standard Euclidean norm. Then based on (PD-LP), we can use the following classical LP metric[3] to measure the progress of iterative algorithms for LP:

$$\begin{aligned} &\text{LPMetric}(\boldsymbol{x}, \boldsymbol{y}) \\ &= \sqrt{\|\max\{-\boldsymbol{x}, \boldsymbol{0}\}\|_2^2 + \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \|\max\{-\boldsymbol{A}^T\boldsymbol{y} - \boldsymbol{c}, \boldsymbol{0}\}\|_2^2 + |\max\{\boldsymbol{c}^T\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{y}, 0\}|^2}, \end{aligned} \tag{7}$$

which can be explicitly and directly computed. For the Euclidean case ($\gamma = 1$), it is well-known [30] that there exists a Hoffman constant H_1 such that

$$\text{LPMetric}(\boldsymbol{w}) \geq H_1 \text{dist}(\boldsymbol{w}, \mathcal{W}^*)_{(1)}. \tag{8}$$

Using the equivalence of norms in finite dimensions, for general $\gamma > 0$, we can conclude that there exists another constant H_γ (to which we refer as the generalized Hoffman's constant) such that

$$\text{LPMetric}(\boldsymbol{w}) \geq H_\gamma \text{dist}(\boldsymbol{w}, \mathcal{W}^*)_{(\gamma)}. \tag{9}$$

Using Eq. (9) and Theorem 1, we then obtain the following bounds for distance and LPMetric.

Theorem 2. *Consider the* CLVR *algorithm applied to the standard-form LP problem* (PD-LP)*, with input $\boldsymbol{w}_0$ and output $\tilde{\boldsymbol{w}}_k$. Given $\gamma > 0$, define $\boldsymbol{w}^* = \arg\min_{w \in \mathcal{W}^*} \|\boldsymbol{w}_0 - \boldsymbol{w}\|_{(\gamma)}$, and define $C_0 = \gamma + 1/\gamma + (\sqrt{2} + 1)\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \|\boldsymbol{w}^*\|_{(\gamma)}$. Then for H_γ defined as in* (9)*, we have*

$$\mathbb{E}\Big[\sqrt{\text{dist}(\tilde{\boldsymbol{w}}_k, \mathcal{W}^*)_{(\gamma)}}\,\Big] \leq 5\sqrt{\frac{\hat{L}mC_0}{H_\gamma k}}\sqrt{\text{dist}(\boldsymbol{w}_0, \mathcal{W}^*)_{(\gamma)}},$$

$$\mathbb{E}\big[\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}\big] \leq 5\sqrt{\frac{\hat{L}mC_0}{H_\gamma k}}\sqrt{\text{LPMetric}(\boldsymbol{w}_0)}.$$

As a result, by Theorem 2, if we know the values of $\hat{L}$, $\|\boldsymbol{w}^*\|_{(\gamma)}$ and H_γ, then by setting $k = \frac{100\hat{L}mC_0}{H_\gamma k}$, we can halve the square root of the distance and the LPMetric in expectation. Thus we can obtain linear convergence if we restart the CLVR algorithm after a fixed number of iterations. However, the values of $\|\boldsymbol{w}^*\|_{(\gamma)}$ and H_γ are often unknown and thus make this strategy unrealistic in practice.

Compared with the above fixed restart strategy, a natural strategy is to restart whenever the LPMetric halves (summarized in Algorithm 4 in the appendix). Since LPMetric is easy to monitor and update, implementation of this strategy is straightforward. However, bounding the number of iterations required to halve the metric (in expectation or with high probability) seems nontrivial. What can be said (based on Theorem 2 and denoting by K the number of iterations on CLVR between restarts) is that $\mathbb{P}[K > \frac{50\hat{L}mC_0}{\delta^2 H_\gamma}] \leq \delta$. This follows by Markov inequality, as $\mathbb{P}[K > k] = \mathbb{P}\Big[\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)} > \sqrt{\frac{\text{LPMetric}(\boldsymbol{w}_0)}{2}}\Big] \leq 5\sqrt{2\frac{\hat{L}mC_0}{H_\gamma k}}$. Although we use adaptive restart in our experiments, we defer its convergence analysis to future work. Finally, as an independent and parallel work to ours, [40] proposed a high probability guarantee for scheduled restart for stochastic extragradient-type methods.

[3] In (PD-LP), we dualize the constraint $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ by $\boldsymbol{y}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})$ instead of $\boldsymbol{y}^T(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x})$, so in our LP metric, there exist a sign difference for $\boldsymbol{y}$ from the more common representation such as the one in [9].

4 Application: DRO

Consider sample vectors $\{\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n\}$ with labels $\{b_1, b_2, \ldots, b_n\}$, where $b_i \in \{1, -1\}$ $(i \in [n])$. The DRO problem with f-divergence based ambiguity set is

$$\min_{\boldsymbol{x} \in \mathcal{X}} \sup_{\boldsymbol{p} \in \mathcal{P}_{\rho,n}} \sum_{i=1}^{n} p_i g(b_i \boldsymbol{a}_i^T \boldsymbol{x}), \tag{10}$$

where $\mathcal{P}_{\rho,n} = \left\{\boldsymbol{p} \in \mathbb{R}^n : \sum_{i=1}^n p_i = 1, p_i \geq 0\,(i \in [n]), D_f(\boldsymbol{p} \| \mathbf{1}/n) \leq \frac{\rho}{n}\right\}$ is the ambiguity set, g is a convex loss function and D_f is an f-divergence defined by $D_f(\boldsymbol{p} \| \boldsymbol{q}) = \sum_{i=1} q_i f(p_i/q_i)$ with $\boldsymbol{p}, \boldsymbol{q} \in \left\{\boldsymbol{p} \in \mathbb{R}^n : \sum_{i=1}^n p_i = 1, p_i \geq 0\right\}$ and f being a convex function [44]. The formulation (10) is a nonbilinearly coupled convex-concave min-max problem with constraint set $\mathcal{P}_{\rho,n}$ for which efficient projections are not available in general. When g is a nonsmooth loss (e.g., the hinge loss), many well-known methods such as the extragradient [34, 45] cannot be used even if we could project onto $\mathcal{P}_{\rho,n}$ efficiently. However, by introducing auxiliary variables, additional linear constraints, and simple convex constraints, we can make the interacting term between primal and dual variables bilinear, as shown next. (See Appendix C for a proof.)

Theorem 3. *Let $\mathcal{X}$ be a compact convex set. Then the DRO problem in Eq.* (10) *is equivalent to*

$$\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{w},\boldsymbol{\mu},\boldsymbol{q},\gamma} \quad & \left\{\gamma + \frac{\rho\mu_1}{n} + \frac{1}{n}\sum_{i=1}^{n} \mu_i f^*\left(\frac{q_i}{\mu_i}\right)\right\} \\
\text{s.t.} \quad & \boldsymbol{w} + \boldsymbol{v} - \frac{\boldsymbol{q}}{n} - \gamma \mathbf{1}_n = \mathbf{0}_n, \\
& u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, && i \in [n] \\
& \mu_1 = \mu_2 = \cdots = \mu_n, \\
& g(u_i) \leq w_i, && i \in [n] \\
& q_i \in \mu_i \operatorname{dom}(f^*), && i \in [n] \\
& v_i \geq 0,\ \mu_i \geq 0, && i \in [n] \\
& \boldsymbol{x} \in \mathcal{X}.
\end{aligned}$$

In Theorem 3, the domain of the one-dimensional convex function $f^*(\cdot)$ is an interval such as $[a, b]$, so that $q_i \in \mu_i \operatorname{dom}(f^*)$ denotes the inequality $\mu_i a \leq q_i \leq \mu_i b$. Since the perspective function $\mu f^*\left(\frac{q}{\mu}\right)$ is a simple convex function of two variables, we can assume that the proximal operator for this function on the domain $\{(\mu, q) : q \in \mu \operatorname{dom}(f^*), \mu > 0\}$ can be computed efficiently [11]. Similarly, we can assume that the constraint $g(u) \leq w$ admits an efficiently computable projection operator. As a result, the formulation (10) can be solved by CLVR. When expressing (10) in the form of (PD-GLP), the primal and dual variable vectors have dimensions $d + 1 + 4n$ and $3n - 1$, respectively. However, according to Table 1, provided that $\mathcal{X}$ is coordinate separable, the overall complexity of CLVR will only be $O\left(\frac{(\operatorname{nnz}(\boldsymbol{A})+n)(R+1)}{\epsilon}\right)$.

The original DRO problem with Wasserstein metric based ambiguity set is an *infinite*-dimensional *nonbilinearly* coupled convex-concave min-max problem defined by

$$\min_{\boldsymbol{w} \in \mathbb{R}^d} \sup_{\mathbb{P} \in \mathcal{P}_{\rho,\kappa}} \mathbb{E}^{\mathbb{P}}[g(b\boldsymbol{a}^T \boldsymbol{w})], \tag{11}$$

where $\boldsymbol{a} \in \mathbb{R}^d, b \in \{1, -1\}$, $\mathbb{P}$ is a distribution on $\mathbb{R}^d \times \{1, -1\}$, g is a convex loss function and $\mathcal{P}_{\rho,\kappa}$ is the Wasserstein metric-based ambiguity set [52]. Our reformulation for Eq. (11) is in Appendix C.2.

5 Numerical Experiments and Discussion

We provide experimental evaluations of our algorithm for the reformulation of the DRO with Wasserstein metric of ℓ_1-norm (with $\kappa = 0.1$ and $\rho = 10$) and hinge loss. For its LP formulation (see Theorem 4 in the Appendix), we compare our CLVR method with three representative methods: PDHG [14], SPDHG [15] and PURE-CD [5]. For all algorithms we use LPMetric (7) as the performance measure and use a restart strategy of successive halving of LPMetric (Section 3.3) to obtain linear convergence. We implemented CLVR and other algorithms in Julia, optimizing all implementations to the best of our ability. Full details for the experimental setting can be found in Appendix D.

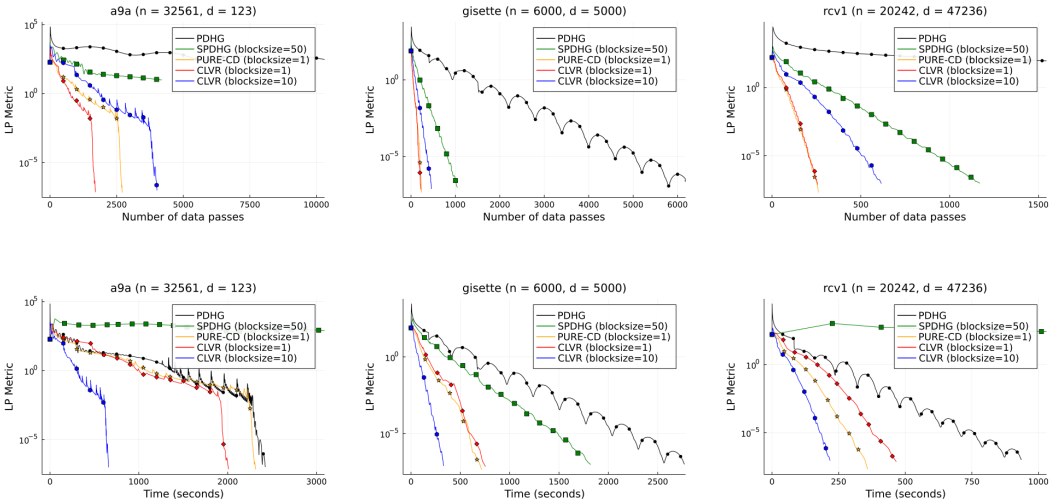

Figure 1: Comparison of numerical results in terms of number of data passes and wall-clock time.

Table 2: Comparison of numerical results against off-the-shelf solvers. Each value represents the time required by a solver/algorithm to reach accuracy $< 10^{-8}$.

Time (seconds)	Reformulated a9a $d = 130738, n = 97929$	Reformulated gisette $d = 44002, n = 28000$	Reformulated rcv1 $d = 269914, n = 155198$
JuMP+GLPK	899	$> 4 \times 10^4$	$> 4 \times 10^4$
JuMP+Gurobi(simplex)	893	2482	7008
JuMP+Gurobi(barrier)	**26**	1039.7	1039.5
CLVR	962	**697**	**582**

Comparison with Primal-Dual Algorithms. Figure 1 provides an algorithm comparison in terms of the number of data passes and wall-clock time. The spikes in all the plots are due restarts; at the beginning of each cycle, the value of LP metric increases significantly, then decreases rapidly. For the number of data passes (top row), CLVR with block size 1 and PURE-CD perform best on all three datasets. CLVR with block size 10 and SPDHG with block size 50 have second-tier performance, while PDHG is worst. For the CLVR algorithm, smaller block size corresponds to smaller $\hat{L}$ in Assumption 3, which corresponds to better complexity in terms of data passes by Theorem 1. Nevertheless, the gap between empirical performance and theoretical guarantee for SPDHG and PURE-CD deserves further research because, to date, they have only been shown to have the same iteration complexity as PDHG. Empirically on `a9a`, CLVR with block size 1 performs better than PURE-CD in terms of data passes.

In terms of wall-clock time (bottom row), because of different per-iteration costs of each algorithm and instruction-level parallelism in modern processors [28], the plots differ significantly from the plots for number of data passes. Even with block size 50, SPDHG spends the most wall-clock time for one data pass and is the slowest on sparse datasets `a9a` and `rcv1`, but is faster than PDHG on the dense dataset `gisette`. Meanwhile, while CLVR with block size 10 is not best in terms of data passes, it remains fastest in terms of wall-clock time on all datasets due to cheaper per-iteration cost and instruction-level parallelism. On `rcv1`, the per-iteration cost of PURE-CD is about 60% of that of CLVR with block size 1. Hence, despite having similar performance in terms of data passes, PURE-CD is faster than CLVR with block size 1, but is still slower than CLVR with block size 10.

Comparison with Off-the-shelf Solvers. Table 2 shows that the performance of CLVR is competitive against state-of-the-art solvers such as GLPK [1] and Gurobi [26]. Moreover, we observe that CLVR reached accurate solutions significantly faster than GLPK and Gurobi in the reformulated problems with `gisette` and `rcv1` datasets. Although CLVR is much slower than Gurobi(barrier) on `a9a` dataset, we believe that much of the performance gap in this case is due to the redundancy in

the problem formulation with the a9a dataset, much of which is removed by Gurobi presolver[4]. We leave presolving and other heuristic speedups of CLVR for future work.

CLVR is fastest in both the number of data passes and wall-clock time among all primal-dual algorithms we compared against on these datasets and it is also competitive with off-the-shelf solvers. Since it has the best known theoretical guarantee, we believe that CLVR could be a method of choice.

References

[1] GLPK (GNU linear programming kit). `https://www.gnu.org/software/glpk/`, 2022.

[2] A. Alacaoglu and Y. Malitsky. Stochastic variance reduction for variational inequality methods. *arXiv preprint arXiv:2102.08352*, 2021.

[3] A. Alacaoglu, Q. T. Dinh, O. Fercoq, and V. Cevher. Smooth primal-dual coordinate descent algorithms for nonsmooth convex optimization. In *Proc. NIPS'17*, 2017.

[4] A. Alacaoglu, O. Fercoq, and V. Cevher. On the convergence of stochastic primal-dual hybrid gradient. *arXiv preprint arXiv:1911.00799*, 2019.

[5] A. Alacaoglu, O. Fercoq, and V. Cevher. Random extrapolation for primal-dual coordinate descent. In *Proc. ICML'20*, 2020.

[6] Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. In *Proc. ACM STOC'17*, 2017.

[7] E. D. Andersen and K. D. Andersen. The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*, pages 197–232. Springer, 2000.

[8] D. Applegate, M. Díaz, O. Hinder, H. Lu, M. Lubin, B. O'Donoghue, and W. Schudy. Practical large-scale linear programming using primal-dual hybrid gradient. *arXiv preprint arXiv:2106.04756*, 2021.

[9] D. Applegate, O. Hinder, H. Lu, and M. Lubin. Faster first-order primal-dual methods for linear programming using restarts and sharpness. *arXiv preprint arXiv:2105.12715*, 2021.

[10] P. Bianchi. Ergodic convergence of a stochastic proximal point algorithm. *SIAM Journal on Optimization*, 26(4):2235–2260, 2016.

[11] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[12] Y. Carmon, Y. Jin, A. Sidford, and K. Tian. Variance reduction for matrix games. In *Proc. NeurIPS'19*, 2019.

[13] Y. Carmon, Y. Jin, A. Sidford, and K. Tian. Coordinate methods for matrix games. *arXiv preprint arXiv:2009.08447*, 2020.

[14] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40(1):120–145, 2011.

[15] A. Chambolle, M. J. Ehrhardt, P. Richtárik, and C.-B. Schonlieb. Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications. *SIAM Journal on Optimization*, 28(4):2783–2808, 2018.

[16] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

[17] C. D. Dang and G. Lan. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. *SIAM Journal on Optimization*, 25(2):856–881, 2015.

[4] In our DRO instance with a9a dataset, Gurobi presolver removed 25% of the columns and 58% of the nonzeros.

[18] D. P. De Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865, 2003.

[19] J. Douglas and H. H. Rachford. On the numerical solution of heat conduction problems in two and three space variables. *Transactions of the American Mathematical Society*, 82(2):421–439, 1956.

[20] J. Duchi and H. Namkoong. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018.

[21] J. C. Duchi, P. W. Glynn, and H. Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 2021.

[22] P. M. Esfahani and D. Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

[23] O. Fercoq and P. Bianchi. A coordinate-descent primal-dual algorithm with large step size and possibly nonseparable functions. *SIAM Journal on Optimization*, 29(1):100–134, 2019.

[24] X. Gao and S.-Z. Zhang. First-order algorithms for convex optimization with nonseparable objective and coupled constraints. *Journal of the Operations Research Society of China*, 5(2): 131–159, 2017.

[25] G. O. Gurobi. Reference manual, gurobi optimization, 2020.

[26] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2022. URL `https://www.gurobi.com`.

[27] E. Y. Hamedani and A. Jalilzadeh. A stochastic variance-reduced accelerated primal-dual method for finite-sum saddle-point problems. *arXiv preprint arXiv:2012.13456*, 2020.

[28] J. L. Hennessy and D. A. Patterson. *Computer architecture: a quantitative approach*. Elsevier, 2011.

[29] N. Ho-Nguyen and S. J. Wright. Adversarial classification via distributional robustness with Wasserstein ambiguity. *arXiv preprint arXiv:2005.13815*, November 2021.

[30] A. J. Hoffman. On approximate solutions of systems of linear inequalities. In *Selected Papers Of Alan J Hoffman: With Commentary*, pages 174–176. World Scientific, 2003.

[31] W. Hu, G. Niu, I. Sato, and M. Sugiyama. Does distributionally robust supervised learning give robust classifiers? In *Proc. ICML'18*, 2018.

[32] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Proc. NIPS'13*, 2013.

[33] A. Juditsky, A. Nemirovski, and C. Tauvel. Solving variational inequalities with stochastic mirror-prox algorithm. *Stochastic Systems*, 1(1):17–58, 2011.

[34] G. Korpelevich. The extragradient method for finding saddle points and other problems. *Ekonomika i Matematicheskie Metody*, 12(5):747–756 (in Russian; English translation in Matekon), 1976.

[35] P. Latafat, N. M. Freris, and P. Patrinos. A new randomized block-coordinate primal-dual proximal algorithm for distributed optimization. *IEEE Transactions on Automatic Control*, 64 (10):4050–4065, 2019.

[36] S. Lee and S. J. Wright. Manifold identification in dual averaging for regularized stochastic online learning. *Journal of Machine Learning Research*, 13(6), 2012.

[37] D. Levy, Y. Carmon, J. C. Duchi, and A. Sidford. Large-scale methods for distributionally robust optimization. In *Proc. NeurIPS'20*, 2020.

[38] J. Li, S. Huang, and A. M.-C. So. A first-order algorithmic framework for wasserstein distributionally robust logistic regression. *arXiv preprint arXiv:1910.12778*, 2019.

[39] C. Liu, T. Arnon, C. Lazarus, C. Strong, C. Barrett, M. J. Kochenderfer, et al. Algorithms for verifying deep neural networks. *Foundations and Trends® in Optimization*, 4, 2020.

[40] H. Lu and J. Yang. Nearly optimal linear convergence of stochastic primal-dual methods for linear programming. *arXiv preprint arXiv:2111.05530*, 2021.

[41] O. Mangasarian. A newton method for linear programming. *Journal of Optimization Theory and Applications*, 121(1):1–18, 2004.

[42] O. L. Mangasarian. Normal solutions of linear programs. In *Mathematical Programming at Oberwolfach II*, pages 206–216. Springer, 1984.

[43] O. L. Mangasarian and R. Meyer. Nonlinear perturbation of linear programs. *SIAM Journal on Control and Optimization*, 17(6):745–752, 1979.

[44] H. Namkoong and J. C. Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Proc. NIPS'16*, 2016.

[45] A. 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.

[46] Y. Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109(2-3):319–344, 2007.

[47] H. Ouyang, N. He, L. Tran, and A. Gray. Stochastic alternating direction method of multipliers. In *Proc. ICML'13*, 2013.

[48] Y. Ouyang and Y. Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, pages 1–35, 2019.

[49] A. Patrascu and I. Necoara. Nonasymptotic convergence of stochastic proximal point methods for constrained convex optimization. *Journal of Machine Learning Research*, 18(1):7204–7245, 2017.

[50] H. Rahimian and S. Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

[51] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.

[52] S. Shafieezadeh Abadeh, P. Mohajerin Esfahani, and D. Kuhn. Distributionally robust logistic regression. In *Proc. NIPS'15*, 2015.

[53] A. Shapiro. On duality theory of conic linear problems. In *Semi-infinite programming*, pages 135–165. Springer, 2001.

[54] C. Song, Y. Jiang, and Y. Ma. Variance reduction via accelerated dual averaging for finite-sum optimization. In *Proc. NeurIPS'20*, 2020.

[55] C. Song, S. J. Wright, and J. Diakonikolas. Variance reduction via primal-dual accelerated dual averaging for nonsmooth convex finite-sums. In *Proc. ICML'21*, 2021.

[56] J. Stoer. Duality in nonlinear programming and the minimax theorem. *Numerische Mathematik*, 5(1):371–379, 1963.

[57] C. Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[58] W. Wiesemann, D. Kuhn, and M. Sim. Distributionally robust convex optimization. *Operations Research*, 62(6):1358–1376, 2014.

[59] L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11(Oct):2543–2596, 2010.

[60] Y. Xu. Primal-dual stochastic gradient method for convex programs with many functional constraints. *SIAM Journal on Optimization*, 30(2):1664–1692, 2020.

[61] Y. Yu, T. Lin, E. Mazumdar, and M. I. Jordan. Fast distributionally robust learning with variance reduced min-max optimization. *arXiv preprint arXiv:2104.13326*, 2021.

[62] L. Zhao and D.-L. Zhu. On iteration complexity of a first-order primal-dual method for nonlinear convex cone programming. *Journal of the Operations Research Society of China*, 10(1):53–87, 2022.

[63] D. Zhu and L. Zhao. Linear convergence of randomized primal-dual coordinate method for large-scale linear constrained convex programming. In *International Conference on Machine Learning*, pages 11619–11628. PMLR, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] All theoretical claims are supported by proofs, which, due to space constraints, are provided in the appendix. Numerical results are provided in Section 5, with full technical details of the experiments provided in Appendix D.
   (b) Did you describe the limitations of your work? [Yes] The three main examples are: (i) the discussion of lazy update strategy and guarantees for $\hat{\boldsymbol{x}}_k$ versus $\tilde{\boldsymbol{x}}_k$ in Appendix A; (ii) the discussion of linear convergence under the restart strategy and theoretical guarantees for scheduled restart versus empirical guarantees for adaptive restart in Section 3.3; and (iii) comparison to off-the-shelf solvers in Section 5 and use of preconditioning.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] This work concerns a generic solver for generalized linear programs and as such is not tied to any specific application that could have a potential negative societal impact.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See assumptions in Section 2.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix B and C for complete proofs.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We include our code in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] In Section 5 and Appendix D.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Error bars are not reported due to almost identical behaviors and results from the large number of stochastic iterations.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] In Appendix D.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We used datasets from LibSVM, which is a open source machine learning library and cited the creators [16] in Appendix D which provides full experimental details.
   (b) Did you mention the license of the assets? [N/A] LibSVM datasets are distributed under the BSD license which is "a low restriction type of license for open source software that does not put requirements on redistribution."

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Our code is provided in the supplementary material.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] As mentioned before, the datasets we use are open source.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Coordinate Linear Variance Reduction for Generalized Linear Programming Appendix

Outline. The appendix of this paper is organized as follows:

- Section A contains details and the pseudo code for CLVR with lazy update.
- Section B provides the proofs for Section 3.
- Section C provides the proofs for Section 4.
- Section D provides the complementary details for Section 5.

A Lazy Update for Sparse and Structured Instances of (PD-GLP)

In Algorithm 1, for dense $\boldsymbol{A}$, the $O(|S^{j_k}|d)$ cost of Steps 6 and 8 dominates the $O(d)$ cost of Steps 4 and 9. However, when $\boldsymbol{A}$ is sparse and $|S^{j_k}|$ is small, the cost of Steps 6 and 8 will be $O(\mathrm{nnz}(\boldsymbol{A}^{S^{j_k}}))$, which may be less than the $O(d)$ cost of Steps 4 and 9. Using this observation, we show that the nature of the dual averaging update enables us to propose an efficient implementation whose complexity depends on $\mathrm{nnz}(\boldsymbol{A})$ rather than $n \cdot d$.

Recall that we partition $[n]$ into subsets $\{S^1, S^2, \ldots, S^m\}$ and use $\boldsymbol{A}^{S^j}$ $(j \in [m])$ to denote the jth row block of $\boldsymbol{A}$. For each block $\boldsymbol{A}^{S^j}$, we use $C^j \subset [d]$ to denote the indices of those columns of $\boldsymbol{A}^{S^j}$ that contain at least one nonzero element. (Of course, $\{C^1, \ldots, C^m\}$ is not in general a partition of $[d]$ as different subsets S^j may have nonzeros in the same columns.) We assume further that $\mathcal{X}$ and r are coordinate separable, that is, $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$ with $\mathcal{X}_i \subset \mathbb{R}$ for all i and $r(\boldsymbol{x}) := \sum_{i=1}^d r(\boldsymbol{x}^i)$ with $\boldsymbol{x}^i \in \mathcal{X}_i$.

In Step 4 of Algorithm 1, the separability of $\mathcal{X}$ and r means that an update to one coordinate block of $\boldsymbol{x}$ — say the C^{j_k} block in the update from $\boldsymbol{x}_{k-1}$ to $\boldsymbol{x}_k$, which requires a projection and an application of the proximal operator — does not influence other coordinates $\boldsymbol{x}_k^i$ for $i \notin C^{j_k}$. Moreover, we can efficiently maintain an implicit representation of $\boldsymbol{q}_k$, in terms of a newly introduced auxiliary vector $\boldsymbol{r}_k$; see Lemma 8 of Appendix B.2. Similarly, to update $\tilde{\boldsymbol{y}}_k$ efficiently, we maintain an implicit representation of $\tilde{\boldsymbol{y}}_k$ via an auxiliary vector $\boldsymbol{s}_k$, as shown in Lemma 9 of Appendix B.2.

It is generally not possible to maintain $\tilde{\boldsymbol{x}}_k$ efficiently since, in principle, all components of $\boldsymbol{x}_k$ can change on every iteration, and we wish to avoid the $O(d)$ cost of evaluating every full $\boldsymbol{x}_k$. We seek instead to output a proxy $\hat{\boldsymbol{x}}_K$ for $\tilde{\boldsymbol{x}}_K$ from Algorithm 1 such that $\mathbb{E}[\hat{\boldsymbol{x}}_K] = \tilde{\boldsymbol{x}}_K$. One possible choice is to pick an index $k' \in [K]$ from the weighted discrete distribution $(\frac{a_1}{A_K}, \frac{a_2}{A_K}, \ldots, \frac{a_K}{A_K})$ (computing the scalar quantities a_k and A_k for $k \in [K]$ in advance), then setting $\hat{\boldsymbol{x}}_K = \boldsymbol{x}_{k'}$. A slightly more sophisticated strategy is to sample a predetermined number $\widehat{K}$ of vectors $\boldsymbol{x}_k$, $k \in [K]$, and define $\hat{\boldsymbol{x}}_K$ to be the simple average of these vectors. Once again, the indices are chosen from the weighted discrete distribution $(\frac{a_1}{A_K}, \frac{a_2}{A_K}, \ldots, \frac{a_K}{A_K})$. Note that the total expected cost of the K iterations of the algorithm (excluding the full-vector updates) is $O(K\mathrm{nnz}(\boldsymbol{A})/m)$, while the total cost of evaluating the $\widehat{K}$ full vectors $\boldsymbol{x}_k$ and accumulating them into $\hat{\boldsymbol{x}}_K$ is $O(\widehat{K}d)$. Thus, for the full-step iterations not to dominate the total cost, we can choose $\widehat{K}$ to be $O(K\mathrm{nnz}(\boldsymbol{A})/(md))$. Finally, we note that Theorem 1 is for $\tilde{\boldsymbol{x}}_k$, not $\hat{\boldsymbol{x}}_K$. However, since $\mathbb{E}[\hat{\boldsymbol{x}}_k] = \tilde{\boldsymbol{x}}_k$, we expect the convergence rates from the theorem to hold for $\hat{\boldsymbol{x}}_K$ in practice.

An implementation of Algorithm 1 that exploits the form of $\boldsymbol{q}_k$ in Lemma 8 and $\tilde{\boldsymbol{y}}_k$ in Lemma 9, and evaluates explicitly only those components of $\boldsymbol{x}_k$ needed to perform the rest of the iteration is given as Algorithm 2. This version also incorporates the strategy for obtaining $\hat{\boldsymbol{x}}_K$ by sampling $\widehat{K}$ iterates on which to evaluate the full vector $\boldsymbol{x}_k$.

Due to the efficient implementation in Algorithm 2, to attain an ϵ-accurate solution in terms of the primal-dual gap in Theorem 1, we need $O(\frac{\mathrm{nnz}(\boldsymbol{A})\hat{L}}{\epsilon})$ FLOPS, which corresponds to $O(\frac{\hat{L}}{\epsilon})$ data passes. As a result, because a smaller batch size leads to a smaller $\hat{L}$, we attain the best performance in

terms of number of data passes when the batch size is set to one. However, as modern computer architecture has particular design for vectorized operations, lower runtime is obtained for a small batch size strictly larger than one (see Section 5).

Remark 1. *While we consider the case of fully coordinate separable r and $\mathcal{X}$ for simplicity, our lazy update approach is also applicable to the coordinate block partitioning case in which $\mathcal{X} = \mathcal{X}_1 \times \cdots \mathcal{X}_m$ with $\mathcal{X}_i \in \mathbb{R}^{d_i}$ ($i \in [m], \sum_{i=1}^m d_i = d$) and $r(\boldsymbol{x}) := \sum_{i=1}^m r(\boldsymbol{x}^i)$ with $\boldsymbol{x}^i \in \mathcal{X}_i$. The difference is that for each coordinate block $\boldsymbol{A}^{S^j}$ ($j \in [m]$), we overload $C^j \subset [m]$ to denote the set of blocks in $\boldsymbol{A}^{S^j}$ where each coordinate block contains at least one nonzero element.*

Remark 2. *Dual averaging has been shown to have significant advantage in producing sparser iterates than mirror descent in the context of online learning [36, 59]. It further leads to better bounds in well-conditioned finite-sum optimization [54]. In this work, we show that dual averaging offers better flexibility with sparse matrices than does mirror descent.*

Algorithm 2 Coordinate Linear Variance Reduction with Lazy Update (Lazy CLVR)

Input: $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_0 = \boldsymbol{x}_{-1} \in \mathcal{X}, \boldsymbol{y}_0 \in \mathbb{R}^n, \boldsymbol{z}_0 = \boldsymbol{A}^T\boldsymbol{y}_0$, $\gamma > 0$, $\hat{L} > 0$, K, $\widehat{K}$, m, $\{S^1, S^2, \ldots, S^m\}$, $\{C^1, C^2, \ldots, C^m\}$.
$a_0 = A_0 = 0, a_1 = A_1 = \frac{1}{2\hat{L}m}$, $\boldsymbol{q}_0 = a_1(\boldsymbol{z}_0 + \boldsymbol{c})$, $\boldsymbol{r}_0 = \boldsymbol{0}_d$, $\boldsymbol{s}_0 = \boldsymbol{0}_n$.
for $k = 1, 2, \ldots, K-1$ **do**
  $a_{k+1} = \frac{\sqrt{1+\sigma A_k/\gamma}}{2\hat{L}m}$, $A_{k+1} = A_k + a_{k+1}$.
end for
Choose indices $\{\ell_1, \ldots, \ell_{\widehat{K}}\}$ i.i.d. from $[K]$ according to the distribution $\left\{\frac{a_1}{A_K}, \frac{a_2}{A_K}, \ldots, \frac{a_K}{A_K}\right\}$.
for $k = 1, 2, \ldots, K$ **do**
  Pick j_k uniformly at random in $[m]$.
  if $k = \ell_i$ for some $i = 1, 2, \ldots, \widehat{K}$ **then**
    $\boldsymbol{q}_{k-1} = A_k(\boldsymbol{c} + \boldsymbol{z}_{k-1}) + \boldsymbol{r}_{k-1}$.
    $\boldsymbol{x}_k = \mathrm{prox}_{\frac{1}{\gamma}A_k r}(\boldsymbol{x}_0 - \frac{1}{\gamma}\boldsymbol{q}_{k-1})$.
  else
    $\boldsymbol{q}_{k-1}^{C^{j_k}} = A_k(\boldsymbol{c}^{C^{j_k}} + \boldsymbol{z}_{k-1}^{C^{j_k}}) + \boldsymbol{r}_{k-1}^{C^{j_k}}$.
    $\boldsymbol{x}_k^{C^{j_k}} = \mathrm{prox}_{\frac{1}{\gamma}A_k r}(\boldsymbol{x}_0^{C^{j_k}} - \frac{1}{\gamma}\boldsymbol{q}_{k-1}^{C^{j_k}})$.
  end if
  $\boldsymbol{y}_k^{S^{j_k}} = \boldsymbol{y}_{k-1}^{S^{j_k}} + \gamma m a_k\big(\boldsymbol{A}^{S^{j_k}, C^{j_k}}\boldsymbol{x}_k^{C^{j_k}} - \boldsymbol{b}^{S^{j_k}}\big), \boldsymbol{y}_{k-1}^{S^i}$ for all $i \neq j_k$.
  $\boldsymbol{z}_k^{C^{j_k}} = \boldsymbol{z}_{k-1}^{C^{j_k}} + (\boldsymbol{A}^{S^{j_k}, C^{j_k}})^T(\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}})$, $\boldsymbol{z}_k^i = \boldsymbol{z}_{k-1}^i$ for all $i \notin C^{j_k}$;
  $\boldsymbol{r}_k^{C^{j_k}} = \boldsymbol{r}_{k-1}^{C^{j_k}} + (ma_k - A_k)(\boldsymbol{z}_k^{C^{j_k}} - \boldsymbol{z}_{k-1}^{C^{j_k}})$, $\boldsymbol{r}_k^i = \boldsymbol{r}_{k-1}^i$ for all $i \notin C^{j_k}$;
  $\boldsymbol{s}_k^{S^{j_k}} = \boldsymbol{s}_{k-1}^{S^{j_k}} + ((m-1)a_k - A_{k-1})(\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}})$, $\boldsymbol{s}_k^i = \boldsymbol{s}_{k-1}^i$ for all $i \notin S^{j_k}$;
end for
$\hat{\boldsymbol{x}}_K = \frac{1}{\widehat{K}}\sum_{i=1}^{\widehat{K}} \boldsymbol{x}_{\ell_i}$.
$\tilde{\boldsymbol{y}}_K = \boldsymbol{y}_K + \frac{1}{A_K}\boldsymbol{s}_K$.
return $\hat{\boldsymbol{x}}_K$ and $\tilde{\boldsymbol{y}}_K$.

B Omitted Proofs from Section 3

B.1 Omitted Proofs from Section 3.1

We state a version of CLVR in Algorithm 3 that is convenient for the analysis, and is equivalent to Algorithm 1 in the main body of the paper. Before analyzing the convergence of CLVR, we justify our claim of equivalence in Proposition 1.

Proposition 1. *The iterates of Algorithm 1 and 3 are equivalent.*

Proof. To argue equivalence, we show that the iterates of Algorithm 1 and 3 solve the same optimization problems. To avoid ambiguity, here we will use $\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{y}}_k$ to denote the iterates $\boldsymbol{x}_k, \boldsymbol{y}_k$ in Algorithm 3, while we retain the notation $\boldsymbol{x}_k, \boldsymbol{y}_k$ for the iterates of Algorithm 1.

Let us first start by writing an equivalent definition of $\boldsymbol{x}_k$ in Algorithm 1. To do so, we first unroll the recursive definitions of $\boldsymbol{z}_k$ and $\boldsymbol{q}_k$. We can observe that, since by definition $\boldsymbol{y}_k$ and $\boldsymbol{y}_{k-1}$ only differ over the coordinate block S^{j_k}, we have

$$\boldsymbol{z}_k = \boldsymbol{z}_0 + \sum_{i=1}^{k} \boldsymbol{A}^T(\boldsymbol{y}_i - \boldsymbol{y}_{i-1}) = \boldsymbol{A}^T \boldsymbol{y}_k. \tag{12}$$

On the other hand, using Eq. (12), the definition of $\boldsymbol{q}_k$ implies

$$\boldsymbol{q}_k = A_{k+1}\boldsymbol{c} + a_1 \boldsymbol{A}^T \boldsymbol{y}_0 + \sum_{i=1}^{k} \boldsymbol{A}^T \left[a_{i+1}\boldsymbol{y}_i + m a_i(\boldsymbol{y}_i - \boldsymbol{y}_{i-1})\right] \tag{13}$$

Using the definition of the proximal operator (see Eq. (1)) and the definition of $\boldsymbol{x}_k$ in Step 4 of Algorithm 1, we have

$$\begin{aligned} \boldsymbol{x}_k &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \left\{ \frac{A_k}{\gamma} r(\boldsymbol{x}) + \frac{1}{2}\left\| \boldsymbol{x} - \boldsymbol{x}_0 + \frac{1}{\gamma}\boldsymbol{q}_{k-1} \right\|^2 \right\} \\ &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \left\{ A_k r(\boldsymbol{x}) + \frac{\gamma}{2}\left\| \boldsymbol{x} - \boldsymbol{x}_0 + \frac{1}{\gamma}\boldsymbol{q}_{k-1} \right\|^2 \right\}. \end{aligned} \tag{14}$$

Now let us consider the optimization problem that defines $\hat{\boldsymbol{x}}_k$. Assume for now that the definitions of $\boldsymbol{y}_k$ and $\hat{\boldsymbol{y}}_k$ agree (we justify this claim below). Observe first that the minimization problem defining $\hat{\boldsymbol{x}}_k$ is independent of $\boldsymbol{u}$, so by unrolling the recursion for ϕ_k, we have

$$\begin{aligned} \hat{\boldsymbol{x}}_k &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \Bigg\{ \frac{\gamma}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2 + A_k r(\boldsymbol{x}) + a_1 \left\langle \boldsymbol{x}, \boldsymbol{c} + \boldsymbol{A}^T \bar{\boldsymbol{y}}_0 \right\rangle \\ &\qquad + \sum_{i=2}^{k} a_i \left\langle \boldsymbol{x}, \boldsymbol{c} + \boldsymbol{A}^T \Big(\boldsymbol{y}_{i-1} + \frac{m a_{i-1}}{a_i}(\boldsymbol{y}_{i-1} - \boldsymbol{y}_{i-2}) \Big) \right\rangle \Bigg\} \\ &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \Bigg\{ \frac{\gamma}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2 + A_k r(\boldsymbol{x}) \\ &\qquad + \left\langle \boldsymbol{x}, A_k \boldsymbol{c} + a_1 \boldsymbol{A}^T \boldsymbol{y}_0 + \boldsymbol{A}^T \Big(\sum_{i=2}^{k} \left[a_i \boldsymbol{y}_{i-1} + m a_{i-1}(\boldsymbol{y}_{i-1} - \boldsymbol{y}_{i-2}) \right] \Big) \right\rangle \Bigg\} \\ &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \left\{ \frac{\gamma}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2 + A_k r(\boldsymbol{x}) + \langle \boldsymbol{x}, \boldsymbol{q}_{k-1} \rangle \right\} \\ &= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \left\{ A_k r(\boldsymbol{x}) + \frac{\gamma}{2}\left\| \boldsymbol{x} - \boldsymbol{x}_0 + \frac{1}{\gamma}\boldsymbol{q}_{k-1} \right\|^2 \right\} \\ &= \boldsymbol{x}_k. \end{aligned}$$

It remains to argue that the definitions of $\boldsymbol{y}_k$ and $\hat{\boldsymbol{y}}_k$ agree. First, observe that since the definitions of ψ_k and ψ_{k-1} differ only over block S^{j_k}, we have $\hat{\boldsymbol{y}}_k^{S^j} = \hat{\boldsymbol{y}}_{k-1}^{S^j}$ for all $j \neq j_k$. For $j = j_k$, we have by unrolling the recursive definition of ψ_k that

$$\begin{aligned} \hat{\boldsymbol{y}}_k^{S^{j_k}} &= \arg\min_{\boldsymbol{y}^{S^{j_k}} \in \mathbb{R}^{n_{j_k}}} \left\{ \frac{1}{2\gamma}\|\boldsymbol{y}^{S^{j_k}} - \boldsymbol{y}_0^{S^{j_k}}\|^2 - \sum_{i=1}^{k} \mathbb{1}_{\{S^{j_i} = S^{j_k}\}} m a_i \left\langle \boldsymbol{y}^{S^{j_i}}, \boldsymbol{A}^{S^{j_i}} \boldsymbol{x}_i - \boldsymbol{b}^{S^{j_i}} \right\rangle \right\} \\ &= \boldsymbol{y}_0^{S^{j_k}} + \gamma \sum_{i=1}^{k} \mathbb{1}_{\{S^{j_i} = S^{j_k}\}} m a_i (\boldsymbol{A}^{S^{j_i}} \boldsymbol{x}_i - \boldsymbol{b}^{S^{j_i}}) \\ &= \hat{\boldsymbol{y}}_{k-1}^{S^{j_k}} + \gamma m a_k (\boldsymbol{A}^{S^{j_k}} \boldsymbol{x}_k - \boldsymbol{b}^{S^{j_k}}) \\ &= \boldsymbol{y}_k^{S^{j_k}}, \end{aligned}$$

as claimed. □

In the following three lemmas, we bound $\phi_k(\boldsymbol{x}_k)$ and $\psi_k(\boldsymbol{y}_k)$ below and above, which is then subsequently used to bound the primal-dual gap in Theorem 1.

Algorithm 3 Coordinate Linear Variance Reduction (Analysis Version)

Input: $\boldsymbol{x}_0 = \boldsymbol{x}_{-1} \in \mathcal{X}, \boldsymbol{y}_0 = \bar{\boldsymbol{y}}_0 \in \mathbb{R}^n, m, \{S^1, S^2, \ldots, S^m\}, K, \gamma > 0, \hat{L} > 0.$
$\phi_0(\cdot) = \frac{\gamma}{2}\|\cdot - \boldsymbol{x}_0\|^2, \psi_0(\cdot) = \frac{1}{2\gamma}\|\cdot - \boldsymbol{y}_0\|^2.$
$a_1 = A_1 = \frac{1}{2\hat{L}m}.$
for $k = 1, 2, 3, \ldots, K$ **do**
  $\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{X}}\{\phi_k(\boldsymbol{x}) = \phi_{k-1}(\boldsymbol{x}) + a_k(\langle \boldsymbol{x} - \boldsymbol{u}, \boldsymbol{A}^T\bar{\boldsymbol{y}}_{k-1} + \boldsymbol{c}\rangle + r(\boldsymbol{x}) - r(\boldsymbol{u}))\}.$
  Pick j_k uniformly at random in $[m]$.
  $\boldsymbol{y}_k = \arg\min_{\boldsymbol{y}\in\mathbb{R}^n}\{\psi_k(\boldsymbol{y}) = \psi_{k-1}(\boldsymbol{y}) + ma_k(-\langle \boldsymbol{y}^{S^{j_k}} - \boldsymbol{v}^{S^{j_k}}, \boldsymbol{A}^{S^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{S^{j_k}}\rangle)\}.$
  $a_{k+1} = \frac{\sqrt{1+\sigma A_k/\gamma}}{2\hat{L}m}, A_{k+1} = A_k + a_{k+1}.$
  $\bar{\boldsymbol{y}}_k = \boldsymbol{y}_k + \frac{ma_k}{a_{k+1}}(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}).$
end for
return $\tilde{\boldsymbol{x}}_K := \frac{1}{A_K}\sum_{k=1}^K a_k\boldsymbol{x}_k, \tilde{\boldsymbol{y}}_K = \frac{1}{A_K}\sum_{k=1}^K(a_k\boldsymbol{y}_k + (m-1)a_k(\boldsymbol{y}_k - \boldsymbol{y}_{k-1})).$

Lemma 1. *For all steps of Algorithm 3 with $k \geq 1$, we have, $\forall(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{X} \times \mathbb{R}^n$,*

$$\begin{aligned}\phi_k(\boldsymbol{x}_k) &\leq \frac{\gamma}{2}\|\boldsymbol{u} - \boldsymbol{x}_0\|^2 - \frac{\gamma + \sigma A_k}{2}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2,\\ \psi_k(\boldsymbol{y}_k) &\leq \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_0\|^2 - \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_k\|^2.\end{aligned}$$

Proof. By the definitions of $\psi_k(\boldsymbol{y})$ and $\phi_k(\boldsymbol{x})$ in Algorithm 3, it follows that, $\forall k \geq 1$,

$$\phi_k(\boldsymbol{x}) = \sum_{i=1}^k a_i(\langle \boldsymbol{x} - \boldsymbol{u}, \boldsymbol{A}^T\bar{\boldsymbol{y}}_{i-1} + \boldsymbol{c}\rangle + r(\boldsymbol{x}) - r(\boldsymbol{u})) + \frac{\gamma}{2}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2, \tag{15}$$

and

$$\psi_k(\boldsymbol{y}) = \sum_{i=1}^k ma_i\Big(-\Big\langle \boldsymbol{y}^{S^{j_i}} - \boldsymbol{v}^{S^{j_i}}, \boldsymbol{A}^{S^{j_i}}\boldsymbol{x}_i - \boldsymbol{b}^{S^{j_i}}\Big\rangle\Big) + \frac{1}{2\gamma}\|\boldsymbol{y} - \boldsymbol{y}_0\|^2. \tag{16}$$

Observe that, as function of $\boldsymbol{x}$, $\phi_k(\boldsymbol{x})$ is $(\gamma + \sigma A_k)$-strongly convex. As, by definition, $\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}\in\mathcal{X}}\phi_k(\boldsymbol{x})$, it follows that

$$\phi_k(\boldsymbol{u}) \geq \phi_k(\boldsymbol{x}_k) + \frac{\gamma + \sigma A_k}{2}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2. \tag{17}$$

Now, writing $\phi_k(\boldsymbol{u})$ explicitly and rearranging the last inequality, the stated bound on $\phi_k(\boldsymbol{x}_k)$ follows.

As a function of $\boldsymbol{y}$, $\psi_k(\boldsymbol{y})$ is $1/\gamma$-strongly convex. The proof for the bound on ψ_k uses the same argument and is omitted. □

In the following proof, for $k \geq 1$, let $\mathcal{F}_k$ denote the natural filtration, containing all the randomness in the algorithm up to and including iteration k. Recall that $\boldsymbol{A} = \begin{pmatrix}\boldsymbol{A}^{S^1}\\ \vdots \\ \boldsymbol{A}^{S^m}\end{pmatrix}$, and let $\boldsymbol{A}^{\bar{S}^j}(j \in [m])$ denote the matrix $\boldsymbol{A}$ with its S^j block of rows replaced by a zero block.

For convenience, for $k = 1, 2, \ldots,$ we define

$$\hat{\boldsymbol{y}}_k = \boldsymbol{y}_{k-1} + \gamma ma_k(\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}). \tag{18}$$

Then from the definition of $\boldsymbol{y}_k$ in Algorithm 1, we have $\mathbb{E}[\boldsymbol{y}_k - \boldsymbol{y}_{k-1}|\mathcal{F}_{k-1}] = \frac{1}{m}(\hat{\boldsymbol{y}}_k - \boldsymbol{y}_{k-1})$.

Motivated by [4], we have the following result.

Lemma 2. *Given the sequences $\{\boldsymbol{y}_k\}$ in Algorithm 3 and $\{\hat{\boldsymbol{y}}_k\}$ in Eq.* (18)*, we define the sequence $\{\check{\boldsymbol{v}}_k\}$ by*

$$\check{\boldsymbol{v}}_k = (\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) - \frac{1}{m}(\hat{\boldsymbol{y}}_k - \boldsymbol{y}_{k-1}).$$

Then for any $\boldsymbol{v} \in \mathbb{R}^d$ that may be correlated with the randomness in $\{\check{\boldsymbol{v}}_i\}_{i=1}^k$, we have

$$\mathbb{E}\Big[-\sum_{i=1}^k \langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle\Big] \le \mathbb{E}\Big[\frac{1}{2}\|\boldsymbol{y}_0 - \boldsymbol{v}\|^2 + \frac{1}{2}\sum_{i=1}^k \|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big], \tag{19}$$

where the expectation is taken over all the randomness in the history.

Proof. First, we prove a bound for $\mathbb{E}\Big[\sum_{i=1}^k \|\check{\boldsymbol{v}}_i\|^2\Big]$. By the fact that $\mathbb{E}[\boldsymbol{y}_i - \boldsymbol{y}_{i-1}|\mathcal{F}_{i-1}] = \frac{1}{m}(\hat{\boldsymbol{y}}_i - \boldsymbol{y}_{i-1})$, for each $\check{\boldsymbol{v}}_i$, we have

$$\mathbb{E}[\|\check{\boldsymbol{v}}_i\|^2|\mathcal{F}_{i-1}] = \mathbb{E}\Big[\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2|\mathcal{F}_{i-1}\Big] - \mathbb{E}\Big[\big\|\frac{1}{m}\big(\hat{\boldsymbol{y}}_i - \boldsymbol{y}_{i-1}\big)\big\|^2|\mathcal{F}_{i-1}\Big] \le \mathbb{E}[\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2|\mathcal{F}_{i-1}].$$

Taking expectation on all the randomness in the history, for $\mathbb{E}\Big[\sum_{i=1}^k \|\check{\boldsymbol{v}}_i\|^2\Big]$, we have

$$\mathbb{E}\Big[\sum_{i=1}^k \|\check{\boldsymbol{v}}_i\|^2\Big] = \mathbb{E}\Big[\sum_{i=1}^k \mathbb{E}[\|\check{\boldsymbol{v}}_i\|^2|\mathcal{F}_{i-1}]\Big] \le \mathbb{E}\Big[\sum_{i=1}^k \mathbb{E}[\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2|\mathcal{F}_{i-1}]\Big] = \mathbb{E}\Big[\sum_{i=1}^k \|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big]. \tag{20}$$

Then to prove (19), we define the sequence $\{\check{\boldsymbol{y}}_k\}_{k=0}^\infty$ by $\check{\boldsymbol{y}}_0 = \boldsymbol{y}_0$ and $\check{\boldsymbol{y}}_k = \check{\boldsymbol{y}}_{k-1} - \check{\boldsymbol{v}}_k$ for $k = 1, 2, \ldots$. By expanding $\frac{1}{2}\|\check{\boldsymbol{y}}_k - \boldsymbol{v}\|^2$,

$$\begin{aligned} \frac{1}{2}\|\check{\boldsymbol{y}}_k - \boldsymbol{v}\|^2 &= \frac{1}{2}\|\check{\boldsymbol{y}}_{k-1} - \boldsymbol{v}\|^2 - \langle \check{\boldsymbol{v}}_k, \check{\boldsymbol{y}}_{k-1} - \boldsymbol{v}\rangle + \frac{1}{2}\|\check{\boldsymbol{v}}_k\|^2 \\ \implies \quad \langle \check{\boldsymbol{v}}_k, \check{\boldsymbol{y}}_{k-1} - \boldsymbol{v}\rangle &= \frac{1}{2}\|\check{\boldsymbol{y}}_{k-1} - \frac{1}{2}\|\check{\boldsymbol{y}}_k - \boldsymbol{v}\|^2 + \frac{1}{2}\|\check{\boldsymbol{v}}_k\|^2. \end{aligned}$$

By summing this expression and telescoping, we obtain

$$\begin{aligned} \sum_{i=1}^k \langle \check{\boldsymbol{y}}_{i-1} - \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle &= \frac{1}{2}\|\check{\boldsymbol{y}}_0 - \boldsymbol{v}\|^2 - \frac{1}{2}\|\check{\boldsymbol{y}}_k - \boldsymbol{v}\|^2 + \sum_{i=1}^k \frac{1}{2}\|\check{\boldsymbol{v}}_i\|^2 \\ &\le \frac{1}{2}\|\check{\boldsymbol{y}}_0 - \boldsymbol{v}\|^2 + \sum_{i=1}^k \frac{1}{2}\|\check{\boldsymbol{v}}_i\|^2 \\ &= \frac{1}{2}\|\boldsymbol{y}_0 - \boldsymbol{v}\|^2 + \sum_{i=1}^k \frac{1}{2}\|\check{\boldsymbol{v}}_i\|^2, \end{aligned} \tag{21}$$

It follows that

$$\begin{aligned} \mathbb{E}\Big[-\sum_{i=1}^k \langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle\Big] &= \mathbb{E}\Big[\sum_{i=1}^k \langle \check{\boldsymbol{y}}_{i-1} - \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle - \sum_{i=1}^k \langle \check{\boldsymbol{y}}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big] \\ &\le \mathbb{E}\Big[\frac{1}{2}\|\boldsymbol{y}_0 - \boldsymbol{v}\|^2 + \sum_{i=1}^k \frac{1}{2}\|\check{\boldsymbol{v}}_i\|^2 - \sum_{i=1}^k \langle \check{\boldsymbol{y}}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big] \\ &\le \mathbb{E}\Big[\frac{1}{2}\|\boldsymbol{y}_0 - \boldsymbol{v}\|^2 + \frac{1}{2}\sum_{i=1}^k \|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 - \sum_{i=1}^k \langle \check{\boldsymbol{y}}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big]. \end{aligned} \tag{22}$$

where the first inequality is by Eq. (21) and the second inequality is by Eq. (20). To obtain the result (19), we use the facts that $\mathbb{E}[\check{\boldsymbol{v}}_i \mid \mathcal{F}_{i-1}] = 0$ and that $\check{\boldsymbol{y}}_{i-1} \in \mathcal{F}_{i-1}$ to obtain $\mathbb{E}\Big[\sum_{i=1}^k \langle \check{\boldsymbol{y}}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big] = 0$. □

We emphasize that Lemma 2 holds for any $\boldsymbol{v}$ that may be even correlated with the randomness in the algorithm.

Lemma 3. *For all steps of Algorithm 3 with $k \geq 1$, we have for all $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{X} \times \mathbb{R}^n$ that*

$$\begin{aligned}
\phi_k(\boldsymbol{x}_k) \geq \phi_{k-1}(\boldsymbol{x}_{k-1}) &+ \frac{\gamma + \sigma A_{k-1}}{2}\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|^2 + a_k(r(\boldsymbol{x}_k) - r(\boldsymbol{u})) \\
&- a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) \right\rangle + a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T \boldsymbol{y}_k + \boldsymbol{c} \right\rangle \\
&+ m a_{k-1} \Big(\left\langle \boldsymbol{x}_{k-1} - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle + \left\langle \boldsymbol{x}_k - \boldsymbol{x}_{k-1}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle \Big), \\
\psi_k(\boldsymbol{y}_k) \geq \psi_{k-1}(\boldsymbol{y}_{k-1}) &+ \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 - a_k \langle \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}, \boldsymbol{y}_k - \boldsymbol{v} \rangle \\
&- (m-1) a_k \langle \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}, \boldsymbol{y}_k - \boldsymbol{y}_{k-1} \rangle + \frac{1}{\gamma} \left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, -(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) + \frac{1}{m}(\hat{\boldsymbol{y}}_k - \boldsymbol{y}_{k-1}) \right\rangle.
\end{aligned}$$

Proof. For the first claim, we have from the definition of $\phi_k(\boldsymbol{x}_k)$, using that $\phi_{k-1}(\boldsymbol{x}_{k-1})$ is $(\gamma + \sigma A_{k-1})$-strongly convex and minimized at $\boldsymbol{x}_{k-1}$, that

$$\begin{aligned}
\phi_k(\boldsymbol{x}_k) \geq \phi_{k-1}(\boldsymbol{x}_{k-1}) &+ \frac{\gamma + \sigma A_{k-1}}{2}\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|^2 \\
&+ a_k \big(\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T \bar{\boldsymbol{y}}_{k-1} + \boldsymbol{c} \rangle + r(\boldsymbol{x}_k) - r(\boldsymbol{u}) \big).
\end{aligned} \tag{23}$$

Meanwhile, by the definition of $\{\bar{\boldsymbol{y}}_k\}$ (using $\boldsymbol{y}_{-1} = \boldsymbol{y}_0$ for the case of $k = 1$), we have that

$$\begin{aligned}
& a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T \bar{\boldsymbol{y}}_{k-1} \right\rangle \\
= \; & a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T \Big(\boldsymbol{y}_{k-1} + \frac{m a_{k-1}}{a_k}(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \Big) \right\rangle \\
= \; & - a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) \right\rangle + a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T \boldsymbol{y}_k \right\rangle \\
& + m a_{k-1} \Big(\left\langle \boldsymbol{x}_{k-1} - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle + \left\langle \boldsymbol{x}_k - \boldsymbol{x}_{k-1}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle \Big).
\end{aligned} \tag{24}$$

The claimed lower bound on $\phi_k(\boldsymbol{x}_k)$ follows when we combine (23) and (24).

For the second claim, we have by the definition of ψ_k that

$$\psi_k(\boldsymbol{y}_k) - \psi_{k-1}(\boldsymbol{y}_{k-1}) = \psi_{k-1}(\boldsymbol{y}_k) - \psi_{k-1}(\boldsymbol{y}_{k-1}) - m a_k \left\langle \boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{v}^{S^{j_k}}, \boldsymbol{A}^{S^{j_k}} \boldsymbol{x}_k - \boldsymbol{b}^{S^{j_k}} \right\rangle. \tag{25}$$

To obtain the claimed lower bound on ψ_k, we proceed to bound the terms on the right-hand side in (25). First, since ψ_{k-1} is $(1/\gamma)$-strongly convex and minimized at $\boldsymbol{y}_{k-1}$, we have

$$\psi_{k-1}(\boldsymbol{y}_k) - \psi_{k-1}(\boldsymbol{y}_{k-1}) \geq \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2. \tag{26}$$

Second, by using the definition of $\boldsymbol{A}^{S_{j_k}}$ and $\boldsymbol{A}^{\bar{S}_{j_k}}$, and by using several times that $\boldsymbol{y}_{k-1}$ and $\boldsymbol{y}_k$ differ only in their S^{j_k} components, we have

$$
\begin{aligned}
&-\left\langle \boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{v}^{S^{j_k}}, \boldsymbol{A}^{S^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{S^{j_k}} \right\rangle \\
&\quad = -\left\langle \boldsymbol{y}_k - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle + \left\langle \boldsymbol{y}_k - \boldsymbol{v}, \boldsymbol{A}^{\bar{S}^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{\bar{S}^{j_k}} \right\rangle \\
&\quad = -\left\langle \boldsymbol{y}_k - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle + \left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, \boldsymbol{A}^{\bar{S}^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{\bar{S}^{j_k}} \right\rangle \\
&\quad = -\left\langle \boldsymbol{y}_k - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle + \frac{m-1}{m}\left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle \\
&\qquad + \left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, \boldsymbol{A}^{\bar{S}^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{\bar{S}^{j_k}} - \frac{m-1}{m}(\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}) \right\rangle \\
&\quad = -\frac{1}{m}\left\langle \boldsymbol{y}_k - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle + \frac{m-1}{m}\left\langle \boldsymbol{y}_{k-1} - \boldsymbol{y}_k, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle \\
&\qquad + \left\langle \boldsymbol{y}_{k-1}^{S^{j_k}} - \boldsymbol{v}^{S^{j_k}}, -(\boldsymbol{A}^{S^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{S^{j_k}}) \right\rangle + \frac{1}{m}\left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b}) \right\rangle \\
&\quad = -\frac{1}{m}\left\langle \boldsymbol{y}_k - \boldsymbol{v}, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle + \frac{m-1}{m}\left\langle \boldsymbol{y}_{k-1} - \boldsymbol{y}_k, \boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b} \right\rangle \\
&\qquad + \left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, -\frac{1}{\gamma m a_k}(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) + \frac{1}{\gamma m^2 a_k}(\hat{\boldsymbol{y}}_k - \boldsymbol{y}_{k-1}) \right\rangle,
\end{aligned} \tag{27}
$$

where in the last equality we used $\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}} = \gamma m a_k(\boldsymbol{A}^{S^{j_k}}\boldsymbol{x}_k - \boldsymbol{b}^{S^{j_k}})$ and $\hat{\boldsymbol{y}}_k - \boldsymbol{y}_{k-1} = \gamma m a_k(\boldsymbol{A}\boldsymbol{x}_k - \boldsymbol{b})$, which both hold by definitions of $\hat{\boldsymbol{y}}_k$ and $\boldsymbol{y}_k$.

Finally, combining (25)–(27), we have the bound on $\psi_k(\boldsymbol{y}_k)$ from the statement of the lemma. □

By combining the two lower bounds in Lemma 3, we obtain the following result.

Lemma 4. *For any $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{X} \times \mathbb{R}^n$, the sum of $\psi_k(\boldsymbol{y}_k) + \phi_k(\boldsymbol{x}_k)$ can be bounded as follows: for all $k \geq 1$,*

$$
\begin{aligned}
&\phi_k(\boldsymbol{x}_k) + \psi_k(\boldsymbol{y}_k) \\
&\geq \phi_{k-1}(\boldsymbol{x}_{k-1}) + \psi_{k-1}(\boldsymbol{y}_{k-1}) \\
&\qquad - m a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) \right\rangle + m a_{k-1}\left\langle \boldsymbol{x}_{k-1} - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle \\
&\qquad + \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 - \frac{1}{4\gamma}\|\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}\|^2 + Q_k,
\end{aligned} \tag{28}
$$

where

$$
\begin{aligned}
Q_k :=& a_k\Big(r(\boldsymbol{x}_k) - r(\boldsymbol{u}) - \langle \boldsymbol{A}\boldsymbol{u}, \boldsymbol{y}_k \rangle + \langle \boldsymbol{x}_k, \boldsymbol{A}^T\boldsymbol{v} \rangle + \langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{c} \rangle + \langle \boldsymbol{b}, \boldsymbol{y}_k - \boldsymbol{v} \rangle \\
&- (m-1)\langle \boldsymbol{A}\boldsymbol{u} - \boldsymbol{b}, \boldsymbol{y}_k - \boldsymbol{y}_{k-1} \rangle\Big) + \frac{1}{\gamma}\left\langle \boldsymbol{y}_{k-1} - \boldsymbol{v}, -(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) + \frac{1}{m}(\hat{\boldsymbol{y}}_k - \boldsymbol{y}_{k-1}) \right\rangle.
\end{aligned} \tag{29}
$$

Proof. Before our proof, as A_{-1} and A_0 are not used in Algorithm 3, without loss of generality, we set $A_{-1} = A_0 = 0$. Fix any $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{X} \times \mathbb{R}^n$. By combining the bounds on $\phi_k(\boldsymbol{x}_k)$ and $\psi_k(\boldsymbol{y}_k)$ from Lemma 3, we have $\forall k \geq 1$ that

$$
\begin{aligned}
&\phi_k(\boldsymbol{x}_k) + \psi_k(\boldsymbol{y}_k) \\
&\geq \phi_{k-1}(\boldsymbol{x}_{k-1}) + \psi_{k-1}(\boldsymbol{y}_{k-1}) \\
&\qquad - m a_k \left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}) \right\rangle + m a_{k-1}\left\langle \boldsymbol{x}_{k-1} - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle \\
&\qquad + P_k + Q_k,
\end{aligned} \tag{30}
$$

where

$$
P_k = \frac{\gamma + \sigma A_{k-1}}{2}\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 + m a_{k-1}\left\langle \boldsymbol{x}_k - \boldsymbol{x}_{k-1}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}) \right\rangle \tag{31}
$$

and Q_k is defined in Eq. (29).

To find a lower bound on P_k we start by bounding the magnitude of the inner product term in (31). Recall that $\boldsymbol{y}_{k-2}$ and $\boldsymbol{y}_{k-1}$ differ only on the coordinate block $S^{j_{k-1}}$. We thus have:

$$\begin{aligned}
&|ma_{k-1}\langle \boldsymbol{x}_k - \boldsymbol{x}_{k-1}, \boldsymbol{A}^T(\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2})\rangle| \\
&= |ma_{k-1}\left\langle \boldsymbol{A}^{S^{j_{k-1}}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}), \boldsymbol{y}_{k-1}^{S^{j_{k-1}}} - \boldsymbol{y}_{k-2}^{S^{j_{k-1}}}\right\rangle| \\
&\leq ma_{k-1}\|\boldsymbol{A}^{S^{j_{k-1}}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})\|\|\boldsymbol{y}_{k-1}^{S^{j_{k-1}}} - \boldsymbol{y}_{k-2}^{S^{j_{k-1}}}\| \\
&\leq (ma_{k-1})^2\gamma\|\boldsymbol{A}^{S^{j_{k-1}}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_{k-1}^{S^{j_{k-1}}} - \boldsymbol{y}_{k-2}^{S^{j_{k-1}}}\|^2 \\
&\leq (m\hat{L}a_{k-1})^2\gamma\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_{k-1}^{S^{j_{k-1}}} - \boldsymbol{y}_{k-2}^{S^{j_{k-1}}}\|^2 \\
&= (m\hat{L}a_{k-1})^2\gamma\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}\|^2 \qquad (32)
\end{aligned}$$

where the second inequality is by Young's inequality, and the third inequality is by Assumption 3 where $\|\boldsymbol{A}^{S^{j_{k-1}}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})\| \leq \hat{L}\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|$. With our setting $a_{k-1} = \frac{\sqrt{1+\sigma A_{k-2}/\gamma}}{2m\hat{L}} \leq \frac{\sqrt{1+\sigma A_{k-1}/\gamma}}{2m\hat{L}}$, for all $k \geq 1$, we have

$$(m\hat{L}a_{k-1})^2\gamma = \frac{\gamma + \sigma A_{k-1}}{4}.$$

By substituting this equality into (32) and then combining with (31), we obtain

$$\begin{aligned}
P_k &\geq \frac{\gamma + \sigma A_{k-1}}{4}\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 - \frac{1}{4\gamma}\|\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}\|^2 \\
&\geq \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 - \frac{1}{4\gamma}\|\boldsymbol{y}_{k-1} - \boldsymbol{y}_{k-2}\|^2. \qquad (33)
\end{aligned}$$

We complete the proof by combining Eqs. (30), (31) and the bound Eq. (33) for P_k. □

By telescoping the inequality in Lemma 4, and using Lemmas 1 and 2, we obtain the next result.

Lemma 5. *For all $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{X} \times \mathbb{R}^n$, we have*

$$\begin{aligned}
&A_k(\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)) + \frac{\gamma + \sigma A_k}{4}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2 + \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_k\|^2 \\
&\leq \frac{\gamma}{2}\|\boldsymbol{u} - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_0\|^2 - \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 + \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle - \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle, \qquad (34)
\end{aligned}$$

where $\check{\boldsymbol{v}}_i$ is defined in Lemma 2.

Proof. Telescoping the inequality in Lemma 4, we have

$$\begin{aligned}
\phi_k(\boldsymbol{x}_k) + \psi_k(\boldsymbol{y}_k) \geq\ & \phi_0(\boldsymbol{x}_0) + \psi_0(\boldsymbol{y}_0) \\
& - ma_k\left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1})\right\rangle + ma_0\left\langle \boldsymbol{x}_0 - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_0 - \boldsymbol{y}_{-1})\right\rangle \\
& + \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 - \frac{1}{4\gamma}\|\boldsymbol{y}_0 - \boldsymbol{y}_{-1}\|^2 + \frac{1}{4\gamma}\sum_{i=1}^{k-1}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 + \sum_{i=1}^{k} Q_i \\
=\ & - ma_k\left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1})\right\rangle + \frac{1}{4\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 \\
& + \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 + \sum_{i=1}^{k} Q_i, \qquad (35)
\end{aligned}$$

where the last equality is by the fact that $\psi_0(\boldsymbol{y}_0) = \phi_0(\boldsymbol{x}_0) = 0$, $a_0 = 0$, and our convention that $\boldsymbol{y}_{-1} := \boldsymbol{y}_0$.

Then by the convexity of $r(\cdot)$ and the definition of $\{Q_i\}$, we have

$$\begin{aligned}\sum_{i=1}^{k} Q_i &\geq A_k(r(\tilde{\boldsymbol{x}}_k) - r(\boldsymbol{u}) - \langle \boldsymbol{A}\boldsymbol{u}, \tilde{\boldsymbol{y}}_k\rangle + \langle \tilde{\boldsymbol{x}}_k, \boldsymbol{A}^T\boldsymbol{v}\rangle + \langle \tilde{\boldsymbol{x}}_k - \boldsymbol{u}, \boldsymbol{c}\rangle + \langle \boldsymbol{b}, \tilde{\boldsymbol{y}}_k - \boldsymbol{v}\rangle) \\ &\quad + \frac{1}{\gamma}\sum_{i=1}^{k}\left\langle \boldsymbol{y}_{i-1} - \boldsymbol{v}, -(\boldsymbol{y}_i - \boldsymbol{y}_{i-1}) + \frac{1}{m}(\hat{\boldsymbol{y}}_i - \boldsymbol{y}_{i-1})\right\rangle \\ &= A_k(\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)) + \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1} - \boldsymbol{v}, -\check{\boldsymbol{v}}_i\rangle\,. \end{aligned} \tag{36}$$

where $\tilde{\boldsymbol{x}}_k = \frac{1}{A_k}\sum_{i=1}^{k} a_i\boldsymbol{x}_i$, $\tilde{\boldsymbol{y}}_k = \frac{1}{A_k}\sum_{i=1}^{k}(a_i\boldsymbol{y}_i + (m-1)a_i(\boldsymbol{y}_i - \boldsymbol{y}_{i-1}))$ (as defined in (4)) and the last equality is by the definition of the Lagrangian $\mathcal{L}(\cdot,\cdot)$ and $\{\check{\boldsymbol{v}}_i\}$ in Lemma 2.

Then by combining Eqs. (35)-(36) and Lemma 1, we have

$$\begin{aligned}&A_k(\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)) \\ &\leq \Big(\psi_k(\boldsymbol{y}_k) + \phi_k(\boldsymbol{x}_k) + ma_k\left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1})\right\rangle\Big) - \frac{1}{4\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 \\ &\quad - \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 + \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle - \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle \\ &\leq \Big(\frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_0\|^2 - \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_k\|^2 + \frac{\gamma}{2}\|\boldsymbol{u} - \boldsymbol{x}_0\|^2 - \frac{\gamma + \sigma A_k}{2}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2\Big) \\ &\qquad + ma_k\left\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1})\right\rangle - \frac{1}{4\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 - \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 \\ &\quad + \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle - \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle\,.\end{aligned}$$

Finally, we have from the fact that $\boldsymbol{y}_k$ and $\boldsymbol{y}_{k-1}$ differ in only the S^{j_k} components, Young's inequality, and the definition of a_k that

$$\begin{aligned}&|ma_k\langle \boldsymbol{x}_k - \boldsymbol{u}, \boldsymbol{A}^T(\boldsymbol{y}_k - \boldsymbol{y}_{k-1})\rangle| \\ &= |ma_k\,\langle \boldsymbol{A}^{S^{j_k}}(\boldsymbol{x}_k - \boldsymbol{u}), \boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}}\rangle| \\ &\leq ma_k\|\boldsymbol{A}^{S^{j_k}}(\boldsymbol{x}_k - \boldsymbol{u})\|\|\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}}\| \\ &\leq (ma_k)^2\gamma\|\boldsymbol{A}^{S^{j_k}}(\boldsymbol{x}_k - \boldsymbol{u})\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}}\|^2 \\ &\leq (m\hat{L}a_k)^2\gamma\|\boldsymbol{x}_k - \boldsymbol{u}\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_k^{S^{j_k}} - \boldsymbol{y}_{k-1}^{S^{j_k}}\|^2 \\ &= (m\hat{L}a_k)^2\gamma\|\boldsymbol{x}_k - \boldsymbol{u}\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 \\ &= \frac{\gamma + \sigma A_{k-1}}{4}\|\boldsymbol{x}_k - \boldsymbol{u}\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2 \\ &\leq \frac{\gamma + \sigma A_k}{4}\|\boldsymbol{x}_k - \boldsymbol{u}\|^2 + \frac{1}{4\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}_{k-1}\|^2\end{aligned}$$

leading to

$$
\begin{aligned}
&A_k(\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)) + \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_k\|^2 + \frac{\gamma + \sigma A_k}{4}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2 \\
\leq\ & \frac{\gamma}{2}\|\boldsymbol{u} - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_0\|^2 - \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 \\
&+ \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle - \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle
\end{aligned}
$$

and completing the proof. □

Lemma 6. *Suppose that* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ *is a Nash point for* (PD-GLP). *Then the iterates* $\boldsymbol{x}_k, \boldsymbol{y}_k$ *from Algorithm 1 satisfy*

$$
\begin{aligned}
&\mathbb{E}\Big[\frac{\gamma + \sigma A_k}{4}\|\boldsymbol{x}^* - \boldsymbol{x}_k\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_k\|^2 + \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \\
&\leq \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2,
\end{aligned} \tag{37}
$$

where the expectation is w.r.t. all the randomness in the algorithm.

Proof. Note that the existence of a Nash point is assumed in Assumption 2. With $(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{x}^*, \boldsymbol{y}^*)$, by the definition of Nash equilibrium, we have

$$
\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{y}^*) - \mathcal{L}(\boldsymbol{x}^*, \tilde{\boldsymbol{y}}_k) = (\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{y}^*) - \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{y}^*)) - (\mathcal{L}(\boldsymbol{x}^*, \tilde{\boldsymbol{y}}_k) - \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{y}^*)) \geq 0. \tag{38}
$$

By setting $(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{x}^*, \boldsymbol{y}^*)$ in the result of Lemma 5, using (38) to eliminate the first term on the left-hand side of the inequality, and rearranging, we obtain

$$
\begin{aligned}
&\frac{\gamma + \sigma A_k}{4}\|\boldsymbol{x}^* - \boldsymbol{x}_k\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_k\|^2 + \frac{1}{4\gamma}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2 \\
&\leq \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2 + \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle - \frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle .
\end{aligned}
$$

The result will follow when we show that the expectation (with respect to all the randomness) of the last two terms on the right-hand side is zero. Since $\boldsymbol{y}_{i-1} \in \mathcal{F}_{i-1}$, we have

$$
\mathbb{E}\left[\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle\right] = \mathbb{E}\left[\mathbb{E}\left[\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle \,\middle|\, \mathcal{F}_{i-1}\right]\right] = \mathbb{E}\left[\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \mathbb{E}\left[\check{\boldsymbol{v}}_i | \mathcal{F}_{i-1}\right]\rangle\right] = 0, \tag{39}
$$

which takes care of the second-last term. For any fixed $\boldsymbol{v} \in \mathbb{R}^n$, we also have

$$
\mathbb{E}\Big[\frac{1}{\gamma}\sum_{i=1}^{k}\langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle\Big] = 0, \tag{40}
$$

which takes care of the last term, and completes the proof. □

Next we state a technical lemma, proved in an earlier paper, which provides the final ingredient for the proof of Theorem 1.

Lemma 7. *[[55]] Let* $\{A_k\}_{k\geq 0}$ *be a sequence of nonnegative real numbers such that* $A_0 = 0$ *and* A_k *is defined recursively via* $A_k = A_{k-1} + \sqrt{c_1^2 + c_2 A_{k-1}}$, *where* $c_1 > 0$, *and* $c_2 \geq 0$. *Define* $K_0 = \lceil \frac{c_2}{9c_1} \rceil$. *Then*

$$
A_k \geq \begin{cases} \frac{c_2}{9}\Big(k - K_0 + \max\Big\{3\sqrt{\frac{c_1}{c_2}}, 1\Big\}\Big)^2, & \text{if } c_2 > 0 \text{ and } k > K_0, \\ c_1 k, & \text{otherwise}. \end{cases}
$$

We are now ready to prove our main result, which we restate here.

Theorem 1. *Let $\boldsymbol{x}_k$, $\boldsymbol{y}_k$, $k \in [K]$, be the iterates of Algorithm 1 and let $\tilde{\boldsymbol{x}}_k$, $\tilde{\boldsymbol{y}}_k$ be defined by*

$$\tilde{\boldsymbol{x}}_k = \frac{1}{A_k}\sum_{i=1}^{k} a_i \boldsymbol{x}_i, \quad \tilde{\boldsymbol{y}}_k = \frac{1}{A_k}\sum_{i=1}^{k}(a_i \boldsymbol{y}_i + (m-1)a_i(\boldsymbol{y}_i - \boldsymbol{y}_{i-1})), \tag{4}$$

for $k \in [K]$. Let $\mathcal{W}_k \subset \mathcal{X} \times \mathbb{R}^n$, $k \in [K]$, be a sequence of compact convex sets such that $(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{y}}_k) \in \mathcal{W}_k \subset \mathcal{W} \subset \mathcal{X} \times \mathbb{R}^n$, where $\mathcal{W}$ is also convex and compact. Then:

$$\begin{aligned} &\mathbb{E}\Big[\sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{W}_k}\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)\}\Big] \\ &\le \frac{1}{A_k}\bigg(\mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{u}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\hat{\boldsymbol{v}} - \boldsymbol{y}_0\|^2\Big] + \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2\bigg), \end{aligned} \tag{5}$$

where $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}) = \arg\sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{W}_k}\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)\}$. Furthermore,

$$\mathbb{E}\Big[\frac{\gamma + \sigma A_k}{4}\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}_k - \boldsymbol{y}^*\|^2\Big] \le \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2. \tag{6}$$

Define $K_0 = \lceil \frac{\sigma}{18\hat{L}m\gamma} \rceil$. Then in the bounds above:

$$A_k \ge \max\bigg\{\frac{k}{2\hat{L}m}, \frac{\sigma}{(6\hat{L}m)^2\gamma}\Big(k - K_0 + \max\Big\{3\sqrt{2\hat{L}m\gamma/\sigma}, 1\Big\}\Big)^2\bigg\}.$$

Proof. To provide a guarantee in terms of primal-dual gap, we need to take the supremum on $\boldsymbol{u}, \boldsymbol{v}$ of $\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k))\}$ over $\mathcal{W}_k$; we denote the $\arg\sup$ by $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}})$. When we subsequently take the expectation, we have to account for the fact that $\hat{\boldsymbol{v}}$ will be correlated with the randomness in the iteration history. We can however, use Lemmas 2 and 6 to give the upper bound on $\mathbb{E}\Big[-\sum_{i=1}^{k}\langle \hat{\boldsymbol{v}}, \check{\boldsymbol{v}}_i\rangle\Big]$.

From (34), using the fact that $\frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_k\|^2 + \frac{\gamma+\sigma A_k}{4}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2 \ge 0$, we have

$$\begin{aligned} &\mathbb{E}\Big[A_k \sup_{(\boldsymbol{u},\boldsymbol{v})\in\mathcal{W}_k}\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)\})\Big] \\ &\le \mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{u}} - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}_0 - \hat{\boldsymbol{v}}\|^2\Big] - \frac{1}{4\gamma}\mathbb{E}\Big[\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] + \frac{1}{\gamma}\mathbb{E}\Big[\sum_{i=1}^{k}\langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big] \\ &\quad + \frac{1}{\gamma}\mathbb{E}\Big[-\sum_{i=1}^{k}\langle \hat{\boldsymbol{v}}, \check{\boldsymbol{v}}_i\rangle\Big] \\ &\le \mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{u}} - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}_0 - \hat{\boldsymbol{v}}\|^2\Big] - \frac{1}{4\gamma}\mathbb{E}\Big[\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \\ &\quad + \frac{1}{\gamma}\mathbb{E}\Big[\frac{1}{2}\|\boldsymbol{y}_0 - \hat{\boldsymbol{v}}\|^2 + \frac{1}{2}\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \\ &= \mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{u}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\boldsymbol{y}_0 - \hat{\boldsymbol{v}}\|^2\Big] + \frac{1}{4\gamma}\mathbb{E}\Big[\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \\ &\le \mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{u}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\boldsymbol{y}_0 - \hat{\boldsymbol{v}}\|^2\Big] + \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2. \end{aligned} \tag{41}$$

where the first inequality is by Eq. (34), the second inequality is by Lemma 2 and (39), and the last inequality is by Lemma 6. This proves the first claim (5). The second claim (6) follows from Lemma 6 with the fact that $\frac{1}{4\gamma}\mathbb{E}\Big[\sum_{i=1}^{k}\|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \ge 0$. The final claim concerning A_k follows from Lemma 7 when we set

$$c_1 = \frac{1}{2\hat{L}m}, \quad c_2 = \frac{\sigma}{(2\hat{L}m)^2\gamma}.$$

□

Corollary 1. *In Algorithm 1, for all $k \geq 1$, $\tilde{\boldsymbol{x}}_k$ satisfies*

$$\mathbb{E}[\|\boldsymbol{y}^*\| \cdot \|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|] \leq \frac{\gamma\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2 + \frac{1}{\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2]}{A_k},$$

$$|\mathbb{E}[(\boldsymbol{c}^T\tilde{\boldsymbol{x}}_k + r(\tilde{\boldsymbol{x}}_k)) - (\boldsymbol{c}^T\boldsymbol{x}^* + r(\boldsymbol{x}^*))]| \leq \frac{\gamma\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2 + \frac{1}{\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2]}{A_k},$$

where $\boldsymbol{v} = 2\frac{\|\boldsymbol{y}^*\|}{\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|}(\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b})$.

Proof. Assume that $\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\| \neq 0$, as otherwise the first bound follows trivially. Let $\boldsymbol{u} = \boldsymbol{x}^*$ and $\boldsymbol{v} = \frac{2\|\boldsymbol{y}^*\|(\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b})}{\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|}$. Then we have

$$\begin{aligned}
&\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k) \\
&= (\boldsymbol{c}^T\tilde{\boldsymbol{x}}_k + r(\tilde{\boldsymbol{x}}_k) + 2\|\boldsymbol{y}^*\|\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|) - (\boldsymbol{c}^T\boldsymbol{x}^* + r(\boldsymbol{x}^*) + \tilde{\boldsymbol{y}}_k^T(\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{b})) \\
&= (\boldsymbol{c}^T(\tilde{\boldsymbol{x}}_k - \boldsymbol{x}^*) + r(\tilde{\boldsymbol{x}}_k) - r(\boldsymbol{x}^*)) + 2\|\boldsymbol{y}^*\|\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|. \qquad (42)
\end{aligned}$$

For any fixed $\boldsymbol{u}$, and any $\boldsymbol{v} \in \mathbb{R}^n$ possibly depending on the randomness in the algorithm, we have from Lemma 5, taking expectation over all the randomness, that

$$\begin{aligned}
&A_k\mathbb{E}[\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)] \\
&\leq \mathbb{E}\left[A_k(\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{v}) - \mathcal{L}(\boldsymbol{u}, \tilde{\boldsymbol{y}}_k)) + \frac{1}{2\gamma}\|\boldsymbol{v} - \boldsymbol{y}_k\|^2 + \frac{\gamma + \sigma A_k}{4}\|\boldsymbol{u} - \boldsymbol{x}_k\|^2\right] \\
&\leq \frac{\gamma}{2}\|\boldsymbol{u} - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2] - \mathbb{E}\Big[\frac{1}{4\gamma}\sum_{i=1}^k \|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \\
&\quad + \frac{1}{\gamma}\mathbb{E}\Big[\sum_{i=1}^k \langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big] - \frac{1}{\gamma}\mathbb{E}\Big[\sum_{i=1}^k \langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle\Big]. \qquad (43)
\end{aligned}$$

Meanwhile, we have

$$\begin{aligned}
&-\frac{1}{4\gamma}\mathbb{E}\Big[\sum_{i=1}^k \|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] - \frac{1}{\gamma}\mathbb{E}\Big[\sum_{i=1}^k \langle \boldsymbol{v}, \check{\boldsymbol{v}}_i\rangle\Big] \\
&\leq \frac{1}{2\gamma}\mathbb{E}\Big[\|\boldsymbol{y}_0 - \boldsymbol{v}\|^2\Big] + \frac{1}{4\gamma}\mathbb{E}\Big[\sum_{i=1}^k \|\boldsymbol{y}_i - \boldsymbol{y}_{i-1}\|^2\Big] \\
&\leq \frac{1}{2\gamma}\mathbb{E}[\|\boldsymbol{y}_0 - \boldsymbol{v}\|^2] + \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2,
\end{aligned}$$

where the first inequality is by Lemma 2 and the second inequality is by Lemma 6. Since $\boldsymbol{y}_{i-1} \in \mathcal{F}_{i-1}$, we have as in (39) that

$$\frac{1}{\gamma}\mathbb{E}\Big[\sum_{i=1}^k \langle \boldsymbol{y}_{i-1}, \check{\boldsymbol{v}}_i\rangle\Big] = 0. \qquad (44)$$

By combining Eq. (42)-(44) with $\boldsymbol{u} = \boldsymbol{x}^*$, we have

$$\begin{aligned}
&\mathbb{E}\big[(\boldsymbol{c}^T(\tilde{\boldsymbol{x}}_k - \boldsymbol{x}^*) + r(\tilde{\boldsymbol{x}}_k) - r(\boldsymbol{x}^*)) + 2\|\boldsymbol{y}^*\|\|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|\big] \\
&\leq \frac{\gamma\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2] + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2}{A_k}. \qquad (45)
\end{aligned}$$

By the KKT condition of (PD-GLP) and the optimality of $(\boldsymbol{x}^*, \boldsymbol{y}^*)$, we have for all $\boldsymbol{x} \in \mathcal{X}$ that

$$(\boldsymbol{c}^T\boldsymbol{x} + r(\boldsymbol{x})) - (\boldsymbol{c}^T\boldsymbol{x}^* + r(\boldsymbol{x}^*)) - \langle \boldsymbol{y}^*, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\rangle \geq 0, \qquad (46)$$

and thus

$$(\boldsymbol{c}^T\boldsymbol{x} + r(\boldsymbol{x})) - (\boldsymbol{c}^T\boldsymbol{x}^* + r(\boldsymbol{x}^*)) \geq -\|\boldsymbol{y}^*\|\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|. \qquad (47)$$

By combining Eq. (45) and Eq. (47), we have

$$\mathbb{E}[\|\boldsymbol{y}^*\| \cdot \|\boldsymbol{A}\tilde{\boldsymbol{x}}_k - \boldsymbol{b}\|] \leq \frac{\gamma\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\mathbb{E}[\|\boldsymbol{v} - \boldsymbol{y}_0\|^2] + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2}{A_k}, \qquad (48)$$

proving our first claim. The second claim is obtained by combining Eqs. (45), (47), and (48). □

B.2 Omitted Proof from Section 3.2

We need to compute explicitly only those components of $\boldsymbol{q}_{k-1}$ and $\boldsymbol{x}_k$ that are needed to update $\boldsymbol{y}_k$.

Lemma 8. *For $\{\boldsymbol{q}_k\}$ defined in Algorithm 1, we have*

$$\boldsymbol{q}_k = A_{k+1}(\boldsymbol{c} + \boldsymbol{z}_k) + \boldsymbol{r}_k, \quad k = 0, 1, 2, \ldots,$$

where $\boldsymbol{r}_0 = 0$ and for all $k = 1, 2, \ldots$

$$\boldsymbol{r}_k = \boldsymbol{r}_{k-1} + (ma_k - A_k)(\boldsymbol{z}_k - \boldsymbol{z}_{k-1}).$$

Proof. The proof is by induction. For $k = 0$, we have $\boldsymbol{q}_0 = A_1(\boldsymbol{c} + \boldsymbol{z}_0)$ which is true by definition. Assuming that the result holds for some index k, we show that it continues to hold for $k + 1$. We have from Step 9 of Algorithm 1, then using the inductive assumption, that

$$\begin{aligned}
\boldsymbol{q}_{k+1} &= a_{k+2}(\boldsymbol{z}_{k+1} + \boldsymbol{c}) + ma_{k+1}(\boldsymbol{z}_{k+1} - \boldsymbol{z}_k) + \boldsymbol{q}_k \\
&= a_{k+2}(\boldsymbol{z}_{k+1} + \boldsymbol{c}) + ma_{k+1}(\boldsymbol{z}_{k+1} - \boldsymbol{z}_k) + A_{k+1}(\boldsymbol{c} + \boldsymbol{z}_k) + \boldsymbol{r}_k \\
&= A_{k+2}\boldsymbol{c} + (A_{k+2} - A_{k+1})\boldsymbol{z}_{k+1} + ma_{k+1}(\boldsymbol{z}_{k+1} - \boldsymbol{z}_k) + A_{k+1}\boldsymbol{z}_k + \boldsymbol{r}_k \\
&= A_{k+2}(\boldsymbol{c} + \boldsymbol{z}_{k+1}) + (ma_{k+1} - A_{k+1})(\boldsymbol{z}_{k+1} - \boldsymbol{z}_k) + \boldsymbol{r}_k \\
&= A_{k+2}(\boldsymbol{c} + \boldsymbol{z}_{k+1}) + \boldsymbol{r}_{k+1},
\end{aligned}$$

as required. □

This lemma indicates that we can reconstruct $\boldsymbol{q}_k$ (or any subvector of $\boldsymbol{q}_k$ that we need, on demand), provided we maintain $\boldsymbol{z}_k$ and $\boldsymbol{r}_k$. Note that the update from $\boldsymbol{z}_k$ to $\boldsymbol{z}_{k+1}$ is sparse; these two vectors differ only in the components corresponding to the block C^{j_k}. To obtain $\boldsymbol{r}_{k+1}$ from $\boldsymbol{r}_k$, we need to add a scalar times the sparse vector $\boldsymbol{z}_{k+1} - \boldsymbol{z}_k$, so this update is also efficient.

We can also maintain an implicit representation of the averaged vector $\tilde{\boldsymbol{y}}_k$ efficiently, as shown in the following lemma. We defer the proofs of Lemmas 8 and 9 to Appendix B.

Lemma 9. *For $\{\tilde{\boldsymbol{y}}_k\}$ defined in* (4)*, we have*

$$\tilde{\boldsymbol{y}}_k = \boldsymbol{y}_k + \frac{1}{A_k}\boldsymbol{s}_k, \quad k = 1, 2, \ldots,$$

where $\boldsymbol{s}_0 = \boldsymbol{0}$ and for all $k = 1, 2, \ldots,$

$$\boldsymbol{s}_k = \boldsymbol{s}_{k-1} + ((m-1)a_k - A_{k-1})(\boldsymbol{y}_k - \boldsymbol{y}_{k-1}).$$

Proof. Recall that $\tilde{\boldsymbol{y}}_K$ $(K \geq 1)$ is defined in Step 11 of Algorithm 1. The proof is by induction. For $k = 1$, $\tilde{\boldsymbol{y}}_1 = \boldsymbol{y}_1 + \frac{1}{A_1}(m-1)a_1(\boldsymbol{y}_1 - \boldsymbol{y}_0) = \boldsymbol{y}_1 + \frac{1}{A_1}\boldsymbol{s}_1$ which is true by the definition of $\boldsymbol{s}_1$ and A_0. Next, we assume that the result holds for some $k \geq 2$ and show that it continues to hold for $k + 1$. We have

$$\begin{aligned}
A_{k+1}\tilde{\boldsymbol{y}}_{k+1} &= \sum_{i=1}^{k+1}(a_i\boldsymbol{y}_i + (m-1)a_i(\boldsymbol{y}_i - \boldsymbol{y}_{i-1})) \\
&= A_k\tilde{\boldsymbol{y}}_k + a_{k+1}\boldsymbol{y}_{k+1} + (m-1)a_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{y}_k) \\
&= A_k\boldsymbol{y}_k + \boldsymbol{s}_k + a_{k+1}\boldsymbol{y}_{k+1} + (m-1)a_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{y}_k) \\
&= A_k(\boldsymbol{y}_k - \boldsymbol{y}_{k+1} + \boldsymbol{y}_{k+1}) + \boldsymbol{s}_k + a_{k+1}\boldsymbol{y}_{k+1} + (m-1)a_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{y}_k) \\
&= A_{k+1}\boldsymbol{y}_{k+1} + \boldsymbol{s}_k + ((m-1)a_{k+1} - A_k)(\boldsymbol{y}_{k+1} - \boldsymbol{y}_k) \\
&= A_{k+1}\boldsymbol{y}_{k+1} + \boldsymbol{s}_{k+1}.
\end{aligned}$$

as required. □

B.3 Omitted Proof from Section 3.3

The standard-form LP (PD-LP) is derived by setting $r(\boldsymbol{x}) \equiv 0$ and $\mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^d : x_i \geq 0, i \in [d]\}$ in (PD-GLP). Given any $\boldsymbol{w} \in \mathcal{X} \times \mathbb{R}^n$, we define a compact convex subset $\mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})$ of $\mathcal{X} \times \mathbb{R}^n$ as follows:

$$\mathcal{W}_{\zeta,\gamma}(\boldsymbol{w}) = \{\hat{\boldsymbol{w}} \in \mathcal{X} \times \mathbb{R}^n : \|\hat{\boldsymbol{w}} - \boldsymbol{w}\|_{(\gamma)} \leq \zeta, \zeta > 0, \gamma > 0\}, \tag{49}$$

where we have defined $\|\cdot\|_{(\gamma)}$ by $\|\boldsymbol{w}\|_{(\gamma)} = \sqrt{\gamma\|\boldsymbol{x}\|^2 + \frac{1}{\gamma}\|\boldsymbol{y}\|^2}$ in Section 3.3.

Lemma 10. *Consider the standard-form LP* (LP)*. Given $\tau > 0$ and $\boldsymbol{w} \in \mathcal{X} \times \mathbb{R}^n$, if $\|\boldsymbol{w}\|_{(\gamma)} \leq \tau$ and $\zeta \leq \tau$, then*

$$\sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})} \{\mathcal{L}(\boldsymbol{x}, \hat{\boldsymbol{y}}) - \mathcal{L}(\hat{\boldsymbol{x}}, \boldsymbol{y})\} \geq \frac{\zeta}{\gamma + 1/\gamma + \tau} \text{LPMetric}(\boldsymbol{w}). \tag{50}$$

Proof. Let $\boldsymbol{F}(\boldsymbol{w}) = \begin{bmatrix} \boldsymbol{A}^T \boldsymbol{y} + \boldsymbol{c} \\ \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x} \end{bmatrix}$. Then we have

$$\rho_{\zeta,\gamma}(\boldsymbol{w}) := \sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})} \{\mathcal{L}(\boldsymbol{x}, \hat{\boldsymbol{y}}) - \mathcal{L}(\hat{\boldsymbol{x}}, \boldsymbol{y})\} = \sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})} \boldsymbol{F}(\boldsymbol{w})^T (\boldsymbol{w} - \hat{\boldsymbol{w}}) \geq 0, \tag{51}$$

where the inequality follows from $\boldsymbol{w} \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})$.

First, we prove

$$\rho_{\zeta,\gamma}(\boldsymbol{w}) \geq \frac{\zeta \| \max\{-\boldsymbol{A}^T \boldsymbol{y} - \boldsymbol{c},\, \boldsymbol{0}\}\|_2}{\gamma}. \tag{52}$$

If $\| \max\{-\boldsymbol{A}^T \boldsymbol{y} - \boldsymbol{c},\, \boldsymbol{0}\}\|_2 > 0$, for $\hat{\boldsymbol{w}}_1 = (\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{y}}_1)$, let $\hat{\boldsymbol{x}}_1 = \boldsymbol{x} + \frac{\zeta}{\gamma \| \max\{-\boldsymbol{A}^T \boldsymbol{y} - \boldsymbol{c},\, \boldsymbol{0}\}\|_2} \max\{-\boldsymbol{A}^T \boldsymbol{y} - \boldsymbol{c},\, \boldsymbol{0}\} \in \mathcal{X}$ and $\hat{\boldsymbol{y}}_1 = \boldsymbol{y}$. Then we have $\|\hat{\boldsymbol{w}}_1 - \boldsymbol{w}\|_{(\gamma)} = \zeta$ and thus $\hat{\boldsymbol{w}}_1 \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})$. So, as $\rho_{\zeta,\gamma}(\boldsymbol{w}) \geq \boldsymbol{F}(\boldsymbol{w})^T (\boldsymbol{w} - \hat{\boldsymbol{w}}_1)$, with the definition of $\hat{\boldsymbol{w}}_1$, Eq. (52) holds. If $\| \max\{-\boldsymbol{A}^T \boldsymbol{y} - \boldsymbol{c},\, \boldsymbol{0}\}\|_2 = 0$, then Eq. (52) holds trivially.

Second, we prove

$$\rho_{\zeta,\gamma}(\boldsymbol{w}) \geq \zeta\gamma \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2. \tag{53}$$

If $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2 > 0$, for $\hat{\boldsymbol{w}}_2 = (\hat{\boldsymbol{x}}_2, \hat{\boldsymbol{y}}_2)$, let $\hat{\boldsymbol{x}}_2 = \boldsymbol{x}$ and $\hat{\boldsymbol{y}}_2 = \boldsymbol{y} + \frac{\zeta\gamma}{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|}(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})$. Then we have $\|\hat{\boldsymbol{w}}_2 - \boldsymbol{w}\|_{(\gamma)} = \zeta$ and thus $\hat{\boldsymbol{w}}_2 \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})$. Then as $\rho_{\zeta,\gamma}(\boldsymbol{w}) \geq \boldsymbol{F}(\boldsymbol{w})^T (\boldsymbol{w} - \hat{\boldsymbol{w}}_2)$, Eq. (53) holds. If $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2 = 0$, then Eq. (53) holds trivially.

Third, we prove that

$$\rho_{\zeta,\gamma}(\boldsymbol{w}) \geq \frac{\zeta}{\tau} \max\{\boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{y},\, \boldsymbol{0}\}. \tag{54}$$

Note that the inequality holds trivially if $\boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{y} \leq 0$, so we assume that $\boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{y} > 0$, and note that $\boldsymbol{F}(\boldsymbol{w})^T \boldsymbol{w} = \boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{y} > 0$ in this case. This condition implies that $\boldsymbol{w} \neq 0$, so we can define $\hat{\boldsymbol{w}}_3 = \boldsymbol{w} - \min\left\{\frac{\zeta}{\|\boldsymbol{w}\|_{(\gamma)}}, 1\right\}\boldsymbol{w}$. Then we have $\|\hat{\boldsymbol{w}}_3 - \boldsymbol{w}\|_{(\gamma)} \leq \frac{\zeta}{\|\boldsymbol{w}\|_{(\gamma)}} \|\boldsymbol{w}\|_{(\gamma)} \leq \zeta$. Meanwhile, we also have $\hat{\boldsymbol{x}}_3 \geq \boldsymbol{x}_3 - \boldsymbol{x}_3 = 0$, so that $\hat{\boldsymbol{w}}_3 \in \mathcal{X} \times \mathbb{R}^n$. Thus, we have $\hat{\boldsymbol{w}}_3 \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})$. Then, with the assumptions $\zeta \leq \tau$ and $\|\boldsymbol{w}\|_{(\gamma)} \leq \tau$, together with $\boldsymbol{F}(\boldsymbol{w})^T \boldsymbol{w} > 0$, we have

$$\rho_{\zeta,\gamma}(\boldsymbol{w}) \geq \min\left\{\frac{\zeta}{\|\boldsymbol{w}\|_{(\gamma)}}, 1\right\} \boldsymbol{F}(\boldsymbol{w})^T \boldsymbol{w} \geq \min\left\{\frac{\zeta}{\tau}, 1\right\} \boldsymbol{F}(\boldsymbol{w})^T \boldsymbol{w} = \frac{\zeta}{\tau} \boldsymbol{F}(\boldsymbol{w})^T \boldsymbol{w} = \frac{\zeta}{\tau}(\boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{y}), \tag{55}$$

completing the proof of the claim (54).

By combining Eqs. (52), (53) and (54), we obtain

$$\begin{aligned} (\gamma + 1/\gamma + \tau)^2 \rho_{\zeta,\gamma}^2(\boldsymbol{w}) &\geq \zeta^2 (\| \max\{-\boldsymbol{A}^T \boldsymbol{y} - \boldsymbol{c}, 0\}\|_2^2 + \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \max\{\boldsymbol{c}^T \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{y}, 0\}^2) \\ &\geq \zeta^2 (\text{LPMetric}(\boldsymbol{w}))^2, \end{aligned} \tag{56}$$

from which the result follows. □

Lemma 11. *Consider the standard-form LP* (LP)*, and let $\boldsymbol{w}^* = (\boldsymbol{x}^*, \boldsymbol{y}^*)$ be a Nash point (that is, a solution of the primal-dual form* (PD-LP)*). For a starting point $\boldsymbol{w}_0$, define $\zeta = \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}$ and choose $\gamma > 0$. Then for all $k = 1, 2, \ldots$, we have*

$$\mathbb{E}[\|\tilde{\boldsymbol{w}}_k - \boldsymbol{w}^*\|_{(\gamma)}] \leq \sqrt{2} \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)},$$

where the expectation is taken w.r.t. all the randomness up to iteration k. Further, for $\mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)$ defined as in (49)*, we have*

$$\mathbb{E}\Big[\sup_{\boldsymbol{w} \in \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)} \{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{y}) - \mathcal{L}(\boldsymbol{y}, \tilde{\boldsymbol{y}}_k)\}\Big] \leq \frac{25\hat{L}m}{k} \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2. \tag{57}$$

Proof. By Theorem 1, we have

$$\mathbb{E}\Big[\frac{\gamma}{4}\|\boldsymbol{x}^* - \boldsymbol{x}_k\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_k\|^2\Big] \leq \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2 = \frac{1}{2}\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2,$$

by the definition of $\|\cdot\|_{(\gamma)}$. Using this definition again, we have that the left-hand side in this expression is bounded below by $\frac{1}{4}\|\boldsymbol{w}_k - \boldsymbol{w}^*\|_{(\gamma)}^2$, so that

$$\mathbb{E}\Big[\|\boldsymbol{w}_k - \boldsymbol{w}^*\|_{(\gamma)}^2] \leq 2\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2, \quad \forall k \geq 1. \tag{58}$$

By convexity of $\|\cdot\|^2$, we obtain

$$\mathbb{E}[\|\tilde{\boldsymbol{w}}_k - \boldsymbol{w}^*\|_{(\gamma)}^2] = \mathbb{E}\Big[\Big\|\frac{1}{k}\sum_{i=1}^{k}(\boldsymbol{w}_i - \boldsymbol{w}^*)\Big\|_{(\gamma)}^2\Big] \leq \frac{1}{k}\mathbb{E}\Big[\sum_{i=1}^{k}\|\boldsymbol{w}_i - \boldsymbol{w}^*\|_{(\gamma)}^2\Big] \leq 2\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2. \tag{59}$$

The first claim of the lemma now follows by applying Jensen's inequality to this bound.

For $\hat{\boldsymbol{w}} = (\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}) \in \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)$, we have the following bound on $\mathbb{E}\Big[\gamma\|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\hat{\boldsymbol{y}} - \boldsymbol{y}_0\|^2\Big]$:

$$\begin{aligned}
&\mathbb{E}\Big[\gamma\|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\hat{\boldsymbol{y}} - \boldsymbol{y}_0\|^2\Big] \\
&= \mathbb{E}\big[\|\boldsymbol{w}_0 - \hat{\boldsymbol{w}}\|_{(\gamma)}^2\big] \\
&= \mathbb{E}\big[\|\boldsymbol{w}_0 - \boldsymbol{w}^* + \boldsymbol{w}^* - \tilde{\boldsymbol{w}}_k + \tilde{\boldsymbol{w}}_k - \hat{\boldsymbol{w}}\|_{(\gamma)}^2\big] \\
&\leq \mathbb{E}\big[(\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \|\boldsymbol{w}^* - \tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\tilde{\boldsymbol{w}}_k - \hat{\boldsymbol{w}}\|_{(\gamma)})^2\big] \\
&\leq \mathbb{E}\big[3(\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2 + \|\boldsymbol{w}^* - \tilde{\boldsymbol{w}}_k\|_{(\gamma)}^2 + \|\tilde{\boldsymbol{w}}_k - \hat{\boldsymbol{w}}\|_{(\gamma)}^2)\big] \\
&\leq \mathbb{E}\big[3(\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2 + 2\|\boldsymbol{w}^* - \boldsymbol{w}_0\|_{(\gamma)}^2 + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2)\big] \\
&= 12\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2,
\end{aligned} \tag{60}$$

where the first inequality is by the triangle inequality, the second inequality is by the arithmetic inequality, the third inequality is by Eq. (59) and our assumption $\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)$ with $\zeta = \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}$.

For the case of linear programming, the strong convexity constant is $\sigma = 0$, so that $A_k = \frac{k}{2\hat{L}m}$ in CLVR. Thus, by applying Theorem 1 with $\mathcal{W}_k = \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)$ and $\hat{\boldsymbol{w}} = (\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}) = \arg\sup_{\boldsymbol{w}=(\boldsymbol{x},\boldsymbol{y})\in\mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)}\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{y}) - \mathcal{L}(\boldsymbol{x}, \tilde{\boldsymbol{y}}_k)\}$, and using the definition of $\|\cdot\|_{(\gamma)}$ and Eq. (60), we have

$$\begin{aligned}
&\mathbb{E}\Big[\sup_{\boldsymbol{w}\in\mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)}\{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \boldsymbol{y}) - \mathcal{L}(\boldsymbol{y}, \tilde{\boldsymbol{y}}_k)\}\Big] \\
&\leq \frac{2\hat{L}m}{k}\Big(\mathbb{E}\Big[\frac{\gamma}{2}\|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|^2 + \frac{1}{\gamma}\|\hat{\boldsymbol{y}} - \boldsymbol{y}_0\|^2\Big] + \frac{\gamma}{2}\|\boldsymbol{x}^* - \boldsymbol{x}_0\|^2 + \frac{1}{2\gamma}\|\boldsymbol{y}^* - \boldsymbol{y}_0\|^2\Big) \\
&\leq \frac{2\hat{L}m}{k}\Big(\mathbb{E}\big[\|\boldsymbol{w}_0 - \hat{\boldsymbol{w}}\|_{(\gamma)}^2\big] + \frac{1}{2}\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2\Big) \\
&\leq \frac{2\hat{L}m}{k}\Big(12\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2 + \frac{1}{2}\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2\Big) \\
&\leq \frac{25\hat{L}m}{k}\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2.
\end{aligned} \tag{61}$$

□

Then with Theorem 1, Lemmas 10 and 11, we give our theorem for the fixed restart strategy.

Theorem 2. *Consider the* CLVR *algorithm applied to the standard-form LP problem* (PD-LP)*, with input* $\boldsymbol{w}_0$ *and output* $\tilde{\boldsymbol{w}}_k$*. Given* $\gamma > 0$*, define* $\boldsymbol{w}^* = \arg\min_{w\in\mathcal{W}^*}\|\boldsymbol{w}_0 - \boldsymbol{w}\|_{(\gamma)}$*, and define*

$C_0 = \gamma + 1/\gamma + (\sqrt{2}+1)\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \|\boldsymbol{w}^*\|_{(\gamma)}$. *Then for H_γ defined as in* (9)*, we have*

$$\mathbb{E}\Big[\sqrt{\text{dist}(\tilde{\boldsymbol{w}}_k, \mathcal{W}^*)_{(\gamma)}}\,\Big] \le 5\sqrt{\frac{\hat{L}mC_0}{H_\gamma k}}\sqrt{\text{dist}(\boldsymbol{w}_0, \mathcal{W}^*)_{(\gamma)}},$$

$$\mathbb{E}\big[\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}\big] \le 5\sqrt{\frac{\hat{L}mC_0}{H_\gamma k}}\sqrt{\text{LPMetric}(\boldsymbol{w}_0)}.$$

Proof. Applying Lemma 10 with $\boldsymbol{w} = \tilde{\boldsymbol{w}}_k, \tau = \|\tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}$ and $\zeta = \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} \le \tau$, we have

$$\sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\boldsymbol{w})} \{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \hat{\boldsymbol{y}}) - \mathcal{L}(\hat{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_k)\} \ge \frac{\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}{\gamma + 1/\gamma + \|\tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}}. \tag{62}$$

As $\frac{x^2}{y}$ is jointly convex in x, y on the domain $\{(x,y) : x \in \mathbb{R}, y > 0\}$ [11, Example 3.18] using Jensen's inequality, we have that $\mathbb{E}[\frac{x^2}{y}] \ge \frac{(\mathbb{E}[x])^2}{\mathbb{E}[y]}$. (In our case, this simply follows as x, y depend on the same source of randomness.) Applying this inequality to (62) with $x = \sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}$ and $y = \gamma + 1/\gamma + \|\tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}$, we obtain

$$\begin{aligned}\mathbb{E}\Big[\sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)} \{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \hat{\boldsymbol{y}}) - \mathcal{L}(\hat{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_k)\}\Big] &\ge \mathbb{E}\left[\frac{\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}(\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)})^2}{\gamma + 1/\gamma + \|\tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}}\right] \\ &\ge \frac{\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}\Big(\mathbb{E}\big[\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}\big]\Big)^2}{\mathbb{E}\big[\gamma + 1/\gamma + \|\tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}\big]}.\end{aligned} \tag{63}$$

Using Lemma 11, we have

$$\begin{aligned}&\mathbb{E}\big[\gamma + 1/\gamma + \|\tilde{\boldsymbol{w}}_k\|_{(\gamma)} + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}\big] \\ &= \gamma + 1/\gamma + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \mathbb{E}[\|\tilde{\boldsymbol{w}}_k\|_{(\gamma)}] \\ &\le \gamma + 1/\gamma + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \mathbb{E}[\|\boldsymbol{w}^*\|_{(\gamma)} + \|\tilde{\boldsymbol{w}}_k - \boldsymbol{w}^*\|_{(\gamma)}] \\ &\le \gamma + 1/\gamma + \|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \|\boldsymbol{w}^*\|_{(\gamma)} + \sqrt{2}\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} \\ &= \gamma + 1/\gamma + (\sqrt{2}+1)\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} + \|\boldsymbol{w}^*\|_{(\gamma)}.\end{aligned} \tag{64}$$

By combining Eqs. (63) and (64) and using the definition of C_0, we have

$$\begin{aligned}\mathbb{E}\Big[\sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)} \{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \hat{\boldsymbol{y}}) - \mathcal{L}(\hat{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_k)\}\Big] &\ge \frac{\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}}{C_0}\Big(\mathbb{E}\big[\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}\big]\Big)^2 \\ &\ge \frac{\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}}{C_0}\Big(\mathbb{E}\big[\sqrt{H_\gamma \text{dist}(\tilde{\boldsymbol{w}}_k, \mathcal{W}^*)_{(\gamma)}}\big]\Big)^2,\end{aligned} \tag{65}$$

where the last inequality is by the definition of Hoffman constant in Eq. (9). Meanwhile, by Lemma 11, we have:

$$\mathbb{E}\Big[\sup_{\hat{\boldsymbol{w}} \in \mathcal{W}_{\zeta,\gamma}(\tilde{\boldsymbol{w}}_k)} \{\mathcal{L}(\tilde{\boldsymbol{x}}_k, \hat{\boldsymbol{y}}) - \mathcal{L}(\hat{\boldsymbol{x}}, \tilde{\boldsymbol{y}}_k)\}\Big] \le \frac{25\hat{L}m}{k}\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)}^2. \tag{66}$$

Now, recalling that $\boldsymbol{w}^* = \arg\min_{\boldsymbol{w} \in \mathcal{W}^*} \|\boldsymbol{w}_0 - \boldsymbol{w}\|_{(\gamma)}$ and using Eq. (9), we have

$$\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|_{(\gamma)} = \text{dist}(\boldsymbol{w}_0, \mathcal{W}^*)_{(\gamma)} \le \frac{1}{H_\gamma}\text{LPMetric}(\boldsymbol{w}_0). \tag{67}$$

By combining Eqs. (65)-(67), we have

$$\begin{aligned}\frac{1}{C_0}\Big(\mathbb{E}\Big[\sqrt{H_\gamma \text{dist}(\tilde{\boldsymbol{w}}_k, \mathcal{W}^*)_{(\gamma)}}\,\Big]\Big)^2 &\le \frac{1}{C_0}\Big(\mathbb{E}\big[\sqrt{\text{LPMetric}(\tilde{\boldsymbol{w}}_k)}\big]\Big)^2 \\ &\le \frac{25\hat{L}m}{k}\text{dist}(\boldsymbol{w}_0, \mathcal{W}^*)_{(\gamma)} \\ &\le \frac{25\hat{L}m}{H_\gamma k}\text{LPMetric}(\boldsymbol{w}_0).\end{aligned} \tag{68}$$

Both bounds follow from this chain of inequalities. □

Proof. By the sharpness property, $\|\tilde{\boldsymbol{w}}_k - \boldsymbol{w}^*\|_{(\gamma)} \leq \frac{\mathrm{LPMetric}(\tilde{\boldsymbol{w}}_k)}{H_\gamma}$. Since one call to CLVR between restarts halves the LPMetric, we have $\mathrm{LPMetric}(\tilde{\boldsymbol{w}}_k) \leq \frac{1}{2}\mathrm{LPMetric}(\boldsymbol{w}_0)$. Combining these two inequalities leads to the claimed bound. □

Finally, we summarize the adaptive restart strategy in Algorithm 4.

Algorithm 4 Lazy CLVR with Adaptive Restarts

1: **Input:** $\epsilon > 0$, $\boldsymbol{x}_0 \in \mathcal{X}, \boldsymbol{y}_0 \in \mathbb{R}^n$, $\gamma > 0$, $\hat{L} > 0$, K, $\widehat{K}$, m, $\{S^1, S^2, \ldots, S^m\}$, $\{C^1, C^2, \ldots, C^m\}$
2: $t = 0$, $\boldsymbol{x}_0^{(0)} = \boldsymbol{x}_0$, $\boldsymbol{y}_0^{(0)} = \boldsymbol{y}_0$, $\boldsymbol{w}^{(0)} = (\boldsymbol{x}_0^{(0)}, \boldsymbol{y}_0^{(0)})$
3: **repeat**
4: Run Lazy CLVR (Algorithm 2) until $\mathrm{LPMetric}(\boldsymbol{w}^{(t+1)}) \leq \frac{1}{2}\mathrm{LPMetric}(\boldsymbol{w}^{(t)})$ where, $\boldsymbol{w}^{(t+1)} = \tilde{\boldsymbol{w}}_K^{(t)} = (\tilde{\boldsymbol{x}}_K^{(t)}, \tilde{\boldsymbol{y}}_K^{(t)})$ and $\tilde{\boldsymbol{x}}_K^{(t)}, \tilde{\boldsymbol{y}}_K^{(t)}$ are the output points of Lazy CLVR
5: $t = t + 1$
6: **until** $\mathrm{LPMetric}(\boldsymbol{w}^{(t)}) \leq \epsilon$
7: **Return:** $\boldsymbol{w}^{(t)}$

C Omitted Proofs from Section 4

C.1 DRO with f-divergence-based Ambiguity Set

Theorem 3. *Let $\mathcal{X}$ be a compact convex set. Then the DRO problem in Eq.* (10) *is equivalent to*

$$\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{w},\boldsymbol{\mu},\boldsymbol{q},\gamma} \quad & \Big\{\gamma + \frac{\rho\mu_1}{n} + \frac{1}{n}\sum_{i=1}^n \mu_i f^*\Big(\frac{q_i}{\mu_i}\Big)\Big\} \\
\text{s.t.} \quad & \boldsymbol{w} + \boldsymbol{v} - \frac{\boldsymbol{q}}{n} - \gamma \mathbf{1}_n = \mathbf{0}_n, \\
& u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, && i \in [n] \\
& \mu_1 = \mu_2 = \cdots = \mu_n, \\
& g(u_i) \leq w_i, && i \in [n] \\
& q_i \in \mu_i \,\mathrm{dom}(f^*), && i \in [n] \\
& v_i \geq 0,\ \mu_i \geq 0, && i \in [n] \\
& \boldsymbol{x} \in \mathcal{X}.
\end{aligned}$$

Proof. Since the context is clear, to simplify the notation, in the following we use $\mathcal{P}$ to denote $\mathcal{P}_{\rho,n}$. First, using Sion's minimax theorem, we have that

$$\min_{\boldsymbol{x}\in\mathcal{X}} \sup_{\boldsymbol{p}\in\mathcal{P}} \sum_{i=1}^n p_i g(b_i \boldsymbol{a}_i^T \boldsymbol{x}) = \sup_{\boldsymbol{p}\in\mathcal{P}} \min_{\boldsymbol{x}\in\mathcal{X}} \sum_{i=1}^n p_i g(b_i \boldsymbol{a}_i^T \boldsymbol{x}).$$

Introducing auxiliary variables $\boldsymbol{w}$ and $\boldsymbol{u}$, the problem is further equivalent to

$$\sup_{\boldsymbol{p}\in\mathcal{P}} \min_{\substack{\boldsymbol{x}\in\mathcal{X},\boldsymbol{w},\boldsymbol{u}: \\ u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, i\in[n], \\ g(u_i)\leq w_i, i\in[n]}} \boldsymbol{p}^T \boldsymbol{w} \equiv \min_{\substack{\boldsymbol{x}\in\mathcal{X},\boldsymbol{w},\boldsymbol{u}: \\ u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, i\in[n], \\ g(u_i)\leq w_i, i\in[n]}} \sup_{\boldsymbol{p}\in\mathcal{P}} \boldsymbol{p}^T \boldsymbol{w},$$

where the last equivalence is by applying minimax equality, which holds due to compactness of $\mathcal{P}$ [56]. Hence, we can conclude that

$$\min_{\boldsymbol{x}\in\mathcal{X}} \sup_{\boldsymbol{p}\in\mathcal{P}} \sum_{i=1}^n p_i g(b_i \boldsymbol{a}_i^T \boldsymbol{x}) = \min_{\substack{\boldsymbol{x}\in\mathcal{X},\boldsymbol{w},\boldsymbol{u}: \\ u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, i\in[n], \\ g(u_i)\leq w_i, i\in[n]}} \sup_{\boldsymbol{p}\in\mathcal{P}} \boldsymbol{p}^T \boldsymbol{w}. \tag{69}$$

For a fixed tuple $(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{u})$, using Lagrange multipliers to enforce the constraints from $\mathcal{P}$, we can further write

$$
\begin{aligned}
&\sup_{\boldsymbol{p}\in\mathcal{P}} \boldsymbol{p}^T\boldsymbol{w} = -\inf_{\boldsymbol{p}\in\mathcal{P}} -\boldsymbol{p}^T\boldsymbol{w} \\
&= -\inf_{\boldsymbol{p}\in\mathbb{R}^n}\Big(-\boldsymbol{p}^T\boldsymbol{w} + \sup_{\boldsymbol{v}\geq 0,\gamma\in\mathbb{R},\mu\geq 0}\Big(-\boldsymbol{p}^T\boldsymbol{v} + \gamma\Big(\sum_{i=1}^n p_i - 1\Big) + \mu\Big(D_f(\boldsymbol{p}\|\mathbf{1}/n) - \frac{\rho}{n}\Big)\Big)\Big) \\
&= \sup_{\boldsymbol{p}\in\mathbb{R}^n}\inf_{\boldsymbol{v}\geq 0,\gamma\in\mathbb{R},\mu\geq 0}\Big(\boldsymbol{p}^T\boldsymbol{w} + \boldsymbol{p}^T\boldsymbol{v} - \gamma\Big(\sum_{i=1}^n p_i - 1\Big) - \mu\Big(D_f(\boldsymbol{p}\|\mathbf{1}/n) - \frac{\rho}{n}\Big)\Big).
\end{aligned}
$$

Now, using the definitions of D_f and the convex conjugate of f, we have

$$
D_f(\boldsymbol{p}\|\mathbf{1}/n) = \sum_{i=1}^n \frac{1}{n} f(np_i) = \sum_{i=1}^n \frac{1}{n}\sup_{\nu_i\in\mathrm{dom}(f^*)}\big(np_i\nu_i - f^*(\nu_i)\big). \tag{70}
$$

As a result, we have

$$
\begin{aligned}
&\inf_{\mu\geq 0} -\mu\Big(D_f(\boldsymbol{p}\|\mathbf{1}/n) - \frac{\rho}{n}\Big) \\
&= \inf_{\mu\geq 0} -\frac{\mu}{n}\Big(\sum_{i=1}^n \sup_{\nu_i\in\mathrm{dom}(f^*), i\in[n]}\big(np_i\nu_i - f^*(\nu_i)\big) - \rho\Big) \\
&= \inf_{\mu\geq 0,\nu_i\in\mathrm{dom}(f^*), i\in[n]} -\frac{\mu}{n}\Big(\sum_{i=1}^n \big(np_i\nu_i - f^*(\nu_i)\big) - \rho\Big) \\
&= \inf_{\mu\geq 0, q_i\in\mu\,\mathrm{dom}(f^*), i\in[n]} \frac{1}{n}\sum_{i=1}^n\big(-p_iq_i + \mu f^*\big(\frac{q_i}{n\mu}\big)\big) + \frac{\mu\rho}{n} \\
&= \inf_{\substack{\mu_1=\mu_2=\cdots=\mu_n\geq 0,\\ q_i\in\mu_i\,\mathrm{dom}(f^*), i\in[n]}} \frac{1}{n}\sum_{i=1}^n\big(-p_iq_i + \mu_i f^*\big(\frac{q_i}{n\mu_i}\big)\big) + \frac{\mu_1\rho}{n},
\end{aligned}
$$

where the first equality is by Eq. (70), the third one is by the variable substitution $q_i = n\mu\nu_i$, and the last one is by introducing $\mu_1, \mu_2, \ldots, \mu_n$ to replace μ. Then each $n\mu_i f^*\big(\frac{q_i}{n\mu_i}\big)(i \in [n])$ is a perspective function of f^* [11], which is jointly convex w.r.t. (μ_i, q_i). Hence, we can conclude that

$$
\begin{aligned}
&\sup_{\boldsymbol{p}\in\mathcal{P}} \boldsymbol{p}^T\boldsymbol{w} \\
&= \sup_{\boldsymbol{p}\in\mathbb{R}^n}\inf_{\substack{\gamma\in\mathbb{R},\boldsymbol{v}\geq\mathbf{0},\\ \mu_1=\mu_2=\cdots=\mu_n\geq 0,\\ q_i\in\mu_i\,\mathrm{dom}(f^*), i\in[n]}}\Big(\boldsymbol{p}^T\boldsymbol{w} + \boldsymbol{p}^T\boldsymbol{v} - \gamma\Big(\sum_{i=1}^n p_i - 1\Big) + \frac{1}{n}\sum_{i=1}^n\big(-p_iq_i + \mu_i f^*\big(\frac{q_i}{n\mu_i}\big)\big) + \frac{\mu_1\rho}{n}\Big) \\
&= \inf_{\substack{\gamma\in\mathbb{R},\boldsymbol{v}\geq\mathbf{0},\\ \mu_1=\mu_2=\cdots=\mu_n\geq 0,\\ q_i\in\mu_i\,\mathrm{dom}(f^*), i\in[n]}}\sup_{\boldsymbol{p}\in\mathbb{R}^n}\Big(\boldsymbol{p}^T\boldsymbol{w} + \boldsymbol{p}^T\boldsymbol{v} - \gamma\Big(\sum_{i=1}^n p_i - 1\Big) + \frac{1}{n}\sum_{i=1}^n\big(-p_iq_i + \mu_i f^*\big(\frac{q_i}{n\mu_i}\big)\big) + \frac{\mu_1\rho}{n}\Big),
\end{aligned}
$$

where the last line is by strong duality. Thus, combining with Eq. (69), we conclude that the original DRO problem with f-divergence based ambiguity set is equivalent to the following problem:

$$
\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{w},\boldsymbol{\mu},\boldsymbol{q},\gamma}\ \max_{\boldsymbol{p}\in\mathbb{R}^n}\ & \Big\{\gamma + \frac{\rho\mu_1}{n} + \frac{1}{n}\sum_{i=1}^n \mu_i f^*\Big(\frac{q_i}{\mu_i}\Big) + \boldsymbol{p}^T\Big(\boldsymbol{w} + \boldsymbol{v} - \frac{\boldsymbol{q}}{n} - \gamma\mathbf{1}_n\Big)\Big\} \\
\text{s.t.}\ \ & u_i = b_i\boldsymbol{a}_i^T\boldsymbol{x}, && i\in[n] \\
& \mu_1 = \mu_2 = \cdots = \mu_n, \\
& g(u_i) \leq w_i, && i\in[n] \\
& q_i \in \mu_i\,\mathrm{dom}(f^*), && i\in[n] \\
& v_i \geq 0, \mu_i \geq 0, && i\in[n] \\
& \boldsymbol{x}\in\mathcal{X}.
\end{aligned}
$$

Finally, noticing that the maximization problem over $\boldsymbol{p} \in \mathbb{R}^n$ enforces the equality constraint $\boldsymbol{w} + \boldsymbol{v} - \frac{\boldsymbol{q}}{n} - \gamma \mathbf{1}_n = \mathbf{0}_n$, we obtain

$$\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{\mu},\boldsymbol{q},\gamma} \; & \Big\{\gamma + \frac{\rho\mu_1}{n} + \frac{1}{n}\sum_{i=1}^n \mu_i f^*\big(\frac{q_i}{\mu_i}\big)\Big\} \\
\text{s.t.} \;\; & \boldsymbol{w} + \boldsymbol{v} - \frac{\boldsymbol{q}}{n} - \gamma \mathbf{1}_n = \mathbf{0}_n, \\
& u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, && i \in [n] \\
& \mu_1 = \mu_2 = \cdots = \mu_n, \\
& g(u_i) \le w_i, && i \in [n] \\
& q_i \in \mu_i \operatorname{dom}(f^*), && i \in [n] \\
& v_i \ge 0, \mu_i \ge 0, && i \in [n] \\
& \boldsymbol{x} \in \mathcal{X},
\end{aligned}$$

as claimed. □

Example: Conditional Value at Risk (CVaR) with Hinge Loss. As a specific example of an application of Theorem 3, we consider CVaR at level $\alpha \in (0,1)$, which leads to the optimization problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} \sup_{\substack{\boldsymbol{p}\ge 0, \mathbf{1}_n^T\boldsymbol{p}=1 \\ p_i \le \frac{1}{\alpha n}(i\in[d])}} \sum_{i=1}^n p_i g(b_i \boldsymbol{a}_i^T \boldsymbol{x}), \tag{71}$$

where $g(u_i) = \max\{0, 1-u_i\}$ is the hinge loss. Here the ambiguity set constraint reduces to simple bounds $p_i \le \frac{1}{\alpha n}$ for $i \in [n]$, so the reformulation based on the convex perspective function can be avoided altogether and replaced by simple Lagrange multipliers for this linear constraint. In particular, in the proof of Theorem 3, we can write

$$\begin{aligned}
\sup_{\boldsymbol{p}\in\mathcal{P}} \boldsymbol{p}^T\boldsymbol{w} &= \sup_{\boldsymbol{p}\in\mathbb{R}^n} \inf_{\boldsymbol{v}\ge 0, \gamma\in\mathbb{R}, \boldsymbol{\mu}\ge 0} \Big(\boldsymbol{p}^T\boldsymbol{w} + \boldsymbol{p}^T\boldsymbol{v} - \gamma\Big(\sum_{i=1}^n p_i - 1\Big) - \sum_{i=1}^n \mu_i\Big(p_i - \frac{1}{\alpha n}\Big)\Big) \\
&= \sup_{\boldsymbol{p}\in\mathbb{R}^n} \inf_{\boldsymbol{v}\ge 0, \gamma\in\mathbb{R}, \boldsymbol{\mu}\ge 0} \Big(\gamma + \boldsymbol{p}^T(\boldsymbol{w} + \boldsymbol{v} - \gamma\mathbf{1}_n - \boldsymbol{\mu}) + \frac{1}{\alpha n}\boldsymbol{\mu}^T\mathbf{1}_n\Big)\Big) \\
&= \inf_{\boldsymbol{v}\ge 0, \gamma\in\mathbb{R}, \boldsymbol{\mu}\ge 0} \sup_{\boldsymbol{p}\in\mathbb{R}^n} \Big(\gamma + \boldsymbol{p}^T(\boldsymbol{w} + \boldsymbol{v} - \gamma\mathbf{1}_n - \boldsymbol{\mu}) + \frac{1}{\alpha n}\boldsymbol{\mu}^T\mathbf{1}_n\Big)\Big).
\end{aligned}$$

Following the argument from the proof of Theorem 3, and expressing $w_i \ge g(u_i)$ equivalently as the pair of constraints $w_i \ge 0$ and $w_i \ge 1 - u_i$, the problem reduces to

$$\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{w},\boldsymbol{\mu},\gamma} \; & \Big\{\gamma + \frac{1}{\alpha n}\boldsymbol{\mu}^T\mathbf{1}_n\Big\} \\
\text{s.t.} \;\; & \boldsymbol{w} + \boldsymbol{v} - \gamma\mathbf{1}_n - \boldsymbol{\mu} = \mathbf{0}_n, \\
& u_i = b_i \boldsymbol{a}_i^T \boldsymbol{x}, && i \in [n], \\
& w_i \ge 0, \; w_i \ge 1 - u_i, && i \in [n], \\
& v_i \ge 0, \; \mu_i \ge 0, && i \in [n],
\end{aligned} \tag{72}$$

which is a linear program. To write it in the standard form, we further introduce slack variables $\boldsymbol{s} \in \mathbb{R}^n$, $\boldsymbol{s} \ge \mathbf{0}$, to replace inequality constraints $w_i \ge 1 - u_i$ by $s_i - u_i - w_i = -1$. For implementation purposes, we define $\mathcal{X}$ to be the set of simple non-negativity constraints ($w_i \ge 0, s_i \ge 0,, v_i \ge 0, \mu_i, \ge 0, \forall i \in [n]$) from Eq. (72). The problem then becomes

$$\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{v},\boldsymbol{w},\boldsymbol{\mu},\boldsymbol{s},\gamma} \; & \Big\{\gamma + \frac{1}{\alpha n}\boldsymbol{\mu}^T\mathbf{1}_n\Big\} \\
\text{s.t.} \;\; & \boldsymbol{w} + \boldsymbol{v} - \gamma\mathbf{1}_n - \boldsymbol{\mu} = \mathbf{0}_n, \\
& u_i - b_i \boldsymbol{a}_i^T \boldsymbol{x} = 0, && i \in [n], \\
& \boldsymbol{s} - \boldsymbol{u} - \boldsymbol{w} = -\mathbf{1}_n, \\
& \boldsymbol{w} \ge \mathbf{0}_n, \; \boldsymbol{v} \ge \mathbf{0}_n, \; \boldsymbol{\mu} \ge \mathbf{0}_n, \; \boldsymbol{s} \ge \mathbf{0}_n.
\end{aligned}$$

C.2 DRO with Wasserstein Metric-based Ambiguity Set

Definition 1. *Let $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ be two probability distributions supported on $\Theta = \mathbb{R}^d \times \{1, -1\}$ and let $\Pi(\boldsymbol{\mu}, \boldsymbol{\nu})$ denote the set of all joint distributions between $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$. Then the Wasserstein metric between $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ is defined by*

$$W(\boldsymbol{\mu}, \boldsymbol{\nu}) = \inf_{\pi \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \int_{\Theta \times \Theta} \zeta(\boldsymbol{\xi}, \boldsymbol{\xi}') \pi(d\boldsymbol{\xi}, d\boldsymbol{\xi}'), \tag{73}$$

where $\boldsymbol{\xi}, \boldsymbol{\xi}' \in \Theta$ and $\zeta(\cdot, \cdot) : \Theta \times \Theta \to \mathbb{R}_+$ is a convex cost function defined by

$$\zeta((\boldsymbol{a}, b), (\boldsymbol{a}', b')) = \|\boldsymbol{a} - \boldsymbol{a}'\| + \kappa |b - b'|,$$

where $\|\cdot\|$ denotes a general norm and $\kappa > 0$ is used to balance the feature mismatch and label uncertainty. Let $\mathbb{Q} = \frac{1}{n} \sum_{i=1}^n \delta_{(\boldsymbol{a}_i, b_i)}$, where $\delta_{(\boldsymbol{a}_i, b_i)}$ is the Dirac Delta function (or a point mass) at point $(\boldsymbol{a}_i, b_i)$. Then, the Wasserstein metric based ambiguity set is defined as

$$\mathcal{P}_{\rho,\kappa} = \Big\{ \mathbb{P} : W(\mathbb{P}, \mathbb{Q}) \leq \rho \Big\}. \tag{74}$$

Assumption 5. $\mathbb{R} \ni M = \sup_{\theta \in \mathrm{dom}(g^*)} |\theta|$.

To obtain our main result in Theorem 12, we first show the following auxiliary lemma that is subsequently used in the proof of Theorem 4. A similar result for the special case of a logistic loss function can be found in [52].

Lemma 12. *Let $(\boldsymbol{a}', b') \in \mathbb{R}^d \times \{1, -1\}$ be a given pair of data sample and label. Then, for every $\lambda > 0$, we have*

$$\sup_{\boldsymbol{a} \in \mathbb{R}^d} g(b' \boldsymbol{a}^T \boldsymbol{w}) - \lambda \|\boldsymbol{a} - \boldsymbol{a}'\| = \begin{cases} g(b' \boldsymbol{a}'^T \boldsymbol{w}), & \text{if } \|\boldsymbol{w}\|_* \leq \lambda / M \\ +\infty, & \text{otherwise} \end{cases}. \tag{75}$$

Proof. Since g is assumed to be proper, convex, and lower semicontinuous, by the Fenchel-Moreau theorem, it is equal to its biconjugate. Applying this property, we have

$$\begin{aligned} & \sup_{\boldsymbol{a} \in \mathbb{R}^d} \quad \{ g(b' \boldsymbol{a}^T \boldsymbol{w}) - \lambda \|\boldsymbol{a} - \boldsymbol{a}'\| \} \\ = & \sup_{\substack{\boldsymbol{a} \in \mathbb{R}^d, \\ \theta \in \mathrm{dom}(g^*)}} \quad \big\{ \theta b' \boldsymbol{a}^T \boldsymbol{w} - g^*(\theta) - \lambda \|\boldsymbol{a} - \boldsymbol{a}'\| \big\} \\ = & \sup_{\substack{\boldsymbol{a} \in \mathbb{R}^d, \\ \theta \in \mathrm{dom}(g^*)}} \quad \big\{ \theta b' (\boldsymbol{a} - \boldsymbol{a}')^T \boldsymbol{w} + \theta b' (\boldsymbol{a}')^T \boldsymbol{w} - g^*(\theta) - \lambda \|\boldsymbol{a} - \boldsymbol{a}'\|. \big\} \end{aligned}$$

Applying the change of variable $\boldsymbol{v} := \boldsymbol{a} - \boldsymbol{a}'$, we further have

$$\begin{aligned} & \sup_{\boldsymbol{a} \in \mathbb{R}^d} \quad \{ g(b' \boldsymbol{a}^T \boldsymbol{w}) - \lambda \|\boldsymbol{a} - \boldsymbol{a}'\| \} \\ = & \sup_{\substack{\boldsymbol{v} \in \mathbb{R}^d, \\ \theta \in \mathrm{dom}(g^*)}} \quad \big\{ \theta b' \boldsymbol{v}^T \boldsymbol{w} + \theta b' (\boldsymbol{a}')^T \boldsymbol{w} - g^*(\theta) - \lambda \|\boldsymbol{v}\| \big\} \\ = & \sup_{\theta \in \mathrm{dom}(g^*)} \big\{ \mathbb{1}_{\{\|\theta b' \boldsymbol{w}\|_* \leq \lambda\}} + \theta b' (\boldsymbol{a}')^T \boldsymbol{w} - g^*(\theta) \big\} \\ = & \begin{cases} g(b' (\boldsymbol{a}')^T \boldsymbol{w}), & \text{if } \sup_{\theta \in \mathrm{dom}(g^*)} \|\theta b' \boldsymbol{w}\|_* \leq \lambda \\ +\infty, & \text{otherwise} \end{cases}, \end{aligned}$$

where the second equality is by the convex conjugate of a norm $\|\cdot\|$ being equal to the convex indicator of the unit ball w.r.t. the dual norm $\|\cdot\|_*$. Finally, it remains to use that, by assumption, $\sup_{\theta \in \mathrm{dom}(g^*)} |\theta| = M$. □

Then following [52, Theorem 1], we provide reformulations of the problem from Eq. (11) that can be addressed by computationally efficient solvers.

Theorem 4. *The optimization problem from Eq.* (11) *is equivalent to:*

$$\begin{aligned}
\min_{\boldsymbol{w},\lambda,\boldsymbol{u},\boldsymbol{v},\boldsymbol{s},\boldsymbol{t}} \quad & \rho\lambda + \frac{1}{n}\sum_{i=1}^{n} s_i \\
\text{s.t.} \quad & u_i = b_i \boldsymbol{a}_i^T \boldsymbol{w}, && i \in [n], \\
& v_i = -u_i, && i \in [n], \\
& t_i = 2\kappa\lambda + s_i, && i \in [n], \\
& g(u_i) \le s_i, && i \in [n], \\
& g(v_i) \le t_i, && i \in [n], \\
& \|\boldsymbol{w}\|_* \le \lambda/M.
\end{aligned} \tag{76}$$

Proof. Let $\boldsymbol{z} = (\boldsymbol{a}, b) \in \Theta := \mathbb{R}^d \times \{1, -1\}$ and let $h_{\boldsymbol{w}}(\boldsymbol{z}) := g(b\boldsymbol{a}^T\boldsymbol{w})$. Then by the definition of the Wasserstein metric,

$$\sup_{\mathbb{P}\in\mathcal{P}_{\rho,\kappa}} \mathbb{E}^{\mathbb{P}}[g(b\boldsymbol{a}^T\boldsymbol{w})] = \begin{cases} \sup_{\pi\in\Pi(\mathbb{P},\hat{\mathbb{P}}_n)} & \int_\Theta h_{\boldsymbol{w}}(\boldsymbol{z})\pi(\mathrm{d}\boldsymbol{z},\Theta) \\ \text{s.t.} & \int_{\Theta\times\Theta} \zeta(\boldsymbol{z},\boldsymbol{z}')\pi(\mathrm{d}\boldsymbol{z},\mathrm{d}\boldsymbol{z}') \le \rho. \end{cases} \tag{77}$$

Assume that the conditional distribution of $\boldsymbol{z}$ given $\boldsymbol{z}' = (\boldsymbol{a}_i, b_i)$ is $\mathbb{P}^i$, for all $i \in [n]$. Then, based on the definition of $\mathbb{P}_n = \frac{1}{n}\sum_{i=1}^n \delta_{(\boldsymbol{a}_i,b_i)}$, we have

$$\pi(\mathrm{d}\boldsymbol{z},\mathrm{d}\boldsymbol{z}') = \frac{1}{n}\sum_{i=1}^{n} \delta_{(\boldsymbol{a}_i,b_i)}\mathbb{P}^i(\mathrm{d}\boldsymbol{z}). \tag{78}$$

As a result, the problem from Eq. (77) is equivalent to

$$\sup_{\mathbb{P}\in\mathcal{P}_{\rho,\kappa}} \mathbb{E}^{\mathbb{P}}[g(b\boldsymbol{a}^T\boldsymbol{w})] = \begin{cases} \sup_{\mathbb{P}^i} & \frac{1}{n}\sum_{i=1}^n \int_\Theta h_{\boldsymbol{w}}(\boldsymbol{z})\mathbb{P}^i(\mathrm{d}\boldsymbol{z}) \\ \text{s.t.} & \frac{1}{n}\sum_{i=1}^n \int_\Theta \zeta(\boldsymbol{z},\boldsymbol{z}')\mathbb{P}^i(\mathrm{d}\boldsymbol{z}) \le \rho \\ & \int_\Theta \mathbb{P}^i(\mathrm{d}\boldsymbol{z}) = 1. \end{cases} \tag{79}$$

Then substituting in $\boldsymbol{z} = (\boldsymbol{a}, b)$, using that the domain of y is $\{1, -1\}$, and decomposing $\mathbb{P}^i$ into unnormalized measures $\mathbb{P}_{\pm 1}(\mathrm{d}\boldsymbol{a}) = \mathbb{P}^i(\mathrm{d}\boldsymbol{a}, \{b = \pm 1\})$ supported on $\mathbb{R}^d$, the RHS of Eq. (79) can be simplified to

$$\begin{aligned}
\sup_{\mathbb{P}^i_\pm} \quad & \frac{1}{n}\sum_{i=1}^{n}\int_{\mathbb{R}^d} \big(h_{\boldsymbol{w}}(\boldsymbol{a},1)\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + h_{\boldsymbol{w}}(\boldsymbol{a},-1)\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a})\big) \\
\text{s.t.} \quad & \frac{1}{n}\sum_{i=1}^{n}\int_{\mathbb{R}^d} \big(\zeta((\boldsymbol{a},1),(\boldsymbol{a}_i,\boldsymbol{b}_i))\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + \zeta((\boldsymbol{a},-1),(\boldsymbol{a}_i,\boldsymbol{b}_i))\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a})\big) \le \rho \\
& \int_{\mathbb{R}^d} \big(\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + \mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a})\big) = 1.
\end{aligned}$$

With the definition of the cost function $\zeta((\boldsymbol{a}, b), (\boldsymbol{a}', b')) = \|\boldsymbol{a} - \boldsymbol{a}'\| + \kappa|b - b'|$, it follows that

$$\begin{aligned}
& \frac{1}{n}\sum_{i=1}^{n}\int_{\mathbb{R}^d} \zeta((\boldsymbol{a},1),(\boldsymbol{a}_i,\boldsymbol{b}_i))\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + \zeta((\boldsymbol{a},-1),(\boldsymbol{a}_i,\boldsymbol{b}_i))\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) \\
= & \frac{1}{n}\int_{\mathbb{R}^d}\sum_{b_i=1}[\|\boldsymbol{a}-\boldsymbol{a}_i\|\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + \|\boldsymbol{a}-\boldsymbol{a}_i\|\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) + 2\kappa\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a})] \\
& + \frac{1}{n}\int_{\mathbb{R}^d}\sum_{b_i=-1}[\|\boldsymbol{a}-\boldsymbol{a}_i\|\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) + \|\boldsymbol{a}-\boldsymbol{a}_i\|\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + 2\kappa\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a})] \\
= & \frac{2\kappa}{n}\int_{\mathbb{R}^d}\Big(\sum_{b_i=1}\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) + \sum_{b_i=-1}\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a})\Big) + \frac{1}{n}\int_{\mathbb{R}^d}\sum_{i=1}^{n}\|\boldsymbol{a}-\boldsymbol{a}_i\|(\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) + \mathbb{P}^i_1(\mathrm{d}\boldsymbol{a})).
\end{aligned} \tag{80}$$

Thus, we have

$$\sup_{\mathbb{P}\in\mathcal{P}_{\rho,\kappa}} \mathbb{E}^{\mathbb{P}}[g(b\boldsymbol{a}^T\boldsymbol{w})] = \begin{cases} \sup\limits_{\mathbb{P}^i_{\pm 1}} \dfrac{1}{n}\sum\limits_{i=1}^n \displaystyle\int_{\mathbb{R}^d} \big(h_{\boldsymbol{w}}(\boldsymbol{a},1)\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + h_{\boldsymbol{w}}(\boldsymbol{a},-1)\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a})\big) \\ \text{s.t.} \quad \dfrac{2\kappa}{n}\displaystyle\int_{\mathbb{R}^d}\left(\sum_{b_i=1}\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) + \sum_{b_i=-1}\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a})\right) \\ \qquad\qquad + \dfrac{1}{n}\displaystyle\int_{\mathbb{R}^d}\sum_{i=1}^n \|\boldsymbol{a}-\boldsymbol{a}_i\|(\mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a}) + \mathbb{P}^i_1(\mathrm{d}\boldsymbol{a})) \le \rho \\ \qquad \displaystyle\int_{\mathbb{R}^d}\big(\mathbb{P}^i_1(\mathrm{d}\boldsymbol{a}) + \mathbb{P}^i_{-1}(\mathrm{d}\boldsymbol{a})\big) = 1. \end{cases}$$

The above problem w.r.t. $\mathbb{P}^i_{\pm 1}$ is an infinite-dimensional linear program with a finite number of constraints. By [53, Proposition 3.4], we get the following equivalent dual formulation:

$$\sup_{\mathbb{P}\in\mathcal{P}_{\rho,\kappa}} \mathbb{E}^{\mathbb{P}}[g(b\boldsymbol{a}^T\boldsymbol{w})] = \begin{cases} \min\limits_{\lambda, s_i} \rho\lambda + \dfrac{1}{n}\sum\limits_{i=1}^n s_i \\ \text{s.t.} \ \sup\limits_{\boldsymbol{a}\in\mathbb{R}^d} h_{\boldsymbol{w}}(\boldsymbol{a},1) - \lambda\|\boldsymbol{a}-\boldsymbol{a}_i\| - \lambda\kappa(1-b_i) \le s_i, \quad i\in[n] \\ \qquad \sup\limits_{\boldsymbol{a}\in\mathbb{R}^d} h_{\boldsymbol{w}}(\boldsymbol{a},-1) - \lambda\|\boldsymbol{a}-\boldsymbol{a}_i\| - \lambda\kappa(1+b_i) \le s_i, \ i\in[n] \\ \qquad \lambda \ge 0. \end{cases}$$

Then, recalling that $h_{\boldsymbol{w}}(\boldsymbol{a},\pm 1)$ is short for $g(\pm\boldsymbol{a}^T\boldsymbol{w})$, by Lemma 12, we have

$$\sup_{\boldsymbol{a}\in\mathbb{R}^d} h_{\boldsymbol{w}}(\boldsymbol{a},\pm 1) - \lambda\|\boldsymbol{a}-\boldsymbol{a}_i\| = \begin{cases} g(\pm\boldsymbol{a}_i^T\boldsymbol{w}), & \text{if } \sup_{\theta\in\mathrm{dom}(g^*)}\|\theta\boldsymbol{w}\|_* \le \lambda \\ +\infty, & \text{otherwise.} \end{cases}$$

Finally, the resulting reformulation is

$$\begin{aligned} \min_{\boldsymbol{w},\lambda,\boldsymbol{s}} \ & \rho\lambda + \frac{1}{n}\sum_{i=1}^n s_i \\ \text{s.t.} \quad & g(b_i\boldsymbol{a}_i^T\boldsymbol{w}) \le s_i, \quad i\in[n], \\ & g(-b_i\boldsymbol{a}_i^T\boldsymbol{w}) - 2\lambda\kappa \le s_i, \quad i\in[n], \\ & \sup_{\theta\in\mathrm{dom}(g^*)} \|\theta\boldsymbol{w}\|_* \le \lambda. \end{aligned}$$

Finally, recalling that, by assumption, $\sup_{\theta\in\mathrm{dom}(g^*)}|\theta| = M$, it follows that the constraint $\sup_{\theta\in\mathrm{dom}(g^*)}\|\theta\boldsymbol{w}\|_* \le \lambda$ is equivalent to $\|\boldsymbol{w}\|_* \le \lambda/M$. Meanwhile, by introducing $u_i = b_i\boldsymbol{a}_i^T\boldsymbol{w}, v_i = -u_i (i\in[n])$ and $s_i, t_i (i\in[n]$, we obtain Theorem 4. □

In Theorem 4, when we assume that the conic constraints $g(u)\le s$ and $\|\boldsymbol{w}\|_* \le \lambda/M$ in Eq. (76) admit efficient proximal operators, we can formulate this problem as (PD-GLP) and apply CLVR. The resulting complexity bounds are similar to those discussed above for the f-divergence formulation.

D Experiment Details

D.1 Comparison between the values of L and R

As described in Section 1, one of the main advantages of CLVR over previous methods is that the complexity of CLVR depends on the max row norm R instead of the spectral norm L, which in the worst case for ill-conditioned problems can have a factor of $\sqrt{n}$ improvement. We claimed that in practical problems where the problem instances are highly structured (e.g., reformulated DRO problems), R can be much smaller than L and in Table 3 we provide empirical evidence for such claims. In all our experiments, we normalize each rows of $\boldsymbol{A}$ to $R = 1$ as stated in Assumption 1 so the values of L demonstrate the theoretical improvements for the experiments described in Section 5.

Table 3: Values of the spectral norm L in the reformulated DRO problems with Wasserstein metric after each row is normalized to $R = 1$.

Reformulated a9a $d = 130738, n = 97929$	Reformulated gisette $d = 44002, n = 28000$	Reformulated rcv1 $d = 269914, n = 155198$	Reformulated news20 $d = 5500750, n = 2770370$
117.3	65.9	196.4	1041.6

D.2 Comparison of adaptive restart schemes

We now provide a brief empirical comparison between our adaptive restart scheme that uses LP metric and the adaptive restart scheme using the normalized duality gap proposed in [9]. We compared the performance of PDHG on benchmark problem sets `qap10`, `qap15`, `nug08`, and `nug20` used in [9], using the two adaptive restart criteria. We ran PDHG until reaching accuracy as described in [9] (i.e., until normalized duality gap is at most 1e-6 and primal & dual infeasibility is at most 1e-8).

Table 4: Number of iterations required for the normalized duality gap and primal & dual infeasiblity to fall below 1e-6 and 1e-8 respectively.

Problem Name	Adaptive Normalized Duality Gap	Adaptive LP Metric
`qap10`	**13041**	14521
`qap15`	12561	**961**
`nug08`	**841**	1481
`nug20`	22001	**16281**

As can be seen from Table 4, the two restart criteria give similar performance in terms of iteration complexity: there is no clear winner, as normalized gap is better on `qap10` and `nug08`, while LP Metric is better on `qap15` and `nug20`. On the other hand, as Figure 2 shows, the two adaptive restart schemes lead to similar performance of PDHG over iterations. A bottom line is that our restart criterion does not seem to hurt the performance in terms of iteration complexity. Further empirical evaluation will be carried out to compare the performance of two restart criteria in terms of wall clock time.

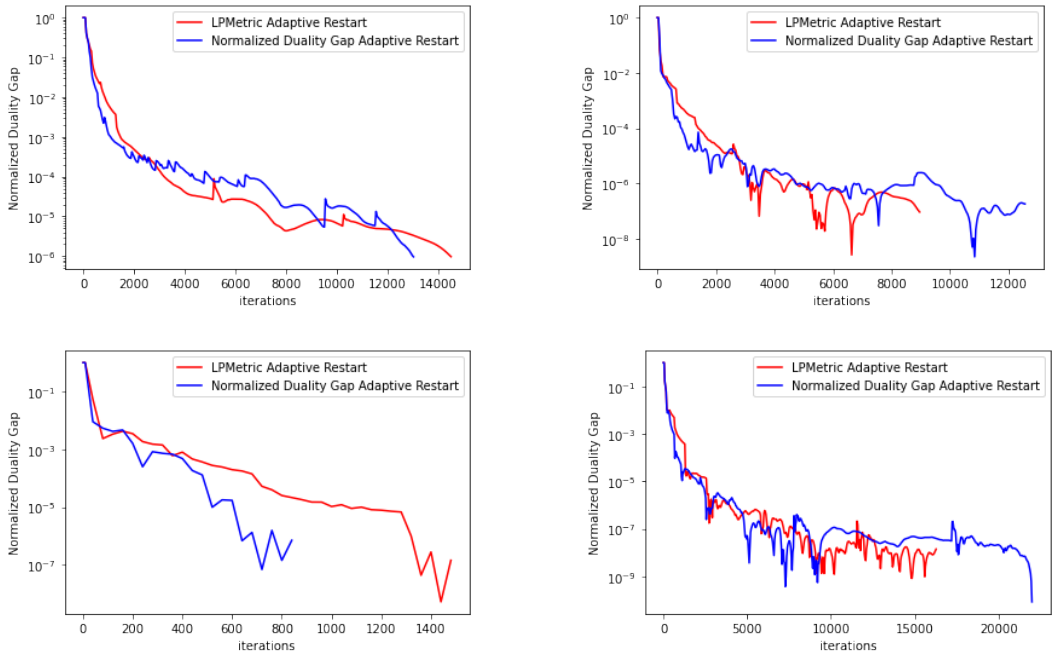

Figure 2: Comparisons of restart schemes that use LP Metric and that use the normalized duality gap. The plots from left to right and then top to bottom are for `qap10`, `qap15`, `nug08` and `nug20`.

D.3 Experiment Details

For the DRO with Wasserstein metric of ℓ_1-norm, since the dual norm of the ℓ_1 norm is the ℓ_∞ norm, the reformulation for this problem in Theorem 4 is a standard-form LP. For the LP formulation, we compare our CLVR method with three representative methods: PDHG [14], SPDHG [15] and PURE-CD [5]. For all algorithms we use LPMetric (7) as the performance measure and use a restart strategy of successive halving of LPMetric (Section 3.3) to obtain linear convergence. We implemented CLVR and other algorithms in Julia, optimizing all implementations to the best of our ability. For SPDHG, whose per-iteration cost is at least $O(d)$, we consider a large batch size 50 to balance the effect of the $O(d)$ cost and improve the overall efficiency. Meanwhile, PURE-CD with block size 1 is already well suited to sparsity. For CLVR, we experiment with block sizes 1 and 10. We conducted our experiments on LibSVM [16] datasets `a9a`, `gisette`, and `rcv1.binary`, each with different sparsity levels. For fair comparison in terms of wall-clock time, we implement all the algorithms in the Julia programming language[5] and run each algorithm using one CPU core, all the experiments are run on a Linux machine with a second generation Intel Xeon Scalable Processor (Cascade Lake-SP) with 128 GB of RAM. As the weight parameter γ (see Theorem 1) between primal and dual variables strongly influences the empirical performance, we tune it for all datasets by trying the values $\{10^{-i}\}$ for $i \in \mathbb{Z}$, for each of the methods. We set the Lipschitz constant of PDHG to be the largest singular value of the constraint matrix in the LP formulation. For PURE-CD and CLVR with block size 1, because the rows of the matrix are normalized, we set the Lipschitz constant to 1. For CLVR with block size 10 and SPDHG with block size 50, the Lipschitz constants are tuned to 3 and 9, respectively.

Remark 3 (Comparisons of using different block sizes)**.** *We conducted experiments to compare the practical performance of* CLVR *against different choices of block sizes and the results are shown in Table 3. We ran the DRO with Wasserstein metric using the same setup as described in Section 5, on the* `rcv1` *dataset and using an early stopping criterion of* `LP Metric` *at* 10^{-1}*. In the plot, we can see that* CLVR *converges to an approximate solution fastest when the block size is set to* 10*. Therefore, this plot provides support for our choice of block size of* 10 *in Section 5. As illustrated in Figure 1,* CLVR *is most efficient in terms of the number of data passes when the block size is 1. However, we observe that when evaluated in terms of the execution time, running* CLVR *with larger block size yields better performance even though it exhibits worse performance in terms of the number of data passes. We attribute this phenomenon to the instructional-level parallelism [28] in modern processors, allowing more computations to be completed in the same number of clock cycles.*

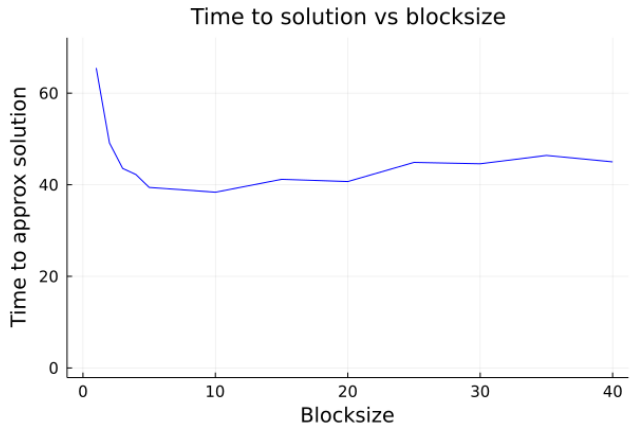

Figure 3: Comparison of the performance of CLVR with various choices of blocksizes.

In Table 5, we list information about the three datasets and the corresponding matrices in the reformulations. As we see, due to the sparse connectivity of auxiliary variables, all the matrices in reformulations are quite sparse. As a commonly adopted preprocessing step for LP, we normalize each row of the matrix in the standard-form LP with Euclidean norm one.

Remark 4 (Performance comparison using multiple cores)**.** *We conducted further experiments to examine the effects of allowing the algorithms to run on more computing cores. However, we did not*

[5] Julia is particularly designed for high performance numerical computation.

Table 5: The dimension and sparsity of the original datasets and the corresponding matrices in reformulations.

Dataset	Original (d, n)	#nonzeros / $(d \times n)$	Reformulated (d, n)	#nonzeros / $(d \times n)$
`a9a`	$(123, 32561)$	0.11	$(130738, 97929)$	9.6×10^{-5}
`gisette`	$(5000, 6000)$	0.99	$(44002, 28000)$	4.9×10^{-2}
`rcv1`	$(47236, 20242)$	1.5×10^{-3}	$(269914, 155198)$	8.8×10^{-5}

observe any meaningful difference in performances in terms of wall-clock time when we repeated the experiments described above and in Section 5 using 2 *CPU cores per algorithm. Our interpretation of this observation is that because most of the steps within* CLVR *and other algorithms we are comparing against are simple and cheap, involving very few large matrix-vector multiplications and no matrix factorization, the practical performance of algorithms becomes memory-bound, hence additional computational resources do not benefit them much if at all.*