LOWER-LEVEL DUALITY BASED PENALTY METHODS FOR HYPERPARAMETER OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Hyperparameter optimization (HO) is essential in machine learning and can be structured as a bilevel optimization. However, many existing algorithms designed for addressing nonsmooth lower-level problems involve solving sequential subproblems with high complexity. To tackle this challenge, we introduce penalty methods for solving HO based on strong duality between the lower level problem and its dual. We illustrate that the penalized problem closely approximates the optimal solutions of the original HO under certain conditions. In many real applications, the penalized problem is a weakly-convex objective with proximal-friendly constraints. Furthermore, we develop two fully first-order algorithms to solve the penalized problems. Theoretically, we prove the convergence of the proposed algorithms. We demonstrate the efficiency and superiority of our method across numerical experiments.

1 INTRODUCTION

In machine learning, the introduction of regularization terms is a common practice aimed at enhancing model generalization and controlling model complexity. This overarching framework can be articulated as an objective function that strikes a balance between data fitting and model simplicity:

$$\min_{\mathbf{x}} \; l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{x}). \tag{1}$$

In this formulation, $l(\mathbf{x})$ represents the loss function and $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_r)$ encompasses hyperparameters, which are not derived from the learning algorithm but rather specified as inputs. Meanwhile, $R_i(\mathbf{x}), i = 1, 2, ..., r$ denotes the regularizers, which are considered in the form of norms in this paper, i.e. $R_i(\cdot) = \|\cdot\|$. The pursuit of optimal hyperparameters that enhance predictive performance is a vital task in machine learning, commonly referred to as hyperparameter optimization (Feurer & Hutter, 2019; Gao et al., 2022; Ye et al., 2021; 2023; Chen et al., 2024). In supervised learning, this process involves partitioning the dataset into training, validation, and test sets, solving (1) for various $\boldsymbol{\lambda}$ values, and selecting the best $(\boldsymbol{\lambda}, \mathbf{x}_{\boldsymbol{\lambda}})$ based on validation and training error. The quality of the selected hyperparameters is ultimately evaluated through the test error function. This structured approach can be encapsulated within a bilevel optimization framework (Dempe & Zemkoho, 2020):

$$\begin{aligned} \min_{\mathbf{x}_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}} \quad & L(\mathbf{x}_{\boldsymbol{\lambda}}) \\ \text{s.t.} \quad & \mathbf{x}_{\boldsymbol{\lambda}} \in \arg\min_{\mathbf{x}} \left\{ l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{x}) \right\}. \end{aligned} \tag{2}$$

In this formulation, L serves as the loss function on the validation set, defining the upper-level (UL) problem, while l represents the training set loss function, constituting the lower-level (LL) problem alongside the regularization terms. The hyperparameters $\boldsymbol{\lambda}$ help delineate the trade-off between fitting the data and maintaining simplicity.

1.1 MAIN CONTRIBUTIONS

We summarize our main contributions as follows. We propose a penalty method based on lower-level duality for hyperparameter optimization (2), which is in the form of bilevel optimization with nonsmooth lower-level problem. Our method avoids any implicit value functions and high-complexity subproblems. Additionally, we introduce first-order algorithms to solve the penalization problem and provide theoretical proof of its convergence. Through experimental results, we demonstrate the superiority of our algorithm, highlighting its independence from any convex optimization solvers while showcasing its exceptional efficiency.

1.2 Related Work

Hyperparameters Optimization. The existing literature presents various strategies for hyperparameter selection. Among the simplest model-free techniques are grid search (Injadat et al., 2020) and random search (Bergstra & Bengio, 2012). Additionally, Bayesian optimization (Bergstra et al., 2011; Snoek et al., 2012) serves as a sequential algorithm that selects future evaluation points by leveraging insights from prior outcomes. However, these gradient-free methods face significant challenges when dealing with a high number of parameters. To address this limitation, Feng & Simon (2018) introduces gradient-based techniques for hyperparameter tuning.

Bilevel Optimization. In general, the problem presented in (2) aligns with the format known as bilevel optimization (BLO), which is pertinent to a diverse array of data-driven challenges, including hyperparameter optimization (Maclaurin et al., 2015; Franceschi et al., 2018), meta-learning (Finn et al., 2017), and reinforcement learning (Shen et al., 2024; Stadie et al., 2020).

The initial strategies for addressing bilevel optimization problems primarily centered on gradient-based algorithms, which can be broadly classified into two categories based on their methods for computing hypergradients. Iterative Differentiation (ITD) involves unrolling the lower-level problem into gradient steps and subsequently utilizing backpropagation to calculate the hypergradient (Franceschi et al., 2017; 2018; Grazzi et al., 2020; Liu et al., 2021b; Antoniou et al., 2018; Shaban et al., 2019). In contrast, Implicit Differentiation (AID) leverages the first-order optimality conditions of the lower-level problem along with the implicit function theorem to derive the hypergradient (Pedregosa, 2016; Rajeswaran et al., 2019; Lorraine et al., 2020; Yang et al., 2021; 2023). However, these methods necessitate the strong convexity of the lower-level problem, thereby constraining their applicability.

Recently, Chen et al. (2023a); Li et al. (2022); Chen et al. (2023b) have introduced a series of fully first-order methods that operate without requiring Hessian computations or implicit gradients. Additionally, many machine learning problems may exhibit multiple minima for the lower-level function. To address this challenge, Liu et al. (2021a) propose a value function based on the optimal value of the lower-level function, which leads to the development of novel algorithms employing a penalization technique (Liu et al., 2023). As a result, penalty-based methods have also emerged as effective solutions for bilevel optimization problems. Shen & Chen (2023); Lu & Mei (2024); Kwon et al. (2023b;a); Liu et al. (2022) construct single-level reformulation for original BLO by penalty method with various penalty terms.

Nonsmoooth Lower-level Problem. When the regularazer is l_1 norm, Bertrand et al. (2020) proposes an implicit differentiation method with block coordinate descent for Lasso-type hyperparameter optimization, later extended to general nonsmooth problems Bertrand et al. (2022). Ye et al. (2021; 2023) utilize diffenrence-of-convex (DC) method for hyperparameter selection, while Gao et al. (2022) combine penalization with DC method for bilevel problems with nonsmooth regularizer. Both methods require computing the lower-level optimal value for subgradients. Recently, Chen et al. (2023a) propose an inexact gradient-free method, though the subproblem remains difficult to solve. Chen et al. (2024) presents a novel reformulation based on LL duality with no value function involved and proposes an iterative algorithm grounded in cone programming for many pratical applications alongside its corresponding off-the-shelf solver. Recent studies have also employed the Moreau envelope to effectively address nonsmooth functions. Works by Gao et al. (2023); Yao et al. (2024b); Liu et al. (2024) have restructured the original bilevel optimization framework using this strategy and propose a series of Moreau envelope-based algorithms, which demonstrate the capability to identify well-defined KKT points.

2 Penalization Framework

In this section, we introduce our **l**ower-level **d**uality based **p**enalty **m**ethod (LDPM) for hyperparameter optimization (2). We begin by separating and simplifying the hierarchical structure of the lower-level problem using Fenchel duality. Unlike traditional primal-dual methods, we employ conjugate functions to transform the subproblems into constrained optimization problems, eliminating the need for any value function. Subsequently, we implement the penalization strategy and discuss the relationship between the penalized formulation and the original problem (2).

2.1 Penalty-based Methods Based on Lower-level Duality

In this subsection, we reconstruct the lower-level problem with Lagrangian function and duality. Based on this, we study the lower-level duality reformulation and propose the penalty-based method. First we introduce augmented variables $\mathbf{z}_i, i = 1, 2, ..., r$ and deduce the equivalent form of LL problem of (2),

$$\min_{\mathbf{x},\mathbf{z}_i} l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{z}_i) \quad \text{s.t. } \mathbf{x} = \mathbf{z}_i. \tag{3}$$

Since l, R_i are convex and the constraints are affine, strong duality holds under Slater's condition. If $\text{ri}(\text{dom } l \cap (\cap_{i=1}^{r} \text{dom } R_i)) \neq \emptyset$, then (3) is equivalent to its Lagrangian dual problem:

$$-\min_{\boldsymbol{\rho}} \max_{\mathbf{x},\mathbf{z}_i} -l(\mathbf{x}) - \sum_{i=1}^{r} \lambda_i R_i(\mathbf{z}_i) - \sum_{i=1}^{r} \boldsymbol{\rho_i}^T (\mathbf{x} - \mathbf{z}_i),$$

where $\boldsymbol{\rho}_i$ is are Lagrangian multipliers associated with constraint $\mathbf{x} = \mathbf{z}_i$. The above problem can be further simplified with definition of conjugate functions as,

$$\max_{\boldsymbol{\rho}} -l^*(-\sum_{i=1}^{r} \boldsymbol{\rho_i}) - \sum_{i=1}^{r} \lambda_i R_i^*(-\frac{\boldsymbol{\rho}_i}{\lambda_i}). \tag{4}$$

Meanwhile, the constraint of (2) is equivalent to

$$\begin{aligned} l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{x}) &\overset{(a)}{\leq} \min_{\mathbf{x}} \{ l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{x}) \} \\ &\overset{(b)}{=} \max_{\boldsymbol{\rho}} -l^*(-\sum_{i=1}^{r} \boldsymbol{\rho_i}) - \sum_{i=1}^{r} \lambda_i R_i^*(-\frac{\boldsymbol{\rho}_i}{\lambda_i}), \end{aligned} \tag{5}$$

where, (a) utilizes the value function of the lower-level problem, which is widely used in relevant literature of BLO Liu et al. (2021a; 2023), (b) is from the equivalence of (3)-(4). Dropping the $\max$ operator, we obtain that the lower-level problem of (2) can be replaced by the inequality constraint,

$$l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{x}) + l^*(-\sum_{i=1}^{r} \boldsymbol{\rho}_i) + \sum_{i=1}^{r} \lambda_i R_i^*(\frac{\boldsymbol{\rho_i}}{\lambda_i}) \leq 0,$$

and obtain the reformulation for (2):

$$\begin{aligned} \min_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho}} \quad & L(\mathbf{x}) \\ \text{s.t.} \quad & l(\mathbf{x}) + \sum_{i=1}^{r} \lambda_i R_i(\mathbf{x}) + l^*(-\sum_{i=1}^{r} \boldsymbol{\rho}_i) + \sum_{i=1}^{r} \lambda_i R_i^*(\frac{\boldsymbol{\rho_i}}{\lambda_i}) \leq 0. \end{aligned} \tag{6}$$

Note that it is independent of any implicit value function, but rather utilizes the conjugate of the atom functions in the lower-level problem. Naturally, the validity of (6) depends on the following assumption.

Assumption 2.1. l and $R_i, i = 1, 2, ..., r$ in the lower-level problem of (2) possess explicit conjugate functions.

The fulfillment of Assumption 2.1 is straightforward to ensure. Indeed, the loss functions in most real-world problems have closed-form conjugate functions, including least squares, hinge loss and logarithmic functions. Similarly, the norm terms $R_i(\cdot)$ also share this property, where we denote $R_i^*(\cdot) = \| \cdot \|^*$ as the conjugate norm of R_i. In this case, we observe that $R_i^*(\frac{\boldsymbol{\rho_i}}{\lambda_i}) = 0$ provided the condition $\|\boldsymbol{\rho}_i\|_* \leq \lambda_i$ holds (Boyd & Vandenberghe, 2004). Meanwhile, with introducing an auxiliary variables r_i satisfying $R_i(\mathbf{x}) \leq r_i$, the constraint of (6) is equivalent to

$$\begin{aligned} & l(\mathbf{x}) + l^*(-\sum_{i=1}^{r} \boldsymbol{\rho}_i) + \sum_{i=1}^{r} \lambda_i r_i \leq 0. \\ & R_i(\mathbf{x}) \leq r_i, \|\boldsymbol{\rho}_i\|_* \leq \lambda_i, i = 1, 2, ..., r. \end{aligned} \tag{7}$$

Consequently, (6) is equivalent to the following problem,

$$\begin{aligned} \min_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}} \quad & L(\mathbf{x}) \\ \text{s.t.} \quad & l(\mathbf{x}) + l^*(-\sum_{i=1}^{r} \boldsymbol{\rho}_i) + \sum_{i=1}^{r} \lambda_i r_i \leq 0. \\ & R_i(\mathbf{x}) \leq r_i, \|\boldsymbol{\rho}_i\|_* \leq \lambda_i, i = 1, 2, ..., r. \end{aligned} \tag{8}$$

We summarize the first inequality constraint of (8) as a penalty term

$$p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) = l(\mathbf{x}) + l^*(-\sum_{i=1}^{r} \boldsymbol{\rho}_i) + \sum_{i=1}^{r} \lambda_i r_i, \tag{9}$$

and employ penalization strategy to handle (8). Then we can rewrite (8) with a penalty constant β as follows,

$$\begin{aligned} \min_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}} \quad & L(\mathbf{x}) + \beta p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}). \\ \text{s.t.} \quad & R_i(\mathbf{x}) \leq r_i, \|\boldsymbol{\rho}_i\|_* \leq \lambda_i, i = 1, 2, ..., r. \end{aligned} \tag{10}$$

Thus, we have fully converted the hyperparameter optimization (2) into a single-level formulation (10). Although the introduced variable $\boldsymbol{\rho}_i$ has the same dimension as $\mathbf{x}$, it does not affect the whole scale and complexity.

2.2 Equivalence Between Penalized and Primal Problem

In this subsection, we discuss the relationship between (2) and (10) from the perspective of duality. We first introduce corresponding assumptions for 2 as follows.

Assumption 2.2. $L(\mathbf{x})$ is L_0-Lipschitz continuous.

Assumption 2.3. $l(\mathbf{x})$ is $(1/\alpha_l)$-strongly convex and l_1-smooth.

Assumption 2.4. For any given $\mathbf{x}$, the optimal solution set of lower-level problem in (2) denoted as $L_{\text{opt}}(\boldsymbol{\lambda})$ is closed and non-empty.

Besides Assumption 2.2, we note that the norm terms $R_i(\mathbf{x})$ are convex but potentially nonsmooth, which implies that the lower-level problem is convex and nonsmooth in $\mathbf{x}$. Regarding Assumptions 2.2 and 2.3, the conjugate function l^* is α_l-smooth (Theorem 5.26 in Beck (2017)). Subsequently, the penalty term $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ is differentiable and $(l_1 + \alpha_l + 1)$-smooth. The above assumptions are prevalent and commonly satisfied in practical applications. From (3)-(8), we know that (2) can be reformulated into (8). From the KKT conditions of (3), we first analyze $\boldsymbol{\rho}_i, i = 1, 2, ..., r$ in (6) and obtain the following lemma.

Lemma 2.5. *If $\mathbf{x}_{\boldsymbol{\lambda}}$ is an optimal solution of the lower-level problem of (2), then there exists the unique multiplier $\boldsymbol{\rho}_i^*$ and $\mathbf{z}_i^* = \mathbf{x}_{\boldsymbol{\lambda}}$ such that $(\mathbf{x}_{\boldsymbol{\lambda}}, \mathbf{z}_i^*, \boldsymbol{\rho}_i^*)$ is a KKT point of (3).*

According to KKT condition of we recover that $\boldsymbol{\rho}_i^*$ in Lemma 2.5 satisfies that

$$\sum_{i=1}^{r} \boldsymbol{\rho}_i^* = -\nabla l(\mathbf{x}_{\boldsymbol{\lambda}}), \ \boldsymbol{\rho}_i^* \in \lambda_i \partial R_i(\mathbf{x}_{\boldsymbol{\lambda}}), i = 1, 2, ..., r, \tag{11}$$

which implies that the KKT point of (3) is also the stationary point of the lower-level problem of (2). Note that the penalty term $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ is derived from duality of lower-level problem, so we summarize the property of $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ regulating $\|\mathbf{x} - \mathbf{x}_{\boldsymbol{\lambda}}\|^2$ as follows.

Lemma 2.6. *Suppose Assumption 2.3 and 2.4 hold, then it holds that $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) \geq \frac{\alpha_l}{2}\|\mathbf{x} - \mathbf{x}_{\boldsymbol{\lambda}}\|^2 \geq 0$ for any given $\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}$. In addition, $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) = 0$ if and only if $\mathbf{x} \in L_{opt}(\boldsymbol{\lambda})$.*

Based on Lemma 2.5, we further derive the equivalence between bilevel form (2) and the constrained problem (6) as follows.

Proposition 2.7. *If $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is a global optimal solution for (2), and $\boldsymbol{\rho}_i^*$ is defined as in (11), then $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}_i^*)$ is global optimal solution for (6).*

From Proposition 2.7, we can further recognize the equivalence between the primal problem (2) and (8). As a result, we now redirect our focus to investigating relationship between (8) and (10).

Due to the non-negativity of the penalty term $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$, we find that there is no interior points in the feasible region of (6)(8), in the sense that the constraint contradicts any standard regularity condition. Therefore, we consider the following ϵ-approximate problem for (6)(8) and discuss the equivalence between it and the penalty problem (10),

$$\begin{array}{ll} \min\limits_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}} & L(\mathbf{x}) \\ \text{s.t.} & p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) \leq \epsilon. \\ & R_i(\mathbf{x}) \leq r_i, \|\boldsymbol{\rho}_i\|_* \leq \lambda_i, i = 1, 2, ..., r. \end{array} \tag{12}$$

Leveraging Lemma 2.6, we establish the relationship between global optimal solutions of (10) and 12 in Proposition 2.8, which is inspired by Shen & Chen (2023).

Proposition 2.8. *Suppose Assumption 2.3 and 2.4 hold. For any $\epsilon_p > 0$, the global optimal solution of (2) is also an ϵ_p-approximation optimal solution of the penalized problem (10) with $\beta > \beta^* = \frac{l_0^2 \alpha_l}{8\epsilon_p}$. Conversely, the ϵ_1-global solution of (10) with $\beta > \beta^*$ is a global optimal solution for ϵ-approximate problem (12) with $0 \leq \epsilon \leq (\epsilon_p + \epsilon_1)/(\beta - \beta^*)$.*

In summary, we confirm the relationship between the penalized problem (10) and primal problem (2). Subsequently, we illustrate the proximity between the optimal value of (10) and (2).

Theorem 2.9. *Suppose that Assumptions 2.2,2.3 and 2.4 hold. If $(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*)$ is ϵ-optimal solution of the penalized problem (10), then we obtain that $|L(\mathbf{x}_\epsilon^*) - L(\mathbf{x}^*)| \leq \mathcal{O}(\epsilon)$, where $\mathbf{x}^*$ with an optimal $\boldsymbol{\lambda}^*$ attains the minimum of (2).*

We provide the related proofs in Appendix A. The primary challenges in solving (10) arise from its nonsmooth and nonconvex properties. To address these, we explore first-order algorithms to solve the penalized problem (10), cleverly leveraging the structure of (2) and (10).

3 Solving the penalty formulations

In this section, we propose our main algorithm grounded in penalty-based problem (10). For convenience, we denote $\mathbf{z} = (\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$. We then introduce the constraint sets for each i as follows,

$$\mathcal{R}_i \stackrel{\triangle}{=} \{\mathbf{z} | R_i(\mathbf{x}) \leq r_i\}, \quad \mathcal{R}_i^* \stackrel{\triangle}{=} \{\mathbf{z} | \|\boldsymbol{\rho_i}\|_* \leq \lambda_i\}. \tag{13}$$

A natural approach to manage the constraints of (10) is through projection onto $\mathcal{R}_i$ and $\mathcal{R}_i^*$. To proceed, we introduce the following assumption regarding $\mathcal{R}_i$ and $\mathcal{R}_i^*$.

Assumption 3.1. For the constraint sets $\mathcal{R}_i, i = 1, 2, ..., r$, each individual set among these r sets can be easy to project, implying that the corresponding indicator functions $\mathcal{I}_{\mathcal{R}_i}(\mathbf{z})$ are proximal-friendly for each i, respectively.

From Moreau decomposition theorem (Theorem 6.44 in Beck (2017)), we know that each individual set $\mathcal{R}_i^*$ and corresponding indicator functions $\mathcal{I}_{\mathcal{R}_i^*}(\mathbf{z})$ satisfy the same property described in Assumption 3.1 for $\mathcal{R}_i$. Assumption 3.1 holds for common norm terms. Even if the constraints of (10) are in conic form, the corresponding projections still have close-form solutions for each i. We explain the specific analytic solutions of projection in Appendix C.

However, significant differences exist between the two groups of constraints related to norms and their conjugate, as the constraints $R_i(\mathbf{x}) \leq r_i$ are all related to the same variable $\mathbf{x}$ while the constraints $\|\boldsymbol{\rho}_i\|_* \leq \lambda_i$ pertain to entirely different variables $\boldsymbol{\rho}_i$. Consequently, the projection process for $\cap_{i=1}^r \mathcal{R}_i$ will involve complicated interactions among the feasible domain of each constraint $R_i(\mathbf{x}) \leq r_i$. In other words, the constraint sets R_i^* are mutually separated, which means that $\cap_{i=1}^r \mathcal{R}_i^*$ is easy to project. Accordingly, the projection onto $\cap_{i=1}^r \mathcal{R}_i$ is hard to directly computed and its indicator function is generally proximal-unfriendly.

Although relevant full projection algorithms for composite constraints are explored by Li et al. (2020); Liu & Liu (2017), these algorithms necessitate additional iterative loop and produce inexact results. Thus, the integration of these full projections with first-order algorithms can lead to divergence and a notable decrease in efficiency. Therefore, we need to consider splitting the mixed constraint sets $\cap_{i=1}^r \mathcal{R}_i$. In the specific scenario of problem (2) with a single regularizer, the obstacles are rendered unnecessary.

Therefore, we introduce the first-order algorithm for a single regularizer ($r = 1$) as a special case in subsection 3.1, while the algorithm for problems requiring multiple norm regularization terms ($r > 1$) is presented in subsection 3.2.

3.1 Single Regularization Term

In this subsection, we explore the algorithm for (2) with a single regularization term $R_1(\mathbf{x})$. Consequently, (10) simplifies to the following formulation:

$$\begin{aligned} \min_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}} \quad & L(\mathbf{x}) + \beta p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}). \\ \text{s.t.} \quad & R_1(\mathbf{x}) \le r_1, \|\boldsymbol{\rho}\|_* \le \lambda_1, \end{aligned} \tag{14}$$

where $p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) = l(\mathbf{x}) + l^*(-\boldsymbol{\rho}) + \lambda_1 r_1$. We adopt the notations $\mathbf{z} = (\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ and define $\mathcal{R}_1, \mathcal{R}_1^*$ as in (13).

Definition 3.2. A function f is called w-weakly convex for some $w \ge 0$ if $f(\cdot) + \frac{w}{2}\|\cdot\|^2$ is convex.

It is noteworthy that the bilinear term $\lambda_1 r_1$ is 1-weakly convex and 1-smooth with respect to $\mathbf{z}$.

Lemma 3.3. *$L(\mathbf{x}) + \beta p(\mathbf{z})$ is l_p-smooth in $\mathbf{z}$ with $l_p \overset{\triangle}{=} l_1 + \beta(l_1 + \alpha_l + 1)$.*

The above results can be directly computed under Assumptions 2.2 and 2.3. Meanwhile, the sets $\mathcal{R}_1$ satisfies Assumption 3.1 and it is separated from $\mathcal{R}_1^*$. Therefore, $\mathcal{R}_1 \cap \mathcal{R}_1^*$ is projected-friendly and (14) can be minimized with projected gradient descent. We summarize our first-order algorithm for (14) in Algorithm 1. In line 1, $\mathbf{x}^0$ is initialized by solving lower-level problem $\min_{\mathbf{x}}\{l(\mathbf{x}) + \lambda_1 R_1(\mathbf{x})\}$ with given λ_1^0 and we set $\mathbf{r}^0 = R_1(\mathbf{x}^0), \boldsymbol{\rho}^0 = -\nabla l(\mathbf{x}^0)$. In this setting, we ensure the feasibility of problem (14). In line 3, the iterative first-order method is performed for problem (14) accompanied by the projection onto $\mathcal{R}_1 \cap \mathcal{R}_1^*$. With the fixed penalty parameter β, we set the step size $\eta \le 2/l_p$ and l_p is computed in Lemma 3.3, which ensures consistent progression throughout the iterations. In line 4, we choose the stopping criterion with the results of two iterative points are sufficiently close, i.e., $\|\mathbf{z}^{k+1} - \mathbf{z}^k\| \le \mathrm{tol}$.

Algorithm 1 First-order Methods for Penalized Problem (14)

1: Initialize $\boldsymbol{\lambda}^0$ and $\mathbf{x}^0, \boldsymbol{\rho}^0, \mathbf{r}^0$, constants β, η.
2: **for** $k = 0, 1, 2, ..., K$ **do**
3:     Update $\mathbf{z}^{k+1} = \mathrm{proj}_{\mathcal{R}_1 \cap \mathcal{R}_1^*}\{\mathbf{z}^k - \eta[\nabla_{\mathbf{z}}(L(\mathbf{x}^k) + \beta p(\mathbf{z}^k))]\}$.
4:     **if** Termination criteria is met. **then**
5:         Stop.
6:     **end if**
7: **end for**

Remark 3.4. We define an indicator function as $g_1(\mathbf{z}) = \mathcal{I}_{\mathcal{R}_1 \cap \mathcal{R}_1^*}(\mathbf{z})$. The iteration 3 in Algorithm 1) can be described as the process of finding an approximate optimal solution of (14).

Since the reformulation (6) involves no implicit value functions related to the lower-level problem of (2), Algorithm 1 does not require an iterative loop for finding the optimal solution $\mathbf{x}_{\boldsymbol{\lambda}}$ of lower-level problem of (2) or the dual multiplier $\boldsymbol{\rho}^*$. Therefore, Algorithm 1 is equipped with a single loop for $\mathbf{z}$, which fully centers on the variables $(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ in problem (14).

In this case, we obtain the sufficient decrease and convergence results of Algorithm 1 as follows.

Lemma 3.5. *Assume $L(\mathbf{x})$ and $p(\mathbf{z})$ are bounded below. For $k \in \mathbb{N}$ and $\{\mathbf{z}^k\}$ generated from Algorithm 1 with penalty parameter $\bar{\beta}$, we have $L(\mathbf{x}^{k+1}) + \bar{\beta} p(\mathbf{z}^{k+1}) \le L(\mathbf{x}^k) + \bar{\beta} p(\mathbf{z}^k)$. In addition, the sequence $\{\mathbf{z}^k\}$ satisfies that $\lim_{k\to\infty} \|\mathbf{z}^{k+1} - \mathbf{z}^k\| = 0$.*

Theorem 3.6. *Assume $L(\mathbf{x})$ and $p(\mathbf{z})$ are bounded below. Based on Lemma 3.5, any limit point of $\{\mathbf{z}^k\}$ is a stationary point of (14).*

The proofs of Lemma 3.5 and Theorem 3.6 are provided in Appendix B. The convergence results in this case follow from Beck & Teboulle (2009; 2010), which introduce the analysis of proximal gradient method. In summary, Algorithm 1 addresses the primal problem (2) with single regularization term by applying the penalized problem in the form of (14). It also inspires the resolution of the cases involving multiple regularization terms.

3.2 Double Regularization Terms

In this subsection, we focus on the algorithm design for (2) involving multiple regularization terms. For convenience, we present the case with double regularization terms in the main text, while the algorithm for addressing (2) with more regularization terms and correspondingly results are provided in Appendix B.5. For this scenario, (10) simplifies to the following formulation:

$$\begin{array}{ll}\min\limits_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}} & L(\mathbf{x})+\beta p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}). \\ \text{s.t.} & R_i(\mathbf{x})\le r_i, \|\boldsymbol{\rho}_i\|_*\le \lambda_i, i=1,2,\end{array} \tag{15}$$

where $p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})=l(\mathbf{x})+l^*(-\boldsymbol{\rho}_1-\boldsymbol{\rho}_2)+\lambda_1 r_1+\lambda_2 r_2$. We adopt the notations $\mathbf{z}=(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})$ and $\mathcal{R}_i,\mathcal{R}_i^*, i=1,2$ defined in (13). From Assumption 3.1, we know that $\mathcal{R}^*\stackrel{\triangle}{=}\mathcal{R}_1^*\cap\mathcal{R}_2^*$ is projected-friendly, so we merely need to perform variable decomposition for $\mathcal{R}_1\cap\mathcal{R}_2$. We define $g_i(\mathbf{z})\stackrel{\triangle}{=}\mathcal{I}_{\mathcal{R}_i\cap\mathcal{R}^*}(\mathbf{z}), i=1,2$. Under this conditions, (15) can be rewritten as the following equivalent form,

$$\min_{\mathbf{z}} \quad L(\mathbf{x})+\beta p(\mathbf{z})+g_1(\mathbf{z})+g_2(\mathbf{z}). \tag{16}$$

Motivated by (3), we introduce an auxiliary variable $\mathbf{u}$ as follows,

$$\begin{array}{ll}\min\limits_{\mathbf{z}} & L(\mathbf{x})+\beta p(\mathbf{z})+g_1(\mathbf{z})+g_2(\mathbf{u}) \\ \text{s.t.} & \mathbf{z}=\mathbf{u}.\end{array} \tag{17}$$

The augmented Lagrangian function of problem (17) is

$$\begin{aligned}\mathcal{L}_\gamma(\mathbf{z},\mathbf{u},\boldsymbol{\mu}) &= L(\mathbf{x})+\beta p(\mathbf{z})+g_1(\mathbf{z})+g_2(\mathbf{u})+\langle\boldsymbol{\mu},\mathbf{u}-\mathbf{z}\rangle+\frac{\gamma}{2}\|\mathbf{u}-\mathbf{z}\|^2 \\ &= L(\mathbf{x})+\beta p(\mathbf{z})+g_1(\mathbf{z})+g_2(\mathbf{u})+\frac{\gamma}{2}\|\mathbf{u}-\mathbf{z}+\frac{\boldsymbol{\mu}}{\gamma}\|^2-\frac{\|\boldsymbol{\mu}\|^2}{2\gamma}.\end{aligned}$$

Now, we naturally employ Alternating Direction Method of Multipliers (ADMM) to solve (17), which cyclically update $\mathbf{u},\mathbf{z},\boldsymbol{\mu}$ by solving the $\mathbf{u}$- and $\mathbf{z}$-subproblems and adopt a dual ascent step for $\boldsymbol{\mu}$. We summarize the iterations in Algorithm 2. In line 1, $\mathbf{x}^0$ is initialized by solving lower-level problem $\min_{\mathbf{x}}\{l(\mathbf{x})+\lambda_1 R_1(\mathbf{x})+\lambda_2 R_2(\mathbf{x})\}$ with given $\boldsymbol{\lambda}^0$ and we set $\mathbf{r}_i^0=R_i(\mathbf{x}^0)$. In line 3, we add a proximal term due to the weakly-convex term $\lambda_i r_i, i=1,2$ with a constant t. In line 4, $\mathbf{u}$-subproblem takes the form of direct projection onto $\mathcal{R}_2$. Under Assumption 3.1, we assume that $\mathbf{u}$-subproblem can be solved exactly in each iteration.

Algorithm 2 ADMM Framework for Problem (15)

1: Initialize $\boldsymbol{\lambda}^0$ and $\mathbf{x}^0,\boldsymbol{\rho}^0,\mathbf{r}^0,\mathbf{u}^0=(\mathbf{x}^0,\boldsymbol{\lambda}^0,\boldsymbol{\rho}^0,\mathbf{r}^0)$, constants β,γ and t.
2: **for** $k=0,1,2,...$ **do**
3: $\quad\mathbf{z}^{k+1}=\arg\min_{\mathbf{z}}\left\{L(\mathbf{x})+\beta p(\mathbf{z})+g_1(\mathbf{z})+\frac{\gamma}{2}\|\mathbf{u}^k-\mathbf{z}+\frac{\boldsymbol{\mu}^k}{\gamma}\|^2+\frac{t}{2}\|\mathbf{z}-\mathbf{z}^k\|^2\right\}$.
4: $\quad\mathbf{u}^{k+1}=\arg\min_{\mathbf{u}}\left\{g_2(\mathbf{u})+\frac{\gamma}{2}\|\mathbf{u}-\mathbf{z}^{k+1}+\frac{\boldsymbol{\mu}^k}{\gamma}\|^2\right\}$.
5: $\quad\boldsymbol{\mu}^{k+1}=\boldsymbol{\mu}^k+\gamma(\mathbf{u}^{k+1}-\mathbf{z}^{k+1})$.
6: **end for**

According to Definition 3.2, we control the proximal coefficient with $t>\alpha_d-\gamma$ where $\alpha_d\stackrel{\triangle}{=}\frac{\beta}{2}-(1+\beta)\alpha_l-\gamma$, then we describe the property of $\mathbf{z}$-subpoblem in the following lemma.

Lemma 3.7. *Suppose Assumptions 2.2 and 2.3 hold. The $\mathbf{z}$-subproblem in line 3 of Algorithm 2 enjoys ($t-\alpha_d$)-strongly convex property, while the objective function is l_d-smooth with $l_d\stackrel{\triangle}{=}\gamma+t+l_1+\beta(l_1+\alpha_l+1)$.*

The above results is obtained from direct computation under Assumptions 2.2 and 2.3. For $\mathbf{z}$-subproblem in line 3, $g_1(\mathbf{z})$ is indicator function and the problem can be expressed in the following form

$$\mathbf{z}^{k+1}=\arg\min_{\mathbf{z}\in\mathcal{R}_1\cap\mathcal{R}^*}\left\{L(\mathbf{x})+\beta p(\mathbf{z})+\frac{\gamma}{2}\|\mathbf{u}^k-\mathbf{z}+\frac{\boldsymbol{\mu}^k}{\gamma}\|^2+\frac{t}{2}\|\mathbf{z}-\mathbf{z}^k\|^2\right\}, \tag{18}$$

which can be solved with projected gradient descent in the form of Algorithm 1 with a constant step size $\eta \leq \frac{1}{l_d}$. The projected gradient descent for the z-subproblem includes an additional proximal term compared to Algorithm 1. Note that (18) is strongly convex and smooth from Lemma 3.7, then we can derive the complexity results for finding an ϵ_k-optimal solution for $\mathbf{z}$-subproblem in k-th iteration of Algorithm 2.

Lemma 3.8. *In k-th iteration of Algorithm 2, an ϵ_k-optimal solution $\mathbf{z}^{k+1}$ is generated in $\mathcal{O}(\frac{l_d}{t-\alpha_d}\log(\frac{1}{\epsilon_k}))$ projected gradient descent oracles.*

The results of complexity of inner iterations utilize the conclusive findings in Bubeck et al. (2015). Then we make the assumptions concerning $\mathbf{z}$-subproblem and $\boldsymbol{\mu}$.

Assumption 3.9. The sequence $\{\epsilon_k\}$ satisfies $\sum\limits_{k=1}^{\infty} \epsilon_k < \infty$.

Assumption 3.10. The sequence $\{\boldsymbol{\mu}^k\}$ is bounded and satisfies $\sum\limits_{k=1}^{\infty} \|\boldsymbol{\mu}^k - \boldsymbol{\mu}^{k+1}\|^2 < \infty$.

Assumption 3.9 is introduced by Wang et al. (2019) and Assumption 3.10 is popularly employed in ADMM approaches Xu et al. (2012); Bai et al. (2021); Shen et al. (2014); Cui et al. (2024). Based on Assumptions 3.9 and 3.10, we propose the convergence result for Algorithm 2 in Theorem 3.11.

Theorem 3.11. *Algorithm 2 can find an ϵ-KKT point $(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^{k+1})$ of (17) within $\mathcal{O}(1/\epsilon^2)$ iterations.*

From Theorem 3.11, we further conclude that Algorithm 2 finds an ϵ-KKT point of (17) within $\mathcal{O}(1/\epsilon^2)$ iterations. we provide the detailed proofs and extension to problem (2) with multiple regularizers in Appendix B.

4 Numerical Experiments

In this section, we conduct experiments to compare LDPM with existing algorithms for hyperparameter optimization on synthetic data and real datasets, respectively. In specific, we mainly compare our LDPM with grid search, random search, TPE (Bergstra et al., 2013), IJGO (Feng & Simon, 2018), VF-iDCA (Gao et al., 2022), LDMMA (Chen et al., 2024), GAFFA (Yao et al., 2024a). All experiments are performed on a computer with Intel(R) Core(TM) i7-10710U CPU @ 1.10GHz 1.61 GHz and 16.00 GB memory. The code is implemented using Python 3.9. We consider hyperparameter optimization for elastic net and (sparse) group lasso. In this section, we present part of the experimental results on synthetic data, with additional results and detailed descriptions of the data generation and parameters for several methods included in Appendix D.

4.1 Sparse Group Lasso

We conduct experiments with different data scales and report results in Figure 1. The results of the search methods and Bayesian method (TPE) are not presented in Figure 1 due to its lower efficiency and instability. We have included the specific numerical results in tabular form in Appendix D.1. We observe that LDPM consistently outperforms other algorithms in terms of computational efficiency. As the data scale increases, the superiority of our approach becomes increasingly evident, demonstrating the advantages of LDPM in large-scale hyperparameter optimization. In contrast, gradient-free methods exhibit significant instability when handling numerous hyperparameters, while IGJO converges slowly and demands substantial computational resources. Our iteration process is independent of any solvers, allowing it to outperform LDMMA and VF-iDCA, both of which rely on specific solvers for their iterative subproblems.

4.2 Elastic Net

The numerical results on elastic net are reported in Figure 2. Overall, LDPM achieves the highest solution quality in the shortest running time on this problem model. Similar to Section 4.1, the results of the search method and Bayesian method are not presented in the figure. Instead, we have included other results in tabular form in the Appendix D.1. Overall, LDPM achieves the lowest test error with

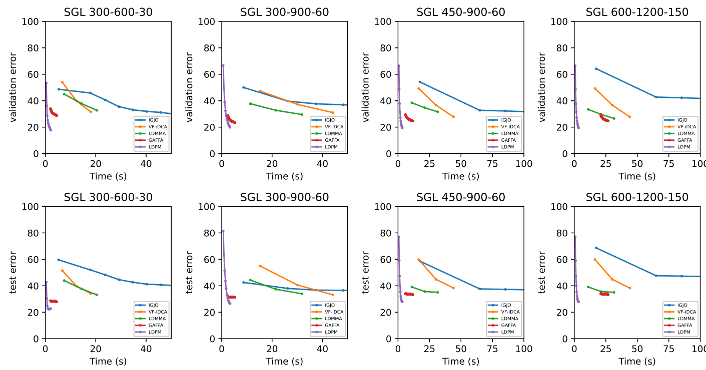

Figure 1: Comparison of the algorithms on Group Lasso problem for synthetic datasets in different scales

significantly lower time costs, particularly in large-scale data scenarios. While the gradient-based method IGJO demonstrates slightly better accuracy and efficiency and its convergence is notably slow as illustrated in the figure. Meanwhile, VF-iDCA and LDMMA maintain consistently low validation errors across all experiments. However, both algorithms suffer from overfitting, resulting in increased test errors as the iterations progress.

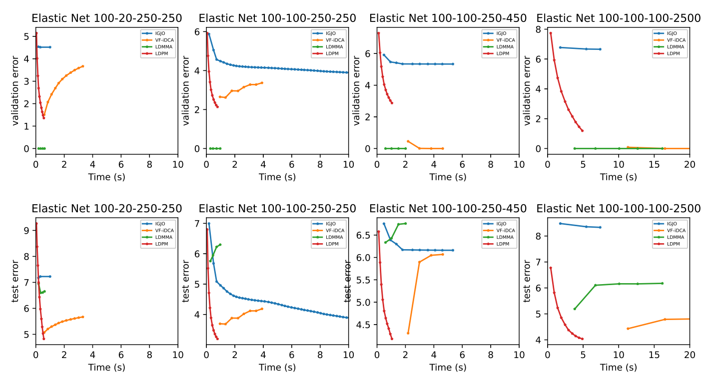

Figure 2: Comparison of the algorithms on Elastic Net problem for synthetic datasets in different scales

We present other experimental results in the form of figures and tables in Appendix D.1 and D.2. demonstrating the robustness and applicability of our algorithm. Notably, our algorithm does not utilize any open-source libraries like CVXPY or commercial optimization solvers, such as MOSEK, which are typically employed in many hyperparameter optimization algorithms.

5 CONLUSIONS

This paper addresses hyperparameter optimization in the context of nonsmooth regularizers by proposing a novel penalty method based on lower-level duality (LDPM). Our approach applies penalization to a single-level reformulation, eschewing any implicit value function and instead utilizing the conjugates of atomic functions. We effectively solve the subproblems within this penalization framework using fully first-order methods, including proximal techniques and the alternating direction method of multipliers, while maintaining simplicity by avoiding complex off-the-shelf solvers or high-complexity iterations. Theoretical analyses substantiate the convergence of our method. Our numerical experiments, conducted on both synthetic and real-world datasets, demonstrate that LDPM consistently outperforms existing methodologies, with its advantages particularly pronounced in large-scale scenarios. Looking ahead, we aim to explore nonsmooth loss functions and develop more general algorithms from a stochastic perspective, thereby broadening the applicability and impact of our approach.

REFERENCES

Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your maml. In *International conference on learning representations*, 2018.

Xiaodi Bai, Jie Sun, and Xiaojin Zheng. An augmented lagrangian decomposition method for chance-constrained optimization problems. *INFORMS Journal on Computing*, 33(3):1056–1069, 2021.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Amir Beck and Marc Teboulle. Gradient-based algorithms with applications to signal-recovery problems., 2010.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pp. 115–123. PMLR, 2013.

Quentin Bertrand, Quentin Klopfenstein, Mathieu Blondel, Samuel Vaiter, Alexandre Gramfort, and Joseph Salmon. Implicit differentiation of lasso-type models for hyperparameter optimization. In *International Conference on Machine Learning*, pp. 810–821. PMLR, 2020.

Quentin Bertrand, Quentin Klopfenstein, Mathurin Massias, Mathieu Blondel, Samuel Vaiter, Alexandre Gramfort, and Joseph Salmon. Implicit differentiation for fast hyperparameter selection in non-smooth convex learning. *Journal of Machine Learning Research*, 23(149):1–43, 2022.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

He Chen, Haochen Xu, Rujun Jiang, and Anthony Man-Cho So. Lower-level duality based reformulation and majorization minimization algorithm for hyperparameter optimization. *arXiv preprint arXiv:2403.00314*, 2024.

Lesi Chen, Yaohua Ma, and Jingzhao Zhang. Near-optimal fully first-order algorithms for finding stationary points in bilevel optimization. *arXiv preprint arXiv:2306.14853*, 2023a.

Lesi Chen, Jing Xu, and Jingzhao Zhang. Bilevel optimization without lower-level strong convexity from the hyper-objective perspective. *arXiv preprint arXiv:2301.00712*, 2023b.

Frank H Clarke. *Optimization and nonsmooth analysis*. SIAM, 1990.

Xiangyu Cui, Rujun Jiang, Yun Shi, Rufeng Xiao, and Yifan Yan. Decision making under cumulative prospect theory: An alternating direction method of multipliers. *INFORMS Journal on Computing*, 2024.

Stephan Dempe and Alain Zemkoho. Bilevel optimization. In *Springer optimization and its applications*, volume 161. Springer, 2020.

Marco F Duarte and Yu Hen Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7):826–838, 2004.

Jean Feng and Noah Simon. Gradient-based regularization parameter selection for problems with nonsmooth penalty functions. *Journal of Computational and Graphical Statistics*, 27(2):426–435, 2018.

Matthias Feurer and Frank Hutter. Hyperparameter optimization. *Automated machine learning: Methods, systems, challenges*, pp. 3–33, 2019.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning*, pp. 1165–1173. PMLR, 2017.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International conference on machine learning*, pp. 1568–1577. PMLR, 2018.

Lucy L Gao, Jane Ye, Haian Yin, Shangzhi Zeng, and Jin Zhang. Value function based difference-of-convex algorithm for bilevel hyperparameter selection problems. In *International Conference on Machine Learning*, pp. 7164–7182. PMLR, 2022.

Lucy L Gao, Jane J Ye, Haian Yin, Shangzhi Zeng, and Jin Zhang. Moreau envelope based difference-of-weakly-convex reformulation and algorithm for bilevel programs. *arXiv preprint arXiv:2306.16761*, 2023.

Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. In *International Conference on Machine Learning*, pp. 3748–3758. PMLR, 2020.

Isabelle Guyon, Steve Gunn, Asa Ben-Hur, and Gideon Dror. Result analysis of the nips 2003 feature selection challenge. *Advances in neural information processing systems*, 17, 2004.

MohammadNoor Injadat, Abdallah Moubayed, Ali Bou Nassif, and Abdallah Shami. Systematic ensemble model selection approach for educational data mining. *Knowledge-Based Systems*, 200: 105992, 2020.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16*, pp. 795–811. Springer, 2016.

Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert Nowak. On penalty methods for nonconvex bilevel optimization and first-order stochastic approximation. *arXiv preprint arXiv:2309.01753*, 2023a.

Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. A fully first-order method for stochastic bilevel optimization. In *International Conference on Machine Learning*, pp. 18083–18113. PMLR, 2023b.

Jiajin Li, Caihua Chen, and Anthony Man-Cho So. Fast epigraphical projection-based incremental algorithms for wasserstein distributionally robust support vector machine. *Advances in Neural Information Processing Systems*, 33:4029–4039, 2020.

Junyi Li, Bin Gu, and Heng Huang. A fully single loop algorithm for bilevel optimization without hessian inverse. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7426–7434, 2022.

Zhouchen Lin, Huan Li, and Cong Fang. *Alternating direction method of multipliers for machine learning*. Springer, 2022.

Bo Liu, Mao Ye, Stephen Wright, Peter Stone, and Qiang Liu. Bome! bilevel optimization made easy: A simple first-order approach. *Advances in neural information processing systems*, 35: 17248–17262, 2022.

Meijiao Liu and Yong-Jin Liu. Fast algorithm for singly linearly constrained quadratic programs with box-like constraints. *Computational Optimization and Applications*, 66:309–326, 2017.

Risheng Liu, Xuan Liu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A value-function-based interior-point method for non-convex bi-level optimization. In *International conference on machine learning*, pp. 6882–6892. PMLR, 2021a.

Risheng Liu, Yaohua Liu, Shangzhi Zeng, and Jin Zhang. Towards gradient-based bilevel optimization with non-convex followers and beyond. *Advances in Neural Information Processing Systems*, 34:8662–8675, 2021b.

Risheng Liu, Xuan Liu, Shangzhi Zeng, Jin Zhang, and Yixuan Zhang. Value-function-based sequential minimization for bi-level optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Risheng Liu, Zhu Liu, Wei Yao, Shangzhi Zeng, and Jin Zhang. Moreau envelope for nonconvex bi-level optimization: A single-loop and hessian-free solution strategy. *arXiv preprint arXiv:2405.09927*, 2024.

Jonathan Lorraine, Paul Vicol, and David Duvenaud. Optimizing millions of hyperparameters by implicit differentiation. In *International conference on artificial intelligence and statistics*, pp. 1540–1552. PMLR, 2020.

Zhaosong Lu and Sanyou Mei. First-order penalty methods for bilevel optimization. *SIAM Journal on Optimization*, 34(2):1937–1969, 2024.

Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*, pp. 2113–2122. PMLR, 2015.

Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *International conference on machine learning*, pp. 737–746. PMLR, 2016.

Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with implicit gradients. *Advances in neural information processing systems*, 32, 2019.

Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. Truncated back-propagation for bilevel optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1723–1732. PMLR, 2019.

Han Shen and Tianyi Chen. On penalty-based bilevel gradient descent method. In *International Conference on Machine Learning*, pp. 30992–31015. PMLR, 2023.

Han Shen, Zhuoran Yang, and Tianyi Chen. Principled penalty-based methods for bilevel reinforcement learning and rlhf. *arXiv preprint arXiv:2402.06886*, 2024.

Yuan Shen, Zaiwen Wen, and Yin Zhang. Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization. *Optimization Methods and Software*, 29(2): 239–263, 2014.

Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of computational and graphical statistics*, 22(2):231–245, 2013.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

Bradly Stadie, Lunjun Zhang, and Jimmy Ba. Learning intrinsic rewards as a bi-level optimization problem. In *Conference on Uncertainty in Artificial Intelligence*, pp. 111–120. PMLR, 2020.

Po-Wei Wang, Matt Wytock, and Zico Kolter. Epigraph projections for fast general convex programming. In *International Conference on Machine Learning*, pp. 2868–2877. PMLR, 2016.

Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78:29–63, 2019.

Yangyang Xu, Wotao Yin, Zaiwen Wen, and Yin Zhang. An alternating direction algorithm for matrix completion with nonnegative factors. *Frontiers of Mathematics in China*, 7:365–384, 2012.

Haikuo Yang, Luo Luo, Chris Junchi Li, Michael Jordan, and Maryam Fazel. Accelerating inexact hypergradient descent for bilevel optimization. In *OPT 2023: Optimization for Machine Learning*, 2023.

Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. *Advances in Neural Information Processing Systems*, 34:13670–13682, 2021.

Wei Yao, Haian Yin, Shangzhi Zeng, and Jin Zhang. Overcoming lower-level constraints in bilevel optimization: A novel approach with regularized gap functions. *arXiv preprint arXiv:2406.01992*, 2024a.

Wei Yao, Chengming Yu, Shangzhi Zeng, and Jin Zhang. Constrained bi-level optimization: Proximal lagrangian value function approach and hessian-free algorithm. *arXiv preprint arXiv:2401.16164*, 2024b.

Jane J Ye, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. Difference of convex algorithms for bilevel programs with applications in hyperparameter selection. *arXiv preprint arXiv:2102.09006*, 2021.

Jane J Ye, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. Difference of convex algorithms for bilevel programs with applications in hyperparameter selection. *Mathematical Programming*, 198(2):1583–1616, 2023.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67, 2006.

Hui Zou and Trevor Hastie. Regression shrinkage and selection via the elastic net, with applications to microarrays. *JR Stat Soc Ser B*, 67:301–20, 2003.

A PROOF IN SECTION 2

In this subsection, we provide the proof for the results concerning the penalty framework in Section 2. First, Lemma 2.5 and Proposition 2.7 hold under the strong duality of (3) (Boyd & Vandenberghe, 2004). We present detailed proofs for Lemma 2.6, Proposition 2.8 and Theorem 2.9 in the subsequent discussion.

A.1 PROOF OF LEMMA 2.6

Proof. We restate the lower-level problem of (2) as follows,

$$\min_{\mathbf{x}}\{l(\mathbf{x})+\sum_{i=1}^{r}\lambda_i R_i(\mathbf{x})\}. \tag{19}$$

We first analyze the maximum and minimum in (5). From Lemma 2.5 and Proposition 2.7, we know that the $\max$ operator with respect to $\boldsymbol{\rho}$ is achieved at $\boldsymbol{\rho}_i^*$ defined in (11). Meanwhile, the $\min$ operator of $\mathbf{x}$ occurs at $\mathbf{x}=\mathbf{x}_{\boldsymbol{\lambda}}$. According to the definition of $p(\mathbf{x},\boldsymbol{\rho},\boldsymbol{\lambda},\mathbf{r})$, we deduce that

$$\begin{aligned}
p(\mathbf{x},\boldsymbol{\rho},\boldsymbol{\lambda},\mathbf{r}) &= l(\mathbf{x})+l^*(-\sum_{i=1}^{r}\boldsymbol{\rho}_i)+\sum_{i=1}^{r}\lambda_i r_i\\
&\overset{(a)}{\geq} l(\mathbf{x})+\sum_{i=1}^{r}\lambda_i R_i(\mathbf{x})+l^*(-\sum_{i=1}^{r}\boldsymbol{\rho}_i)\\
&\overset{(b)}{\geq} l(\mathbf{x})+\sum_{i=1}^{r}\lambda_i R_i(\mathbf{x})-\min_{\boldsymbol{\rho}}\{l^*(-\sum_{i=1}^{r}\boldsymbol{\rho}_i)+\sum_{i=1}^{r}\lambda_i R_i^*(\frac{\boldsymbol{\rho_i}}{\lambda_i})\}\\
&\overset{(c)}{=} l(\mathbf{x})+\sum_{i=1}^{r}\lambda_i R_i(\mathbf{x})-l(\mathbf{x}_{\boldsymbol{\lambda}})+\sum_{i=1}^{r}\lambda_i R_i(\mathbf{x}_{\boldsymbol{\lambda}})\\
&\overset{(d)}{\geq} \frac{\alpha_l}{2}\|\mathbf{x}-\mathbf{x}_{\boldsymbol{\lambda}}\|^2.
\end{aligned}$$

In the above inequalities, (a) results from the constraint $R_i(\mathbf{x})\leq r_i$, (b) is from the $\min$ operator where the $\min$ and $\max$ operators have been exchanged by adding the negative sign, (c) follows from the results in (5) and (d) leverages the strong convexity of $l(\mathbf{x})$ and the quadratic-growth condition established in Theorem 2 of Karimi et al. (2016). Moreover, when $\mathbf{x}=\mathbf{x}_{\boldsymbol{\lambda}}$ attains the minimum of the lower-level problem of (2), (a) and (c) hold as "=". Then we complete the proof. □

A.2 PROOF OF PROPOSITION 2.8

Proof. For any $(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})$ feasible to (8), we have $L(\mathbf{x}^*)\leq L(\mathbf{x})$. From Lemma 2.6, if holds that $p(\mathbf{x}^*,\boldsymbol{\lambda}^*,\boldsymbol{\rho}^*,\mathbf{r}^*)=0$. Let $\bar{\mathbf{x}}$ be the projection into $L_{\text{opt}}(\boldsymbol{\lambda})$ of $\mathbf{x}$, i.e., $\|\mathbf{x}-\bar{\mathbf{x}}\|=\text{dist}(\mathbf{y},L_{\text{opt}}(\boldsymbol{\lambda}))$. Then we have

$$\begin{aligned}
&\quad L(\mathbf{x})+\beta^*p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})-L(\bar{\mathbf{x}})\\
&\geq L(\mathbf{x})-L(\bar{\mathbf{x}})+\tfrac{\alpha_l\beta^*}{2}\|\mathbf{x}-\bar{\mathbf{x}}\|^2\\
&\overset{(a)}{\geq} L_0\|\mathbf{x}-\bar{\mathbf{x}}\|+\tfrac{\alpha_l\beta^*}{2}\|\mathbf{x}-\bar{\mathbf{x}}\|^2\\
&\geq \min_{\mathbf{t}} L_0\mathbf{t}+\tfrac{\alpha_l\beta^*}{2}\mathbf{t}^2\\
&\overset{(b)}{\geq} -\epsilon_p.
\end{aligned} \tag{20}$$

Here, (a) is from the Lipschitz continuity assumption of $L(\mathbf{x})$, (b) is from the fact that $L_0\mathbf{t}+\frac{\alpha_l\beta^*}{2}\mathbf{t}^2$ attains its minimum at $\mathbf{t}=\frac{L_0}{\alpha_l\beta^*}$. Since $\bar{\mathbf{x}}\in L_{\text{opt}}(\boldsymbol{\lambda})$ is feasible to (2), we know that

$$L(\mathbf{x})+\beta p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})-L(\bar{\mathbf{x}})\geq L(\mathbf{x})+\beta^*p(\mathbf{x}^*,\boldsymbol{\lambda}^*,\boldsymbol{\rho}^*,\mathbf{r}^*)-L(\mathbf{x}^*)\geq -\epsilon_p,\forall\beta\geq\beta^*.$$

Along with the fact that $p(\mathbf{x}^*,\boldsymbol{\lambda}^*,\boldsymbol{\rho}^*,\mathbf{r}^*)=0$, we know that

$$L(\mathbf{x}^*)+p(\mathbf{x}^*,\boldsymbol{\lambda}^*,\boldsymbol{\rho}^*,\mathbf{r}^*)<L(\mathbf{x})+\beta p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})+\epsilon_p,\forall\beta\geq\beta^*. \tag{21}$$

Therefore, we conclude that $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*)$ is a ϵ_p-global optimal solution of (10) with $\beta \geq \beta^*$.

On the converse, for any $(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ feasible for (10), we have $L(\mathbf{x}_\beta^*) + \beta p(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*) \leq L(\mathbf{x}) + \beta(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) + \epsilon_1$. Substituting $(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) = (\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*)$, we deduce that

$$\begin{aligned} L(\mathbf{x}_\beta^*) + \beta p(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*) &\leq L(\mathbf{x}^*) + \epsilon_1 \\ &\overset{(c)}{\leq} L(\mathbf{x}_\beta^*) + \beta p(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*) + \epsilon + \epsilon_1. \end{aligned} \tag{22}$$

where (c) follows from the inequality relation in (20). Therefore, we have $p(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*) \leq (\epsilon + \epsilon_1)/(\beta - \beta^*)$. Define $\epsilon_\beta = p(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*)$, then we have $\epsilon_\beta \leq (\epsilon + \epsilon_1)/(\beta - \beta^*)$. Then for any $(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$ feasible for (12) with $\epsilon = \epsilon_\beta$, it holds that $L(\mathbf{x}_\beta^*) + \beta p(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*) \leq L(\mathbf{x}) + \beta(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r})$, which implies that

$$L(\mathbf{x}_\beta^*) - L(\mathbf{x}) \leq \beta(p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) - \epsilon_\beta) \leq 0.$$

Here, we prove that $(\mathbf{x}_\beta^*, \boldsymbol{\lambda}_\beta^*, \boldsymbol{\rho}_\beta^*, \mathbf{r}_\beta^*)$ is a global solution for 12 with $\epsilon = \epsilon_\beta$. □

A.3 Proof of Theorem 2.9

Proof. Since $(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*)$ is an ϵ-optimal solution of (10), we have

$$L(\mathbf{x}_\epsilon^*) + \beta p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*) \leq L(\mathbf{x}) + p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) + \epsilon. \tag{23}$$

Note that the conclusion in Proposition 2.8 still holds. Substituting $(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) = (\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*)$ with the fact $p(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*) = 0$, we have

$$L(\mathbf{x}_\epsilon^*) + \beta p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*) \leq L(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*) + \epsilon \leq L(\mathbf{x}_\epsilon^*) + \beta^* p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*) + 2\epsilon,$$

where the last inequality follows from (21). Then we have

$$p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*) \leq \frac{2\epsilon}{\beta - \beta^*}.$$

Meawhile, $(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*)$ is feasible for the following problem

$$\begin{aligned} \min_{\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}} \quad & L(\mathbf{x}) \\ \text{s.t.} \quad & p(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{r}) \leq p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*). \end{aligned} \tag{24}$$

From (23), we have $L(\mathbf{x}_\epsilon^*) - L(\mathbf{x}^*) \leq \beta(p(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*) - p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*)) + \epsilon$. While $p(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\rho}^*, \mathbf{r}^*) = 0 \leq p(\mathbf{x}_\epsilon^*, \boldsymbol{\lambda}_\epsilon^*, \boldsymbol{\rho}_\epsilon^*, \mathbf{r}_\epsilon^*)$, we have $L(\mathbf{x}_\epsilon^*) - L(\mathbf{x}^*) \leq \epsilon$. □

B Proof in Section 3

In this section, we provide the proofs for the convergence results of our proposed algorithms in Section 3.

B.1 Proof of Lemma 3.5

Proof. From the definition $g_1(\mathbf{z}) = \mathcal{I}_{\mathcal{R}_1 \cap \mathcal{R}_1^*}(\mathbf{z})$, it holds that $\text{prox}_{tg_1} = \text{proj}_{\mathcal{R}_1 \cap \mathcal{R}_1^*}$ for $t > 0$. We define $P_L(\mathbf{z}) = L(\mathbf{x}) + \bar{\beta} p(\mathbf{z})$, then the update of $\mathbf{z}$ can be written as

$$\mathbf{z}^{k+1} = \text{prox}_{\bar{\eta} g_1}(\mathbf{z}^k - \bar{\eta} \nabla P_L(\mathbf{z}^k)).$$

From the l_p-smooth of $P_L(\mathbf{z})$, we have

$$P_L(\mathbf{z}^{k+1}) \leq P_L(\mathbf{z}^k) + \langle \nabla P_L(\mathbf{z}^k), \mathbf{z}^{k+1} - \mathbf{z}^k \rangle + \frac{l_p}{2} \|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2. \tag{25}$$

In addition, we denote $\bar{\mathbf{z}}^{k+1} = \mathbf{z}^k - \bar{\eta} \nabla_{\mathbf{z}} P_L(\mathbf{z}^k)$, then we have

$$\langle \bar{\mathbf{z}}^{k+1} - \mathbf{z}^k, \mathbf{z}^{k+1} - \mathbf{z}^k \rangle \overset{(a)}{\leq} \bar{\eta} g_1(\mathbf{z}^k) - \bar{\eta} g_1(\mathbf{z}^{k+1}) \overset{(b)}{=} 0.$$

where (a) is from Theorem 6.39 in Beck (2017) and (b) follows from the fact that $\mathbf{z}^{k+1}, \mathbf{z}^k \in \mathcal{R}_1 \cap \mathcal{R}_1^*$. Substituting the $\bar{\mathbf{z}}^{k+1} = \mathbf{z}^k - \bar{\eta}\nabla_{\mathbf{z}} P_L(\mathbf{z}^k)$, we have

$$\langle \nabla P_L(\mathbf{z}^k), \mathbf{z}^{k+1} - \mathbf{z}^k \rangle \leq -\frac{1}{\bar{\eta}} \|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2. \tag{26}$$

Combining (25) and (26), we obtain that

$$P_L(\mathbf{z}^{k+1}) \leq P_L(\mathbf{z}^k) + (-\frac{1}{\bar{\eta}} + \frac{l_p}{2})\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2,$$

which implies that $P_L(\mathbf{z}^{k+1}) - P_L(\mathbf{z}^k) \leq 0$ from $\bar{\eta} \leq \frac{l_p}{2}$. Utilizing the definition of $P(\mathbf{z})$, we have $L(\mathbf{x}^{k+1}) + \bar{\beta}p(\mathbf{z}^{k+1}) - L(\mathbf{x}^k) - \bar{\beta}p(\mathbf{z}^k) \leq 0$. In addition, we observe that $\{P_L(\mathbf{z}^k)\}$ is nonincreasing and bounded below, it converges. Therefore, $P_L(\mathbf{z}^k) - P_L(\mathbf{z}^{k+1}) \to 0$ as $k \to \infty$, along with $\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 \to 0$ because $\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 \leq 1/(\frac{1}{\bar{\eta}} - \frac{l_p}{2})(P_L(\mathbf{z}^k) - P_L(\mathbf{z}^{k+1}))$. Then we complete the proof.

□

B.2 Proof of Theorem 3.6

Proof. According to the definition of $P_L(\mathbf{z})$ and $g_1(\mathbf{z})$, we know that (14) can be equivalently presented as the following form:

$$\min_{\mathbf{z}} \quad P_L(\mathbf{z}) + g_1(\mathbf{z}). \tag{27}$$

Then we define $M(\mathbf{z}) = \frac{1}{\bar{\eta}}[\mathbf{z} - \mathrm{prox}_{\bar{\eta}g_1}(\mathbf{z} - \bar{\eta}\nabla P_L(\mathbf{z}))] = \frac{1}{\bar{\eta}}[\mathbf{z} - \mathrm{proj}_{\mathcal{R}_1\cap\mathcal{R}_1^*}(\mathbf{z} - \bar{\eta}\nabla P_L(\mathbf{z}))]$, representing the gradient mapping used for updating $\mathbf{z}$ in Algorithm 1 with respesct to (27). Then it holds that $M(\mathbf{z})$ is $(\frac{2}{\bar{\eta}} + l_p)$-Lipschitz continuous (Lemma 10.10 in Beck (2017)). Let $\bar{\mathbf{z}}$ is a limit point of $\{\mathbf{z}^k\}$. Then there exists a subsequence $\{\mathbf{z}^{k_j}\}$ converging to $\bar{\mathbf{z}}$. For any $j \geq 0$, we have

$$\|M(\bar{\mathbf{z}})\| \leq \|M(\mathbf{z}^{k_j}) - M(\bar{\mathbf{z}})\| + \|M(\mathbf{z}^{k_j})\| \leq (\frac{2}{\bar{\eta}} + l_p)\|\mathbf{z}^{k_j} - \bar{\mathbf{z}}\| + \|M(\mathbf{z}^{k_j})\|.$$

Based on proof for Lemma 3.5, we know that $\|M(\mathbf{z}^{k_j})\| \to 0$ as $j \to \infty$. Therefore, we conclude that $\|M(\bar{\mathbf{z}})\| = 0$ with taking the limit of the above inequality. According to the definition of $M(\mathbf{z})$, we observe that

$$\bar{\mathbf{z}} - \bar{\eta}\nabla P_L(\bar{\mathbf{z}}) \in \bar{\eta}\partial g_1(\bar{\mathbf{z}}),$$

which implies $\nabla P_L(\bar{\mathbf{z}}) \in \partial g_1(\bar{\mathbf{z}})$. From the first-order optimality condition, we conclude that $\bar{\mathbf{z}}$ serves as a stationary point of (14). □

B.3 Proof of Lemma 3.8

Theorem B.1. *(Theorem 3.10 in Bubeck et al. (2015)) Let f be α-strongly convex and β-smooth on $\mathcal{X}$. Then projected gradient descent with $\eta = \frac{1}{\beta}$ satisfies for $t \geq 0$,*

$$\|x_{t+1} - x^*\|^2 \leq \exp(-\frac{t\beta}{\alpha})\|x_1 - x^*\|^2$$

According to Lemma 3.7, we know that the $\mathbf{z}$-subproblem in Algorithm 2 is $(t-\alpha_d)$-strongly convex and l_d-smooth, where we denote $\alpha_d = \frac{\beta}{2} - (1+\beta)\alpha_l - \gamma$ and $l_d = \gamma + t + l_1 + \beta(l_1 + \alpha_l + 1)$. Therefore, the complexity for finding an ϵ_k-optimal solution of $\mathbf{z}$-subproblem with projected grdient descent is $\mathcal{O}(\frac{l_d}{t-\alpha_d}\log(\frac{1}{\epsilon_k}))$.

B.4 Proof of Theorem 3.11

Proof. From the update of $\mathbf{u}$-subproblem, we have

$$\mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^{k+1}, \boldsymbol{\mu}^k) \leq \mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^k, \boldsymbol{\mu}^k).$$

Similarly, we derive from the iteration form and strong convexity of $\mathbf{z}$-subproblem that

$$\mathcal{L}_\gamma(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^k) - \mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^{k+1}, \boldsymbol{\mu}^k) \geq \frac{2t - \alpha_d}{2}\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2.$$

Furthermore, we obtain from the update of $\boldsymbol{\mu}$ that

$$\begin{aligned} & \mathcal{L}_\gamma(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^k) - \mathcal{L}_\gamma(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^{k+1}) \\ = \ & \langle \boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k, \mathbf{u}^{k+1} - \mathbf{z}^{k+1} \rangle \\ = \ & -\frac{1}{\gamma}\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\|^2. \end{aligned}$$

In summary, we obtain that

$$\mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^k, \boldsymbol{\mu}^k) - \mathcal{L}_\gamma(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^{k+1}) \geq \frac{2t - \alpha_d}{2}\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 - \frac{1}{\gamma}\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\|^2 \tag{28}$$

We use the extended formula for Clark generalized gradient of a sum of two functions. $\partial(f_1 + f_2)(x) \subset \partial f_1(x) + \partial f_2(x)$ if f_1 and f_2 are finite at $\mathbf{x}$ and f_2 is differentiable at x. The equality holds if f_1 is regular at x (Theorem 2.9.8 in Clarke (1990)). Then we have

$$\begin{aligned} B_k \ & \stackrel{\triangle}{=} \ \partial_{\mathbf{z}} \left\{ L(\mathbf{x}^{k+1}) + \beta p(\mathbf{z}^{k+1}) + \langle \boldsymbol{\mu}^k, \mathbf{z}^{k+1} \rangle + \tfrac{\gamma}{2}\|\mathbf{u}^{k+1} - \mathbf{z}^{k+1}\|^2 + \tfrac{t}{2}\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 \right\} \\ & = \ \partial_{\mathbf{z}}\{L(\mathbf{x}^{k+1}) + \beta p(\mathbf{z}^{k+1})\} + (\boldsymbol{\mu}^k + \gamma(\mathbf{u}^{k+1} - \mathbf{z}^{k+1})) + t(\mathbf{z}^{k+1} - \mathbf{z}^k) \\ & = \ \partial_{\mathbf{z}}\{L(\mathbf{x}^{k+1}) + \beta p(\mathbf{z}^{k+1})\} + \boldsymbol{\mu}^{k+1} + t(\mathbf{z}^{k+1} - \mathbf{z}^k). \end{aligned} \tag{29}$$

From the ϵ_k-optimality condition, we obtain that $\|B_k\| \leq \epsilon_k$. From the assumption the L and p is bounded below, we know that

$$\mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^k, \boldsymbol{\mu}^k) = L(\mathbf{x}^k) + \beta p(\mathbf{z}^k) + g_1(\mathbf{z}^k) + g_2(\mathbf{u}^k) + \frac{\gamma}{2}\|\mathbf{u}^k + \mathbf{z}^k + \boldsymbol{\mu}^k/\gamma\|^2 - \|\boldsymbol{\mu}^k\|^2/2\gamma > -\infty$$

Therefore, $\mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^k, \boldsymbol{\mu}^k)$ is lower bounded by some $\mathcal{L}_b$. Moreover, with Assumption 3.10 holding, we find that $\mathcal{L}_\gamma(\mathbf{z}^0, \mathbf{u}^0, \boldsymbol{\mu}^0) - \mathcal{L}_\gamma(\mathbf{z}^{K+1}, \mathbf{u}^{K+1}, \boldsymbol{\mu}^{K+1}) + \frac{2}{\gamma}\sum_{k=1}^{K+1}\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\|^2 < \infty$ for all $K \in \mathbb{N}$. We compress (28) from $k = 1$ to $K + 1$ and obtain that

$$\begin{aligned} & \mathcal{L}_\gamma(\mathbf{z}^0, \mathbf{u}^0, \boldsymbol{\mu}^0) - \mathcal{L}_\gamma(\mathbf{z}^{K+1}, \mathbf{u}^{K+1}, \boldsymbol{\mu}^{K+1}) + \tfrac{2}{\gamma}\sum_{k=1}^{K+1}\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\|^2 \\ \geq \ & \tfrac{2t - \alpha_d}{2}\sum_{k=1}^{K+1}\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 + \tfrac{1}{\gamma}\sum_{k=1}^{K+1}\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\|^2. \end{aligned} \tag{30}$$

We take the minimum operation from K iterations in (30) and obtain

$$\min_{k \leq K}\left\{\frac{2t - \alpha_d}{2}\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 + \frac{1}{\gamma}\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\|^2\right\} \leq \frac{\mathcal{L}_\gamma(\mathbf{z}^0, \mathbf{u}^0, \boldsymbol{\mu}^0) - \mathcal{L}_b}{K + 1}$$

Therefore, we observe that algorithm 2 execute $\mathcal{O}(1/\epsilon^2)$ iterations to find $(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^{k+1})$ such that

$$\|\mathbf{z}^{k+1} - \mathbf{z}^k\| \leq \epsilon, \ \|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\| \leq \epsilon.$$

From the update of $\boldsymbol{\mu}$, we further derive that

$$\|\mathbf{u}^{k+1} - \mathbf{z}^{k+1}\| \leq \mathcal{O}(\epsilon)$$

From Assumption 3.9, it holds that

$$\mathrm{dist}(-\boldsymbol{\mu}^{k+1}, \partial_{\mathbf{z}}\{L(\mathbf{x}^{k+1}) + \beta p(\mathbf{z}^{k+1})\}) \leq \mathcal{O}(\epsilon).$$

(29) and Now we consider the optimity condition of $\mathbf{u}$, then we have

$$0 \in \partial g_2(\mathbf{u}^{k+1}) + \boldsymbol{\mu}^k + \gamma(\mathbf{z}^{k+1} - \mathbf{u}^k).$$

Thus, we have

$$\mathrm{dist}(-\boldsymbol{\mu}^{k+1}, \partial g_2(\mathbf{u}^{k+1})) \leq \gamma\|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\| = \mathcal{O}(\epsilon).$$

Then we conclude that $(\mathbf{z}^{k+1}, \mathbf{u}^{k+1}, \boldsymbol{\mu}^{k+1})$ attains an ϵ-KKT point of (17). The proof is adapted from Theorem 4.1 in Lin et al. (2022). □

B.5 EXTENSION TO THE CASES WITH MULTIPLE REGULARIZATION TERMS

We focus on the case (2) involving multiple regularization terms. For this scenario, (10) simplifies to the following formulation:

$$\begin{aligned} \min_{\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}} \quad & L(\mathbf{x}) + \beta p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}). \\ \text{s.t.} \quad & R_i(\mathbf{x}) \le r_i, \|\boldsymbol{\rho}_i\|_* \le \lambda_i, i = 1,2,...,r, \end{aligned} \tag{31}$$

where $p(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r}) = l(\mathbf{x}) + l^*(-\sum_{i=1}^{r}\boldsymbol{\rho}_i) + \sum_{i=1}^{r}\lambda_i r_i$. We adopt the notations $\mathbf{z} = (\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\rho},\mathbf{r})$ and $\mathcal{R}_i, \mathcal{R}_i^*, i = 1,2,...,r$ defined in (13). Similar to Section 3.2, we denote $\mathcal{R}^* \triangleq \cap_{i=1}^r \mathcal{R}_i^*$ and consider variable decomposition for $\cap_{i=1}^r \mathcal{R}_i$. We define $g_i(\mathbf{z}) \triangleq \mathcal{I}_{\mathcal{R}_i \cap \mathcal{R}^*}(\mathbf{z}), i = 1,2,...,r$. Under this conditions, (31) can be rewritten as the following equivalent form,

$$\min_{\mathbf{z}} \quad L(\mathbf{x}) + \beta p(\mathbf{z}) + \sum_{i=1}^{r} g_i(\mathbf{z}). \tag{32}$$

Then we introduce an auxiliary variable $\mathbf{u}_i$ as follows,

$$\begin{aligned} \min_{\mathbf{z}} \quad & L(\mathbf{x}) + \beta p(\mathbf{z}) + g_1(\mathbf{z}) + \sum_{i=1}^{r-1} g_2(\mathbf{u}_i) \\ \text{s.t.} \quad & \mathbf{z} = \mathbf{u}_i, i = 1,2,...,r-1. \end{aligned} \tag{33}$$

We denote the constraints of (33) as $\sum_{i=1}^{r-1}\mathbf{I}_i\mathbf{u}_i + \mathbf{z} = 0$, where $\mathbf{I}_i$ is row full-rank matrix. (33) is a multi-block linearly constrained problem and its augmented Lagrangian function can be expressed as

$$\mathcal{L}_\gamma(\mathbf{z},\mathbf{u},\boldsymbol{\mu}) = L(\mathbf{x}) + \beta p(\mathbf{x}) + \sum_{i=1}^{r} g_i(\mathbf{u}_i) + \langle \boldsymbol{\mu}, \sum_{i=1}^{r-1}\mathbf{I}_i\mathbf{u}_i + \mathbf{z}\rangle + \frac{\gamma}{2}\|\sum_{i=1}^{r-1}\mathbf{I}_i\mathbf{u}_i + \mathbf{z}\|^2$$

Now, we employ multi-block ADMM to minimize equation 33, which cyclically update $\mathbf{u}_i, \mathbf{z}, \boldsymbol{\mu}$ by solving the $\mathbf{u}_i$- and $\mathbf{z}$- subproblems and adopt a dual ascent step for $\boldsymbol{\mu}$. We summarize the iterations in Algorithm 3.

Algorithm 3 ADMM Framework for Problem (33)

1: Initialize $\mathbf{z}^0, \mathbf{u}^0, \sigma^0, \gamma$ and t.
2: **for** $k = 0,1,2,...$ **do**
3: $\quad \mathbf{z}^{k+1} = \arg\min_{\mathbf{z}} \left\{ L(\mathbf{x}) + \beta p(\mathbf{x}) + \langle \boldsymbol{\mu}^k, \mathbf{z}\rangle + \frac{\gamma}{2}\|\sum_{i=1}^{r}\mathbf{I}_i\mathbf{u}_i^{k+1} + \mathbf{z}\|^2 + \frac{t}{2}\|\mathbf{z} - \mathbf{z}^k\|^2 \right\}$.
4: $\quad$ **for** $i = 1,2,...,r-1$ **do**
5: $\quad\quad \mathbf{u}_i^{k+1} = \arg\min_{\mathbf{u}_i} \left\{ g_i(\mathbf{u}_i) + \langle \boldsymbol{\mu}^k, \mathbf{I}_i\mathbf{u}_i\rangle + \frac{\gamma}{2}\|\sum_{j<i}\mathbf{I}_j\mathbf{u}_j^{k+1} + \mathbf{I}_i\mathbf{u}_i + \sum_{j>i}\mathbf{I}_j\mathbf{u}_j^k + \mathbf{z}^k\|^2 \right\}$.
6: $\quad$ **end for**
7: $\quad \boldsymbol{\mu}^{k+1} = \boldsymbol{\mu}^k + \gamma(\sum_{i=1}^{r-1}\mathbf{I}_i\mathbf{u}_i^{k+1} + \mathbf{z}^{k+1})$.
8: **end for**

Theorem B.2. *Suppose that the sequence $\{(\mathbf{z}^k, \mathbf{u}_i^k, \boldsymbol{\mu}^k)\}$ is bounded and $L(\mathbf{x}) + \beta p(\mathbf{x})$ is bounded below with bounded $(\mathbf{z},\mathbf{u})$. Then Algorithm 3 can find an ϵ-approximation KKT point $(\mathbf{z}^{k+1}, \mathbf{u}_i^{k+1}, \boldsymbol{\mu}^{k+1})$ of (equation 33).*

From the update of $\mathbf{u}$-subproblem, we have

$$\mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}_{j\le i}^{k+1}, \mathbf{u}_{j>i}^k, \boldsymbol{\mu}^k) \le \mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}_{j<i}^{k+1}, \mathbf{u}_{j\ge i}^k, \boldsymbol{\mu}^k).$$

Summing over $i = 1,2,...,r$, we have

$$\mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^{k+1}, \boldsymbol{\mu}^k) \le \mathcal{L}_\gamma(\mathbf{z}^k, \mathbf{u}^k, \boldsymbol{\mu}^k).$$

Consequently, the proof of Theorem B.2 follows from the proof of Theorem 3.11 in Appendix B.4.

C CLOSE-FORM PROJECTIONS

We observe that the set $\mathcal{R}_i$ and $\mathcal{R}_i^*$ takes the form of a norm cone, which are epigraphs of the norm and conjugate norm. The corresponding projections are orthogonal projections onto epigraphs, which are explored in Beck (2017); Wang et al. (2016).

Theorem C.1. *(Theorem 6.36 in Beck (2017)) Let*

$$C = \mathrm{epi}(g) = \{(\mathbf{x}, t) | g(\mathbf{x}) \le t\},$$

where g is convex. Then

$$\mathrm{proj}_C((\mathbf{x}, s)) = \begin{cases} (\mathbf{x}, s), & g(\mathbf{x}) \le s, \\ (\mathrm{prox}_{\lambda^* g}, s + \lambda^*), & g(\mathbf{x}) > s, \end{cases}$$

where λ^* is any positive root of the function

$$\psi(\lambda) = g(\mathrm{prox}_{\lambda g}(\mathbf{x}) - \lambda - s.)$$

In addition, ψ is nonincreasing.

Based on Theorem C.1, the projections onto the epigraphs of the l_1 and l_2 norm can be calculated as follows. Let $C_1 = \{(\mathbf{x}, t) | \|\mathbf{x}\|_1 \le t\}$ and $C_2 = \{(\mathbf{x}, t) | \|\mathbf{x}\|_2 \le t\}$. Then it holds that (Example 6.37 and 6.38 in Beck (2017)),

$$\mathrm{proj}_{C_1}((\mathbf{x}, s)) = \begin{cases} (\mathbf{x}, s), & \|\mathbf{x}\|_1 \le s, \\ (\mathcal{T}_{\lambda^*}(\mathbf{x}), s + \lambda^*), & \|\mathbf{x}\|_1 > s, \end{cases}$$

We denote the proximal of l_1-norm as $\mathcal{T}_\lambda = \mathrm{prox}_{\lambda \|\cdot\|_1}$, which is formed as

$$\mathcal{T}_\lambda(y) = [|y| - \lambda]_+ \,\mathrm{sgn}(y) = \begin{cases} y - \lambda, & y \ge \lambda \\ 0, & |y| < \lambda, \\ y + \lambda, & y \le -\lambda. \end{cases}$$

λ^* is any positive root of the nonincreasing function $\varphi(\lambda) = \|\mathcal{T}_\lambda(\mathbf{x})\|_1 - \lambda - s$.

$$\mathrm{proj}_{C_2}((\mathbf{x}, s)) = \begin{cases} \left(\frac{\|\mathbf{x}\|_2 + s}{2\|\mathbf{x}\|_2}\mathbf{x}, \frac{\|\mathbf{x}\|_2 + s}{2}\right), & \|\mathbf{x}\|_2 \ge |s|, \\ (\mathbf{0}, 0), & s < \|\mathbf{x}\|_2 < -s, \\ (\mathbf{x}, s), & \|\mathbf{x}\|_2 \le s. \end{cases}$$

D EXPERIMENTS

We consider hyperparameter optimization for elastic net, group lasso, and sparse group lasso. These three models only use a combination of regularization terms $||\cdot||_1$, $||\cdot||_2$, $\frac{1}{2}||\cdot||_2^2$, as the form equation 2 . The elastic net (Zou & Hastie (2003)) is a linear combination of the lasso and ridge penalties, the group lasso (Yuan & Lin (2006)) is an extension of the Lasso with penalty to predefined groups of coefficients, and the sparse group lasso (Simon et al. (2013)) combines the group lasso and lasso penalties, which are designed to encourage sparsity and grouping of predictors (Feng & Simon (2018)). We consider hyperparameter optimization for elastic net, group lasso, and sparse group lasso. To compare the performance of each method, we calculate validation and test error with obtained LL minimizers in each experiment. Our competitors are implemented using code from https://github.com/SUSTech-Optimization/VF-iDCA, https://github.com/libralicoho/LDMMA, and https://github.com/SUSTech-Optimization/BiC-GAFFA. Note that in the experiments, besides our method, solvers are all needed to solve the subproblems. and we uniformly apply the CVXPY package to them with the open source solvers ECOS and SCS only. All experiments are run on a computer with Intel(R) Core(TM) i7-10710U CPU @ 1.10GHz 1.61 GHz and 16.00 GB memory.

In our experiments, the parameter settings for LDPM are relatively loose. Since we use an exact penalty function, good results can be obtained with small penalty coeficient β 1 or 10. Additionally, for smooth problems, we use the APG algorithm for the sub-problems, so the choice of step size α is not very sensitive due to the accelerated convergence rate. It is worth emphasizing that our algorithm is completely first-order and does not rely on any solver.

D.1 EXPERIMENTS ON SYNTHETIC DATA

D.1.1 ELASTIC NET

The synthetic data is simulated in a similar manner as Gao et al. (2022) We sample the feature vectors $\mathbf{a}_i \in \mathbb{R}^p$ from a normal distribution with mean 0 and covariance $\operatorname{cor}(a_{ij}, a_{ik}) = 0.5^{|j-k|}$. We obtain the response vector $\mathbf{b}$ according to $b_i = \boldsymbol{\beta}^\top \mathbf{a}_i + \sigma\epsilon_i$, where β_i is randomly 0 or 1 and $\sum_{i=1}^p \beta_i = 15$; the noise $\boldsymbol{\epsilon}$ is sampled from the standard normal distribution, and σ is chosen such that the signal-to-noise ratio SNR $\triangleq \|A\boldsymbol{\beta}\|/\|\mathbf{b} - A\boldsymbol{\beta}\|$ equals 2.

We implement the algorithms we compared with the same parameters according to Chen et al. (2024). In our experiment, we set $\beta = 1$, $\gamma = 10$, and $t = 1$. Besides, we set the same initial guess $\lambda = (0.01, 0.01)$ and $r = (0.1, 0.1)$ as LDMMA and VF-iDCA, as well as the stopping criteria

$$\max\left\{\frac{\|\mathbf{z}^{k+1} - \mathbf{z}^k\|}{\sqrt{1 + \|\mathbf{z}^k\|^2}}, t^{k+1}\right\} < 0.1, \tag{34}$$

We conduct repeated experiments with 30 randomly generated synthetic data, and calculate the mean and variance. The numerical results on elastic net are reported in Table 4 and we also plot the performance curve of the algorithms under each experiment setting in Figure 2. Overall, our algorithm achieves the lowest test error and validation error is also among lowest , along with its significantly lowest time cost, especially in large-scale data cases. Traditional gradient-free methods (grid search, random search, and TPE) have expensive search time cost and poor performance on test dataset. Gradient-based method IGJO perform slightly better on accuracy and efficiency, but it converges very slowly as shown in the figure. Considering the two solver-based algorithm, i.e. VF-iDCA and LDMMA, their validation error keeps very low in all experiments but they both suffer overfitting, where the test error goes higher as the iteration increases.

D.1.2 SPARSE GROUP LASSO

The synthetic data is simulated according to Gao et al. (2022). Each dataset contains 100 training data, 100 validation data and 100 test data. The feature vector $\mathbf{a}_i \in \mathbb{R}^p$ is sampled from the standard normal distribution. The response vector $\boldsymbol{b}$ is calculated by $b_i = \boldsymbol{\beta}^\top \mathbf{a}_i + \sigma\epsilon_i$, where $\boldsymbol{\beta} = \left[\boldsymbol{\beta}^{(1)}, \boldsymbol{\beta}^{(2)}, \boldsymbol{\beta}^{(3)}\right]$, $\boldsymbol{\beta}^{(i)} = (1, 2, 3, 4, 5, 0, \ldots, 0)$, for $i = 1, 2, 3$. $\boldsymbol{\epsilon}$ are generated from the standard normal distribution, and σ is chosen such that the SNR is 2.

We implement the algorithms we compared with the same parameters according to Chen et al. (2024). In our experiment, we set $\beta = 1$, $\gamma = 10$, and $t = 1$. Besides, we set the same initial guess $\lambda = 0.05\mathbf{1}$ and $r = 0.5\mathbf{1}$ as LDMMA and VF-iDCA, and tol=0.1.

We conduct experiments with different data scales and report numerical results over 30 repetitions in Table 5 with Figure 1. For each experiment, the generated datasets consist of n training, n/3 validation, and 100 test samples. LDPM achieves both lowest test error and validation error and outperforms other algorithms in terms of time cost. As the scale of data increases, our method consistently finds the best hyperparameters and model solutions, which indicates the superiority of LDPM in large-scale hyperparameter optimization. In comparison, gradient-free methods become extremely unstable when facing dozens of hyperparameters, while IGJO converges slowly and requires huge amount of computation. Due to the size of the problem, solving each subproblem (constrained optimization) is extremely time-consuming for LDMMA and VF-iDCA, even though they only needs several iterations to find good solutions. BiC-GAFFA runs as fast as LDPM in the gradient step iterations, but still requires some time to obtain an initial feasible point by solver in the first iteration.

D.1.3 GROUP LASSO

The synthetic data is generated in the same way as sparse group lasso. We set $\beta = 1$, and $\eta = 0.001$, with initial guess $\lambda = 0.1\mathbf{1}$ and $r = 0.5\mathbf{1}$ and $tol = 0.05$ in LDPM. We implement the rest algorithms with a slight modification for the problem with the same parameter setting in sparse group lasso experiments.

Table 1: Group lasso problems on synthetic data, where p and M represent the number of covariates and covariate groups, respectively, and n represent the data scale described above.

Settings	Methods	Time(s)	Val. Err.	Test Err.	Settings	Time(s)	Val. Err.	Test Err.
	Grid	85.18 ± 4.61	45.33 ± 6.79	48.84 ± 6.76		100.91 ± 7.80	45.38 ± 5.74	48.19 ± 6.69
$n = 300$	Random	79.11 ± 5.10	37.92 ± 5.13	45.66 ± 6.34	$n = 450$	93.06 ± 6.72	45.18 ± 7.41	43.86 ± 4.89
$p = 600$	IGJO	99.01 ± 9.41	34.86 ± 5.80	45.87 ± 4.67	$p = 900$	94.22 ± 7.79	38.75 ± 7.72	43.99 ± 5.30
$M = 30$	VF-iDCA	9.70 ± 2.30	27.21 ± 5.37	32.95 ± 7.16	$M = 60$	21.14 ± 6.22	24.07 ± 2.20	36.15 ± 6.01
	LDMMA	27.02 ± 2.52	25.76 ± 3.60	34.74 ± 4.34		38.80 ± 4.59	26.95 ± 4.33	33.69 ± 6.17
	GAFFA	3.56 ± 0.11	29.73 ± 6.48	31.22 ± 5.88		10.88 ± 0.63	25.84 ± 7.19	29.74 ± 6.43
	LDPM	$\mathbf{0.55 \pm 0.04}$	$\mathbf{17.42 \pm 3.74}$	$\mathbf{25.10 \pm 3.68}$		$\mathbf{0.91 \pm 0.03}$	$\mathbf{19.20 \pm 5.11}$	$\mathbf{22.28 \pm 4.28}$
	Grid	107.95 ± 10.36	46.13 ± 5.54	46.21 ± 7.94		128.77 ± 9.68	45.33 ± 6.43	47.32 ± 7.24
$n = 300$	Random	95.02 ± 7.27	43.66 ± 6.31	42.18 ± 6.77	$n = 600$	131.50 ± 11.36	48.79 ± 7.66	48.91 ± 9.13
$p = 900$	IGJO	122.64 ± 9.96	30.56 ± 6.46	47.36 ± 5.76	$p = 1200$	152.10 ± 15.19	37.21 ± 6.89	42.30 ± 7.59
$M = 60$	VF-iDCA	9.12 ± 0.07	24.40 ± 5.62	30.25 ± 4.03	$M = 150$	67.71 ± 9.53	27.53 ± 5.16	35.61 ± 6.98
	LDMMA	38.13 ± 3.41	24.94 ± 6.68	30.12 ± 4.85		47.11 ± 5.86	18.51 ± 4.09	27.58 ± 4.19
	GAFFA	5.17 ± 0.17	28.39 ± 6.22	29.95 ± 5.23		34.88 ± 9.98	25.39 ± 5.41	26.81 ± 5.39
	LDPM	$\mathbf{0.86 \pm 0.02}$	$\mathbf{20.69 \pm 3.88}$	$\mathbf{27.04 \pm 4.58}$		$\mathbf{1.83 \pm 0.02}$	$\mathbf{19.18 \pm 5.03}$	$\mathbf{25.35 \pm 6.27}$

We conduct experiments with different data scales and report numerical results over 30 repetitions in Table 6 with Figure 3. For each experiment, the generated datasets consist of n training, n/3 validation, and 100 test samples. As a comparison to the Sparse Group Lasso experiment, we simply use APG to solve our problem thanks to the single regularization term (see 1), making our algorithm faster. LDPM achieves both lowest test error and validation error and outperforms other algorithms in terms of time cost. Performance of the rest algorithms is similar to that in the previous Sparse Group Lasso experiments. Note that in the experiments, We observe that the solver-based algorithms like LDMMA and VF-iDCA sometimes unable to run because of the solvers failure facing large scale data.

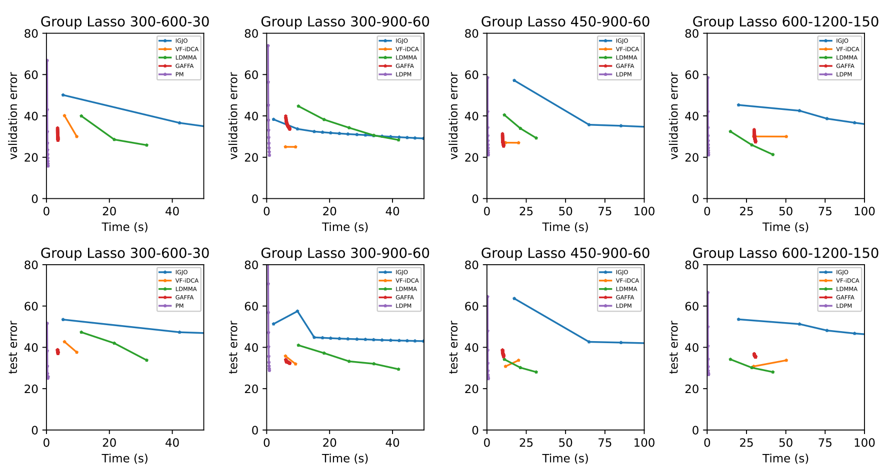

Figure 3: Comparison of the algorithms on Group Lasso problem for synthetic datasets in different scales

D.2 Experiments on real data

We conduct experiments on the algorithm using real datasets from libsvm (Chang & Lin (2011)). The datasets we selected are gisette (Guyon et al. (2004)) and sensit (Duarte & Hu (2004)). Following the data participation rule as Gao et al. (2022), we randomly extracted 50, 25 examples as training set; 50, 25 examples as validation set, respectively; and the remaining for testing. We test different algorithms on the same partition for 30 repeated experiments. We perform hyperparameter tuning for elastic net and sparse group lasso on the two high-dimensional real datasets. The parameters are set the same as in the synthetic data experiments. We set The results are reported in Table 2, 3, and Figure 4, 5, showing the consistent effectiveness of our method.

Table 2: Elastic net problem on datasets gisette and sensit, where $|I_{tr}|$, $|I_{val}|$, $|I_{te}|$ and p represent the number of training samples, validation samples, test samples and features, respectively.

Settings	Methods	Time(s)	Val. Err.	Test Err.	Settings	Time(s)	Val. Err.	Test Err.
gisette	Grid	58.77 ± 3.37	0.25 ± 0.04	0.23 ± 0.02	sensit	1.08 ± 0.15	1.24 ± 0.49	1.22 ± 0.47
$p = 5000$	Random	65.42 ± 8.56	0.25 ± 0.04	0.23 ± 0.02	$p = 78823$	1.12 ± 0.19	1.21 ± 0.58	1.33 ± 0.32
$\lvert I_{tr}\rvert = 50$	TPE	62.14 ± 6.92	0.24 ± 0.03	0.24 ± 0.02	$\lvert I_{tr}\rvert = 25$	1.64 ± 0.08	1.19 ± 0.55	1.26 ± 0.09
$\lvert I_{val}\rvert = 50$	IGJO	18.10 ± 2.77	0.24 ± 0.05	0.22 ± 0.03	$\lvert I_{val}\rvert = 25$	0.47 ± 0.12	0.52 ± 0.15	0.71 ± 0.19
$\lvert I_{te}\rvert = 5900$	VF-iDCA	12.85 ± 2.25	0.00 ± 0.00	0.19 ± 0.01	$\lvert I_{te}\rvert = 50$	0.76 ± 0.17	0.25 ± 0.11	0.55 ± 0.06
	LDMMA	10.99 ± 0.87	0.00 ± 0.00	0.20 ± 0.01		0.20 ± 0.05	0.25 ± 0.12	0.51 ± 0.09
	LDPM	$\mathbf{5.69 \pm 0.95}$	0.14 ± 0.03	$\mathbf{0.18 \pm 0.01}$		$\mathbf{0.20 \pm 0.03}$	0.31 ± 0.05	$\mathbf{0.49 \pm 0.05}$

Table 3: Sparse Group Lasso problem on datasets gisette and sensit, where $|I_{tr}|$, $|I_{val}|$, $|I_{te}|$ and p represent the number of training samples, validation samples, test samples and features, respectively.

Settings	Methods	Time(s)	Val. Err.	Test Err.	Settings	Time(s)	Val. Err.	Test Err.
gisette	Grid	62.87 ± 5.65	0.34 ± 0.05	0.35 ± 0.04	sensit	26.13 ± 4.72	1.39 ± 0.32	1.42 ± 0.38
$p = 5000$	Random	63.25 ± 6.10	0.32 ± 0.04	0.33 ± 0.03	$p = 78823$	29.38 ± 4.92	1.47 ± 0.59	1.37 ± 0.49
$\lvert I_{tr}\rvert = 50$	TPE	60.28 ± 9.43	0.32 ± 0.03	0.31 ± 0.04	$\lvert I_{tr}\rvert = 25$	38.60 ± 6.59	0.93 ± 0.37	1.03 ± 0.45
$\lvert I_{val}\rvert = 50$	IGJO	31.16 ± 5.81	0.28 ± 0.03	0.27 ± 0.04	$\lvert I_{val}\rvert = 25$	29.88 ± 3.75	0.97 ± 0.38	0.83 ± 0.31
$\lvert I_{te}\rvert = 5900$	VF-iDCA	16.30 ± 3.87	0.10 ± 0.02	0.25 ± 0.01	$\lvert I_{te}\rvert = 50$	16.46 ± 6.72	0.43 ± 0.19	0.52 ± 0.11
$M = 10$	LDMMA	25.86 ± 4.46	0.30 ± 0.03	0.32 ± 0.03	$M = 10$	7.28 ± 1.62	0.47 ± 0.10	0.64 ± 0.17
	GAFFA	10.17 ± 3.62	0.26 ± 0.03	0.29 ± 0.04		6.93 ± 1.68	0.60 ± 0.21	0.66 ± 0.14
	LDPM	$\mathbf{7.35 \pm 0.84}$	0.20 ± 0.03	$\mathbf{0.25 \pm 0.02}$		$\mathbf{3.72 \pm 1.61}$	0.45 ± 0.11	$\mathbf{0.52 \pm 0.05}$

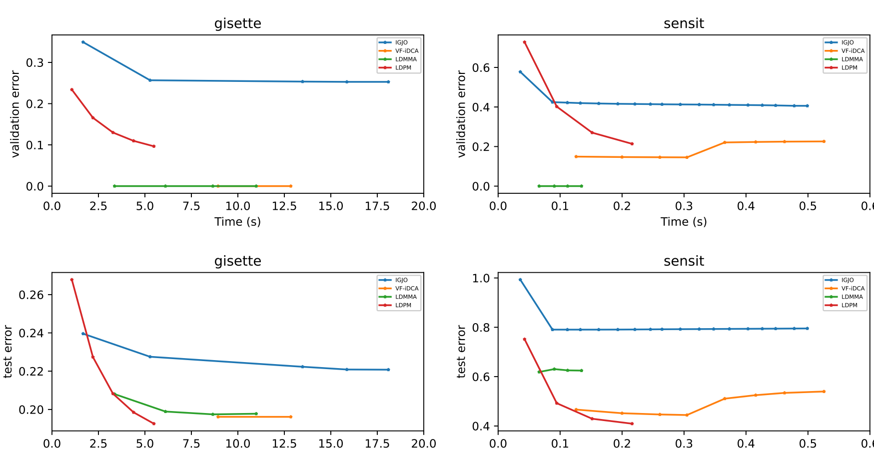

Figure 4: Comparison of the algorithms on Elastic Net problem for 2 datasets: gisette, sensit

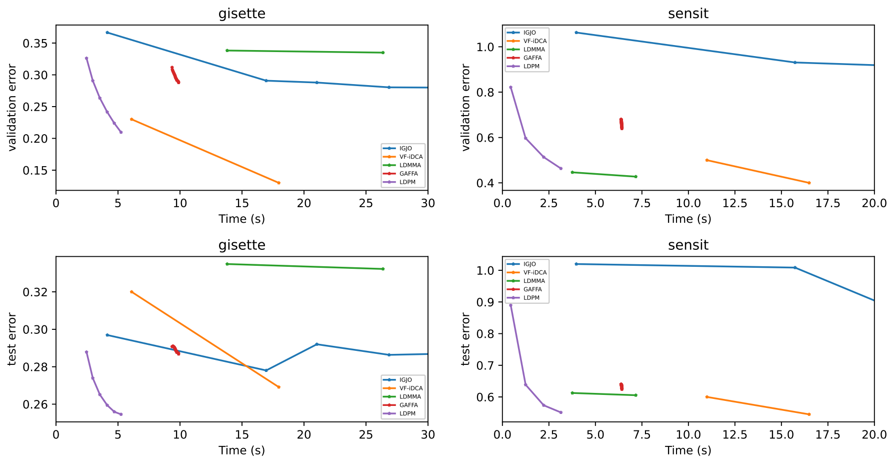

Figure 5: Comparison of the algorithms on Sparse Group Lasso problem for 2 datasets: gisette, sensit

Table 4: Elastic net problems on synthetic data, where $|I_{tr}|$, $|I_{val}|$, $|I_{te}|$ and p represent the number of training observations, validation observations, predictors and features, respectively.

Settings	Methods	Time(s)	Val. Err.	Test Err.	Settings	Time(s)	Val. Err.	Test Err.
	Grid	5.76 ± 0.33	7.05 ± 2.02	6.98 ± 1.14		11.72 ± 1.32	6.05 ± 1.47	6.49 ± 0.82
$\lvert I_{tr}\rvert = 100$	Random	5.74 ± 0.26	7.01 ± 2.01	7.01 ± 1.11	$\lvert I_{tr}\rvert = 100$	12.85 ± 2.11	6.04 ± 1.45	6.49 ± 0.83
$\lvert I_{val}\rvert = 20$	TPE	6.55 ± 0.26	7.01 ± 2.00	7.01 ± 1.09	$\lvert I_{val}\rvert = 100$	13.92 ± 1.72	6.03 ± 1.44	6.50 ± 0.83
$\lvert I_{te}\rvert = 250$	IGJO	1.54 ± 0.84	4.99 ± 1.69	5.42 ± 1.21	$\lvert I_{te}\rvert = 250$	3.37 ± 1.85	5.22 ± 1.50	5.72 ± 0.91
$p = 250$	VF-iDCA	3.16 ± 0.63	2.72 ± 1.57	5.18 ± 1.40	$p = 450$	6.08 ± 2.24	3.13 ± 0.78	5.39 ± 0.92
	LDMMA	1.64 ± 0.07	0.00 ± 0.00	6.97 ± 0.79		3.95 ± 0.22	0.00 ± 0.00	6.56 ± 0.70
	LDPM	$\mathbf{0.60 \pm 0.02}$	2.56 ± 0.80	$\mathbf{4.92 \pm 0.51}$		$\mathbf{1.02 \pm 0.03}$	3.42 ± 0.39	$\mathbf{4.23 \pm 0.37}$
	Grid	6.09 ± 0.60	6.39 ± 1.09	6.27 ± 1.02		32.99 ± 3.81	7.81 ± 1.53	8.82 ± 0.92
$\lvert I_{tr}\rvert = 100$	Random	6.44 ± 1.28	4.39 ± 1.10	6.27 ± 1.05	$\lvert I_{tr}\rvert = 100$	33.82 ± 2.66	6.44 ± 1.53	8.67 ± 0.94
$\lvert I_{val}\rvert = 100$	TPE	7.28 ± 1.23	6.37 ± 1.09	6.29 ± 1.09	$\lvert I_{val}\rvert = 100$	42.70 ± 3.89	7.71 ± 1.32	8.43 ± 0.80
$\lvert I_{te}\rvert = 250$	IGJO	3.86 ± 2.09	4.41 ± 0.98	4.31 ± 0.95	$\lvert I_{te}\rvert = 100$	31.30 ± 6.41	7.78 ± 1.12	8.61 ± 0.82
$p = 250$	VF-iDCA	4.74 ± 1.77	2.35 ± 1.56	4.47 ± 1.11	$p = 2500$	23.21 ± 4.96	0.00 ± 0.00	4.61 ± 0.77
	LDMMA	0.98 ± 0.09	0.00 ± 0.00	5.61 ± 0.77		16.26 ± 1.44	0.00 ± 0.00	5.67 ± 1.21
	LDPM	$\mathbf{0.73 \pm 0.08}$	3.41 ± 0.48	$\mathbf{3.51 \pm 0.40}$		$\mathbf{4.83 \pm 0.08}$	1.65 ± 0.14	$\mathbf{4.37 \pm 0.65}$

Table 5: Sparse group lasso problems on synthetic data, where p and M represent the number of covariates and covariate groups, respectively, and n represent the data scale described above.

Settings	Methods	Time(s)	Val. Err.	Test Err.	Settings	Time(s)	Val. Err.	Test Err.
	Grid	96.36 ± 2.88	44.73 ± 5.29	47.34 ± 5.91		103.68 ± 5.49	44.68 ± 4.31	46.00 ± 4.43
$n = 300$	Random	83.02 ± 3.01	35.17 ± 5.95	47.43 ± 5.54	$n = 450$	108.64 ± 8.84	37.87 ± 4.21	46.25 ± 5.52
$p = 600$	IGJO	117.58 ± 7.28	31.93 ± 4.07	46.36 ± 3.72	$p = 900$	120.35 ± 6.46	30.56 ± 4.02	46.70 ± 4.01
$M = 30$	VF-iDCA	19.00 ± 0.55	26.96 ± 2.58	36.84 ± 5.33	$M = 60$	29.63 ± 2.91	26.38 ± 3.40	37.58 ± 5.90
	LDMMA	24.62 ± 0.13	22.70 ± 2.03	31.44 ± 4.72		22.72 ± 2.15	23.93 ± 2.32	31.03 ± 4.08
	GAFFA	2.59 ± 0.02	27.42 ± 3.28	28.45 ± 4.74		11.52 ± 0.79	22.21 ± 3.03	29.81 ± 4.66
	LDPM	$\mathbf{1.26 \pm 0.03}$	$\mathbf{15.11 \pm 1.62}$	$\mathbf{23.48 \pm 2.40}$		$\mathbf{1.95 \pm 0.04}$	$\mathbf{19.39 \pm 1.51}$	$\mathbf{25.11 \pm 2.35}$
	Grid	104.23 ± 4.05	45.63 ± 4.13	44.86 ± 5.09		117.09 ± 6.34	48.94 ± 4.11	49.41 ± 7.62
$n = 300$	Random	98.17 ± 6.85	40.04 ± 5.36	46.77 ± 6.70	$n = 600$	126.3 ± 5.57	49.41 ± 6.55	52.49 ± 9.46
$p = 900$	IGJO	117.14 ± 7.44	31.59 ± 4.97	45.98 ± 4.94	$p = 1200$	169.76 ± 9.44	39.75 ± 5.14	46.49 ± 7.48
$M = 60$	VF-iDCA	44.31 ± 1.45	23.21 ± 3.36	31.92 ± 3.54	$M = 150$	45.12 ± 3.10	23.66 ± 4.53	35.09 ± 3.14
	LDMMA	37.50 ± 0.21	26.23 ± 3.47	32.09 ± 3.75		36.14 ± 3.65	18.61 ± 2.32	27.81 ± 3.43
	GAFFA	5.11 ± 0.10	26.83 ± 3.53	30.38 ± 3.60		33.03 ± 4.63	24.34 ± 4.19	26.05 ± 5.13
	LDPM	$\mathbf{1.87 \pm 0.05}$	$\mathbf{19.32 \pm 2.62}$	$\mathbf{27.14 \pm 2.79}$		$\mathbf{3.08 \pm 0.07}$	$\mathbf{17.35 \pm 2.04}$	$\mathbf{24.21 \pm 2.74}$

Table 6: Group lasso problems on synthetic data, where p and M represent the number of covariates and covariate groups, respectively, and n represent the data scale described above.

Settings	Methods	Time(s)	Val. Err.	Test Err.	Settings	Time(s)	Val. Err.	Test Err.
	Grid	85.18 ± 4.61	45.33 ± 6.79	48.84 ± 6.76		100.91 ± 7.80	45.38 ± 5.74	48.19 ± 6.69
$n = 300$	Random	79.11 ± 5.10	37.92 ± 5.13	45.66 ± 6.34	$n = 450$	93.06 ± 6.72	45.18 ± 7.41	43.86 ± 4.89
$p = 600$	IGJO	99.01 ± 9.41	34.86 ± 5.80	45.87 ± 4.67	$p = 900$	94.22 ± 7.79	38.75 ± 7.72	43.99 ± 5.30
$M = 30$	VF-iDCA	9.70 ± 2.30	27.21 ± 5.37	32.95 ± 7.16	$M = 60$	21.14 ± 6.22	24.07 ± 2.20	36.15 ± 6.01
	LDMMA	27.02 ± 2.52	25.76 ± 3.60	34.74 ± 4.34		38.80 ± 4.59	26.95 ± 4.33	33.69 ± 6.17
	GAFFA	3.56 ± 0.11	29.73 ± 6.48	31.22 ± 5.88		10.88 ± 0.63	25.84 ± 7.19	29.74 ± 6.43
	LDPM	$\mathbf{0.55 \pm 0.04}$	$\mathbf{17.42 \pm 3.74}$	$\mathbf{25.10 \pm 3.68}$		$\mathbf{0.91 \pm 0.03}$	$\mathbf{19.20 \pm 5.11}$	$\mathbf{22.28 \pm 4.28}$
	Grid	107.95 ± 10.36	46.13 ± 5.54	46.21 ± 7.94		128.77 ± 9.68	45.33 ± 6.43	47.32 ± 7.24
$n = 300$	Random	95.02 ± 7.27	43.66 ± 6.31	42.18 ± 6.77	$n = 600$	131.50 ± 11.36	48.79 ± 7.66	48.91 ± 9.13
$p = 900$	IGJO	122.64 ± 9.96	30.56 ± 6.46	47.36 ± 5.76	$p = 1200$	152.10 ± 15.19	37.21 ± 6.89	42.30 ± 7.59
$M = 60$	VF-iDCA	9.12 ± 0.07	24.40 ± 5.62	30.25 ± 4.03	$M = 150$	67.71 ± 9.53	27.53 ± 5.16	35.61 ± 6.98
	LDMMA	38.13 ± 3.41	24.94 ± 6.68	30.12 ± 4.85		47.11 ± 5.86	18.51 ± 4.09	27.58 ± 4.19
	GAFFA	5.17 ± 0.17	28.39 ± 6.22	29.95 ± 5.23		34.88 ± 9.98	25.39 ± 5.41	26.81 ± 5.39
	LDPM	$\mathbf{0.86 \pm 0.02}$	$\mathbf{20.69 \pm 3.88}$	$\mathbf{27.04 \pm 4.58}$		$\mathbf{1.83 \pm 0.02}$	$\mathbf{19.18 \pm 5.03}$	$\mathbf{25.35 \pm 6.27}$