A Variational Analysis Approach for Bilevel Hyperparameter Optimization with Sparse Regularization

Pedro Pérez-Aros[1†], Emilio Vilches[2†], David Villacís[3*†]

[1]Departamento de Ingeniería Matemática and Centro de Modelamiento Matemático (CNRS UMI 2807), Universidad de Chile, Santiago, Chile.
[2]Instituto de Ciencias de la Ingeniería, Universidad de O'Higgins, Rancagua, Chile.
[3*]Departamento de Métodos Cuantitativos, Universidad Loyola Andalucía, Seville, Spain.

*Corresponding author(s). E-mail(s): davillacis@uloyola.es;
Contributing authors: pperez@dim.uchile.cl; emilio.vilches@uoh.cl;
†These authors contributed equally to this work.

Abstract

We study a bilevel optimization framework for hyperparameter learning in variational models, with a focus on sparse regression and classification tasks. In particular, we consider a weighted elastic-net regularizer, where feature-wise regularization parameters are learned through a bilevel formulation. A key novelty of our approach is the use of a Forward-Backward (FB) reformulation of the nonsmooth lower-level problem while preserving its set of minimizers. This reformulation yields a bilevel objective composed with a locally Lipschitz solution map, allowing the application of generalized subdifferential techniques to derive calculus rules and enable efficient subgradient-based optimization methods. Crucially, this coderivative-based construction provides nonzero subgradient information at biactive coordinates, where standard implicit-differentiation methods suffer from gradient starvation. Empirical results on synthetic, semi-synthetic and real sparse classification datasets demonstrate that our approach improves support recovery in degenerate sparse regimes while remaining competitive in predictive performance.

Keywords: Bilevel Optimization, Variational Analysis, Nonsmooth Analysis, Sparse Regression and Classification Models

MSC Classification: 90C26 , 49J53 , 65K10 , 90C30 , 62J07

1 Introduction

In this work, we focus on a class of models formulated using a variational framework, which balances data fidelity with prior knowledge through a regularization term. The general model takes the form:

$$\min_{y \in \mathcal{Y}} \quad F(y; d) + G_x(y),$$

where $y \in \mathcal{Y}$ denotes the vector of unknown model parameters to be estimated, $F(y; d)$ is the *data fidelity* term, which enforces consistency with the observations $d \in \mathcal{D}$, $G_x(y)$ is a *regularization term* encoding prior assumptions such as smoothness, sparsity, or piecewise constancy. This regularization term is parametrized by the so-called hyperparameter $x \in \mathcal{X}$ that balances the trade-off between data fidelity and regularization.

Furthermore, the choice of the regularization hyperparameter x is critical to the success of the variational model. If x is too small, the solution may overfit the data, leading to poor generalization or reconstructions that amplify noise. Conversely, if it is too large, the regularization may dominate, resulting in oversmoothed or biased solutions that fail to capture relevant features of the underlying signal or structure.

Traditionally, this selection is handled using methods such as cross-validation (CV) or Bayesian optimization. In CV, one evaluates the out-of-sample performance across a predefined grid of parameter values, typically using techniques like K-fold CV to estimate the validation error. While effective in low-dimensional settings, this brute-force strategy becomes computationally expensive as the number of hyperparameters increases. Beyond two or three parameters, grid and random search become infeasible.

In such cases, Bayesian optimization offers a more adaptive alternative by modeling performance as a probabilistic function of the hyperparameters [1, 2]. However, these gradient-free methods also face scalability limits, typically struggling when the number of tunable parameters exceeds 10 to 20, see e.g. [3].

A compelling alternative is to cast hyperparameter selection as a bilevel optimization problem. In this framework, hyperparameters are treated as variables in an outer (upper-level) optimization problem that minimizes validation error, while an inner (lower-level) problem fits the model using those hyperparameters. This nested formulation mirrors a Stackelberg game, where the leader (outer problem) selects parameters to optimize a generalization estimator, and the follower (inner problem) solves the task-specific model accordingly. This approach enables principled, scalable, and potentially gradient-based optimization of hyperparameters–even in high-dimensional settings.

We consider the following *Bilevel Hyperparameter Learning* (BHL) framework:

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & L(x, y^*(x)) \\ \text{s.t.} \quad & y^*(x) \in S(x) := \arg\min_{y \in \mathcal{Y}} \{F(y) + G_x(y)\}. \end{aligned}$$

Unlike gradient-free approaches, gradient-based hyperparameter optimization scales more naturally to high-dimensional settings, enabling the tuning of dozens or even hundreds of hyperparameters simultaneously. The numerical solution methods typically fall into two broad categories, depending on how the gradient of the upper-level objective is computed. In the smooth setting, this quantity is commonly referred to as the *hypergradient*; in the nonsmooth framework developed here, the analogous object is a subgradient of the upper-level objective, and we will use this terminology throughout.

The first category, known as implicit gradient methods, treats the lower-level optimization problem as defining an implicit mapping from hyperparameters to model parameters. By applying the implicit function theorem to the optimality conditions of the lower-level problem, one can derive the hypergradient analytically. This typically requires solving a linear adjoint system [4] or, in large-scale settings, approximating the inverse Hessian-vector product through iterative schemes such as Neumann series expansions [5, 6]. Applications include validation-based model selection for support vector machines and kernel classifiers, where regularization or kernel parameters are selected by differentiating the KKT system of the training problem [7, 8]. More recently, implicit differentiation has been used to optimize very high-dimensional hyperparameter vectors [9] and to compute meta-gradients without backpropagating through long inner optimization trajectories [10]. However, these approaches typically rely on smoothness and suitable nonsingularity or strong-regularity assumptions for the lower-level optimality system.

In contrast, explicit gradient methods approximate the solution to the lower-level problem using a fixed number of iterative updates (e.g., gradient descent steps). The hypergradient is then computed by differentiating through this iterative process, using either forward-mode or reverse-mode automatic differentiation [11, 12]. While more flexible and easier to implement, explicit methods often require more memory and computational effort, especially when the number of inner iterations is large.

1.1 On the sparse regularization of machine learning models

Sparse regularization has become a cornerstone in modern machine learning, where high dimensionality and ill-posedness are common challenges. The central idea is to promote solutions that rely on a small subset of relevant features or components, enhancing both interpretability and generalization.

In machine learning, sparsity-inducing regularizers are widely used to perform implicit feature selection. By penalizing the number of non-zero coefficients, sparse regularization encourages models that are simpler and more robust, often outperforming dense alternatives in high-dimensional regimes.

A wide range of regularizers have been proposed to achieve this sparsity effect. The most direct approach is the ℓ_0 pseudo-norm, which counts the number of nonzero

entries in a vector but leads to combinatorial NP-hard problems. To address this, the convex ℓ_1-norm is widely adopted as a tractable surrogate that promotes sparsity and enables efficient optimization [13]. The elastic-net regularizer combines ℓ_1 and ℓ_2 penalties to enhance stability in the presence of correlated features, balancing sparsity and shrinkage. Beyond convex methods, nonconvex penalties such as MCP [14], SCAD [15], and ℓ_p-quasi-norms (with $0 < p < 1$) provide closer approximations to the ℓ_0 norm and mitigate bias on large coefficients, though at the cost of increased optimization complexity. In settings where structure matters, group sparsity and variants like the group Lasso [16] and fused Lasso [17] regularization are used to enforce zero patterns aligned with known feature groupings or spatial arrangements.

When it comes to sparsity-regularized lower-level problems, applying the bilevel optimization framework becomes significantly more challenging. This is because both implicit and explicit gradient methods typically rely on two key assumptions: strong convexity and smoothness of the lower-level problem. However, these assumptions break down in the context of sparsity-inducing regularizers, which are inherently nonsmooth by design. As a result, standard bilevel techniques cannot be directly applied, and more careful theoretical and algorithmic treatment is required to handle these nonsmooth settings effectively.

The numerical solution of bilevel problems with nonsmooth lower-level objectives has attracted growing attention in recent years, leading to a range of strategies across different applications. A common approach, particularly in inverse problems, is to approximate the nonsmooth regularization with smooth surrogates. This idea is explored in the seminal works [18, 19], where bilevel formulations are solved using smoothed versions of nonsmooth terms like total variation. More recent developments have extended this line of research to nonsmooth formulations without smoothing, as seen in [20, 21], specifically addressing total variation regularization via tailored optimality conditions.

A distinct line of research focuses on leveraging the structural properties of bilevel problems to manage nonsmoothness more effectively. For example, [22] reformulates the bilevel problem as a difference-of-convex (DC) program and addresses it through a value function approach. This direction is further developed in [23], where the Moreau envelope is applied to the value function to obtain a smooth approximation. Meanwhile, [24] addresses a nonsmooth support vector machine by exploiting the duality of the lower-level problem, resulting in a mathematical program with equilibrium constraints (MPEC)—an approach that builds on foundational work from [25, 26].

Another line of work embraces the unrolled optimization paradigm, as in [27], where the lower-level problem is replaced by a fixed number of iterations of a primal-dual algorithm and differentiated through. Finally, bundle methods, traditionally used in nonsmooth optimization, have also been adapted to bilevel settings, as demonstrated in [28].

The closest prior work is the `Sparse-HO` framework of [29], which applies implicit differentiation to sparse bilevel learning by restricting the adjoint system to the primal support—an approach that is exact under strict complementarity but assigns zero subgradient to biactive coordinates where the optimality gap also vanishes. The present paper addresses this limitation through an implicit-gradient framework for

bilevel hyperparameter optimization with sparse lower-level regularization. Rather than smoothing the lower-level problem, we use a Forward–Backward (FB) reformulation that preserves the original solution set and yields a structured residual mapping. This reformulation allows us to bring tools from variational analysis into the sparse lower-level setting and to derive computable subgradients for the upper-level problem, including degenerate biactive regimes beyond strict complementarity. The structure of the paper is outlined as follows:

- Section 2 presents the key concepts and results from convex and variational analysis that underpin the developments in this work. These tools form the theoretical foundation for the analysis and algorithms introduced later.
- Section 3 contains the theoretical core. Subsection 3.1 introduces the FB reformulation of the lower-level problem, establishing that it preserves the original minimizers and yields a single-valued, locally Lipschitz solution map S. Subsection 3.2 recasts the bilevel problem as the single-level composition $\Phi(x) = L(x, S(x))$ and reduces the computation to a coderivative problem for S. The section concludes with a full coderivative calculus for the weighted ℓ_1 case, treating the soft-thresholding operator $\mathcal{T}$ and the FB residual R^γ, and delivering a computable subgradient formula valid at biactive coordinates.
- Section 4 addresses the computational realization of the framework. It introduces the self-consistent biactive selection policy (Definition 2) and a support-reduced subgradient oracle (Algorithm 1) that restricts the adjoint system to the active working set $\mathcal{S}$, compressing the linear solve from $\mathcal{O}(p^3)$ to $\mathcal{O}(|\mathcal{S}|^3)$. Two outer-level solvers embed this oracle: Algorithm 2 (`NBA-`$w\ell_1$), a projected normalized subgradient method, and Algorithm 3 (`NTRBA-`$w\ell_1$), a trust-region variant whose radius is adapted via a predicted-versus-actual reduction ratio.
- In Section 5, we evaluate the proposed framework on an illustrative exactness-and-gradient-starvation example, synthetic sparse-regression studies, a controlled oracle–optimizer ablation, comparisons against scalar tuning and `Sparse-HO`, and high-dimensional classification benchmarks.

2 Preliminaries from Convex and Variational Analysis

We collect here the definitions and results from convex and variational analysis used throughout the paper. For a comprehensive treatment of these topics, we refer the reader to the following monographs [30–32].

In what follows $\mathcal{X}$ and $\mathcal{Y}$ represent two finite dimensional Hilbert spaces. To simplify the notation, we denote the usual Euclidean norm by $\|\cdot\|$ with the corresponding standard inner product as $\langle\cdot,\cdot\rangle$ in all the spaces. We use $\mathbb{R}^n$ to represent the n-dimensional Euclidean space, and the extended real line $\overline{\mathbb{R}} := [-\infty, \infty]$.

2.1 Elements of Convex Analysis

Given a nonempty set $C \subset \mathcal{X}$, we denote the polar cone of C by

$$C^\circ := \{v \in \mathcal{X} \mid \langle v, w\rangle \leq 0, \text{ for all } w \in C\}.$$

Let $\psi : \mathcal{X} \to \overline{\mathbb{R}}$ be a convex, proper, and lower semicontinuous function, and let $\gamma > 0$ be a fixed parameter. We define the proximal operator as follows:

$$\operatorname{prox}_{\gamma\psi}(x) := \operatorname*{arg\,min}_{y \in \mathcal{X}} \left\{ \psi(y) + \frac{1}{2\gamma}\|x - y\|^2 \right\}.$$

2.2 Generalized Differentiation

Given a closed set $C \subset \mathcal{X}$, we begin by defining the *tangent cone* (also known as contingent or Bouligand cone) to a set C at a point $\bar{x} \in C$ by

$$T(\bar{x}; C) = \left\{ d \in \mathcal{X} \;\middle|\; \exists\, t_k \downarrow 0, \exists\, x_k \in C \text{ s.t. } x_k \to \bar{x} \text{ and } \frac{x_k - \bar{x}}{t_k} \to d \right\}.$$

This cone captures the directions d in which the set C can be locally approximated near $\bar{x}$. Furthermore, the *Fréchet normal cone* to C at $\bar{x} \in C$ is given by $\widehat{N}(\bar{x}; C) := T(\bar{x}; C)^\circ$, where $T(\bar{x}; C)^\circ$ denotes the polar cone of $T(\bar{x}; C)$. Moreover, the *Mordukhovich (limiting) normal cone* to C at $\bar{x} \in C$ is defined as

$$N(\bar{x}; C) = \left\{ x^* \in \mathcal{X} \;\middle|\; \exists\, x_k^* \in \widehat{N}(x_k; C) \text{ such that } (x_k, x_k^*) \to (\bar{x}, x^*) \right\}. \tag{1}$$

Since it is defined as a limit of Fréchet normals, we always have the inclusion:

$$\widehat{N}(\bar{x}; C) \subset N(\bar{x}; C), \quad \text{for all } \bar{x} \in \mathcal{X}.$$

Whenever C is convex, both constructions coincide with the normal cone of convex analysis, that is,

$$\widehat{N}(\bar{x}; C) = N(\bar{x}; C) = \{x^* : \langle x^*, x - \bar{x} \rangle \leq 0, \text{ for all } x \in C\}, \quad \text{for all } \bar{x} \in \mathcal{X}.$$

For a proper lower-semicontinuous function $f : \mathcal{X} \to \overline{\mathbb{R}}$, the *Fréchet* and *Mordukhovich subdifferentials* at $\bar{x}$ are respectively defined as

$$\hat{\partial} f(\bar{x}) := \left\{ x^* \in \mathcal{X} \;\middle|\; (x^*, -1) \in \widehat{N}((\bar{x}, f(\bar{x})); \operatorname{epi} f) \right\},$$
$$\partial f(\bar{x}) := \{ x^* \in \mathcal{X} \,|\, (x^*, -1) \in N((\bar{x}, f(\bar{x})); \operatorname{epi} f) \}.$$

Moreover, the normal cone constructions also enable the definition of generalized derivatives for set-valued mappings. Specifically, for a multifunction $F : \mathcal{X} \rightrightarrows \mathcal{Y}$, the *Mordukhovich coderivative* of F at a point $(x, y) \in \operatorname{gph} F := \{(x, y) \mid y \in F(x)\}$ is defined for each $y^* \in \mathcal{Y}$ as

$$D^* F(x, y)(y^*) = \{ x^* \in \mathcal{X} \,|\, (x^*, -y^*) \in N((x, y); \operatorname{gph} F) \}.$$

When $F(x) = \{y\}$ is a singleton, the point y is often omitted from the notation. Moreover, if F is continuously differentiable at x, the coderivative simplifies to

$D^*F(x)(y^*) = \{JF(x)^*y^*\}$, where $JF(x)^*$ denotes the adjoint of the Jacobian of F at x. Additionally, when F is single-valued and locally Lipschitzian around x, the coderivative can be expressed in terms of the Mordukhovich subdifferential of the scalarization as follows (see, e.g., [31, Theorem 1.32]):

$$D^*F(x)(y^*) = \partial\langle y^*, F\rangle(x), \qquad \text{where } \langle y^*, F\rangle := \langle y^*, F(x)\rangle. \tag{2}$$

Let us recall that a mapping $F : \mathcal{X} \rightrightarrows \mathcal{Y}$ is said to be lower-regular at (x, y) provided that

$$D^*F(x,y)(y^*) = \left\{x^* \in \mathcal{X} \mid (x^*, -y^*) \in \widehat{N}((x,y); \operatorname{gph} F)\right\}, \text{ for all } y^* \in \mathcal{Y}.$$

As is standard in set-valued analysis, we identify single-valued set-valued mappings with ordinary (single-valued) functions.

Similar to the coderivative, the Mordukhovich subdifferential of the scalarization is equipped with a set of calculus rules. The specific rules relevant to our work are summarized in the following lemma.

Lemma 1 *Let $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z}$ be finite dimensional Hilbert spaces. Let $\bar{x} \in \mathcal{X}$, let $F_1 : \mathcal{X} \to \mathcal{Y}$ be continuously differentiable around $\bar{x}$ with $\nabla F_1(\bar{x})$ of full rank. Furthermore, let $F_2 : \mathcal{X} \to \mathcal{Y}$ and $F_3 : \mathcal{Y} \to \mathcal{Z}$ locally Lipschitz continuous around $\bar{x}$ and $\bar{y} = F_1(\bar{x})$, respectively. Then, for any $z^* \in \mathcal{Z}$ and $y^* \in \mathcal{Y}$, we have:*

(i) The coderivative of the sum $F_1 + F_2$ at $\bar{x}$ for $y^* \in \mathcal{Y}$ reads:

$$D^*\left(F_1 + F_2\right)(\bar{x})(y^*) = \nabla F_1(\bar{x})^\top y^* + D^*F_2(\bar{x})(y^*).$$

(ii) For $\bar{y} = F_1(\bar{x})$ the coderivative of the composition $(F_3 \circ F_1)(\bar{x})$ at $\bar{x}$ for $z^* \in \mathcal{Z}$ is:

$$D^*\left(F_3 \circ F_1\right)(\bar{x})(z^*) = \nabla F_1(\bar{x})^\top \circ D^*F_3(\bar{y})(z^*).$$

(iii) The coderivative of $-F_2$ at $\bar{x}$ for $y^* \in \mathcal{Y}$ reads:

$$D^*(-F_2)(\bar{x})(y^*) = D^*F_2(\bar{x})(-y^*).$$

Proof Let us begin with item (i). The coderivative sum rule, as presented in [31, Theorem 3.9 (i)], offers a precise characterization for the coderivative of the sum of two multifunctions– provided that the following qualification condition is satisfied:

$$D^*F_1(\bar{x})(0) \cap (-D^*F_2(\bar{x})(0)) = \{0\}.$$

This condition is automatically met when F_1 is continuously differentiable and its Jacobian matrix $\nabla F_1(\bar{x})$ has full rank. Moreover, since both F_1 and F_2 are single-valued mappings, the conclusion follows directly.

Next, for item (ii), we apply the coderivative chain rule from [31, Theorem 3.11 (iii)], which characterizes the coderivative of a composition. This result holds under the assumption $\nabla F_1(\bar{x})$ is of full rank.

Finally, item (iii) follows directly from (2). □

3 Bilevel hyperparameter learning for sparse models

We define the *Nonsmooth Bilevel Hyperparameter Learning* problem (NBHL) as the following bilevel optimization problem:

$$\min_{x \in \mathcal{X}} \quad L(x, y^*(x)) \tag{3a}$$

$$\text{s.t.} \quad y^*(x) \in S(x) := \arg\min_{y \in \mathcal{Y}} \{\varphi_x(y) := F(y) + G_x(y)\}, \tag{3b}$$

where $L : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, $F : \mathcal{Y} \to \mathbb{R}$ and $G_x : \mathcal{Y} \to \mathbb{R}$ are *convex but not necessarily differentiable* and parametrized for a given $x \in \mathcal{X}$. Moreover, to ensure that (3) is well-defined, we will consider the following assumptions:

Assumption 1 The function F is twice continuously differentiable and Λ_F-smooth, that is,

$$\|\nabla F(y) - \nabla F(\tilde{y})\| \leq \Lambda_F \|y - \tilde{y}\|, \quad \forall y, \tilde{y} \in \mathcal{Y}.$$

Assumption 2 The function F is strongly convex with constant μ_F, i.e.,

$$F(y) \geq F(z) + \langle \nabla F(z),\, y - z \rangle + \mu_F \|y - z\|^2 \quad \text{for all } y, z \in \mathcal{Y}.$$

Assumption 3 The function $G_x(y) = \sum_{i=1}^p \psi(x_i) g_i(y)$ where each $g_i \colon \mathcal{Y} \to \mathbb{R}$ is a finite-valued convex function and $\psi \colon \mathbb{R} \to \mathbb{R}_+$ is a continuously differentiable function. Moreover, $\psi'(t) > 0$ for all $t \in \mathbb{R}$.

We refer to L as the *upper-level* objective function and φ_x as the *lower-level* objective function, parameterized by a given upper-level variable $x \in \mathcal{X}$. Solving (3) is particularly challenging due to the implicit nature of the lower-level optimal solution mapping $S \colon \mathcal{X} \rightrightarrows \mathcal{Y}$, as defined in (3b). Nonetheless, Theorem 2 establishes that under our assumptions, the solution set S is single-valued and a locally Lipschitz mapping. This property serves as a key starting point for analyzing the behavior of the lower-level problem using techniques from generalized differentiation.

Theorem 2 *Under assumptions 2 and 3, the set $S(x)$ is single-valued for every $x \in \mathcal{X}$. Moreover, the mapping $S \colon \mathcal{X} \to \mathcal{Y}$ is locally Lipschitz.*

Proof First, we observe that the mapping $(x, y) \mapsto \varphi_x(y)$ is continuous. Indeed, F is continuous by its strong convexity, each g_i is finite-valued convex and therefore continuous on the finite-dimensional space $\mathcal{Y}$, and ψ is continuously differentiable; hence $(x, y) \mapsto F(y) + \sum_{i=1}^p \psi(x_i) g_i(y)$ is continuous. Furthermore, for any fixed $x \in \mathcal{X}$, the lower-level objective function φ_x, as defined in (3b), is strongly convex. Consequently, the corresponding problem admits a unique solution.

Next, we show that the solution mapping S is continuous. Consider a sequence $x_k \to x$. It follows that

$$\eta_i := \sup \{\psi((x_k)_i) : k \in \mathbb{N}\} < +\infty, \text{ for all } i = 1, \ldots, p.$$

In particular, it implies that

$$\varphi_{x_k}(y) \leq F(y) + \sum_{i=1}^{p} \eta_i \max\{g_i(y), 0\} \text{ for all } y \in \mathcal{Y}, \text{ and all } k \in \mathbb{N}. \tag{4}$$

Since each function g_i is convex, there exist constants $\alpha_i, \beta_i \in \mathbb{R}$ such that

$$g_i(y) \geq \alpha_i \|y\| + \beta_i \quad \text{for all } y \in \mathcal{Y}, \text{ and for all } i = 1, \ldots, p.$$

Moreover, the strong convexity of F yields that there exist constants $a > 0$ and $b, c \in \mathbb{R}$ such that

$$\varphi_{x_k}(y) \geq a\|y\|^2 + b\|y\| + c, \text{ for all } y \in \mathcal{Y} \text{ and all } k \in \mathbb{N}. \tag{5}$$

Therefore, by combining (4) and (5), we conclude that the sequence $\{S(x_k)\}_{k\in\mathbb{N}}$ is bounded and thus admits at least one accumulation point. Moreover, by the definition of $S(x_k)$, we have

$$\varphi_{x_k}(S(x_k)) \leq \varphi_{x_k}(y) \quad \text{for all } y \in \mathcal{Y}.$$

By continuity of the function $(x, y) \mapsto \varphi_x(y)$, any accumulation point of the sequence $\{S(x_k)\}_{k\in\mathbb{N}}$ is a solution to the lower-level problem at x. Since this solution is unique, it follows that $S(x_k) \to S(x)$. This concludes the proof of the continuity of S.

Now, let us consider $u, v \in \mathcal{X}$. Using the strong convexity of the function φ_u, we obtain

$$\begin{aligned}
\mu_F\|S(u) - S(v)\|^2 \leq& \varphi_u(S(v)) - \varphi_u(S(u)) \\
\leq& \varphi_u(S(v)) - \varphi_v(S(v)) + \varphi_v(S(u)) - \varphi_u(S(u)) \\
=& \sum_{i=1}^{p} \big(\psi(u_i) - \psi(v_i)\big)\big(g_i(S(v)) - g_i(S(u))\big) \\
\leq& \sum_{i=1}^{p} |\psi(u_i) - \psi(v_i)| \cdot |g_i(S(u)) - g_i(S(v))|.
\end{aligned}$$

Now, fix an arbitrary point $\bar{x} \in \mathcal{X}$, and let U be a compact neighborhood of $\bar{x}$. Since S is continuous and U is compact, the image $S(U)$ is compact; hence each finite-valued convex g_i admits a common Lipschitz constant $\kappa > 0$ on $S(U)$. Then, for every $u, v \in U$, we have

$$\begin{aligned}
\mu_F\|S(u) - S(v)\|^2 &\leq \sum_{i=1}^{p} |\psi(u_i) - \psi(v_i)| \cdot |g_i(S(u)) - g_i(S(v))| \\
&\leq \kappa\|S(u) - S(v)\| \sum_{i=1}^{p} |\psi(u_i) - \psi(v_i)|,
\end{aligned}$$

which, by virtue of Assumption 3, implies that S is locally Lipschitz continuous. □

3.1 A Forward-Backward reformulation of the lower-level problem

Using the single-valuedness and Lipschitz continuity of the solution mapping established in the previous section, we now characterize the lower-level solution set through an alternative implicit representation. To this end, we introduce a Forward–Backward (FB) reformulation. Its role in our analysis is not to smooth the problem, but to provide an exact residual mapping that preserves the original solution set and is amenable to coderivative analysis.

Indeed, for $\gamma > 0$, let us introduce the Forward–Backward operator and residual, respectively, as follows (see, e.g., [33]):

$$T_x^\gamma(y) := T^\gamma(x,y) := \operatorname{prox}_{\gamma G_x}(y - \gamma \nabla F(y)),, \tag{6}$$

$$R_x^\gamma(y) := R^\gamma(x,y) := \gamma^{-1}(y - T_x^\gamma(y)).. \tag{7}$$

Given these definitions, we can characterize the solutions of the original lower-level problem (3b) for a fixed parameter $x \in \mathcal{X}$, as the zeros of the associated residual mapping. This well-established correspondence is formalized in the following proposition (cf. [34, Cor. 27.3]). Since the proof follows standard arguments in convex optimization, we omit it.

Proposition 3 *Under Assumptions 2 and 3, and assuming F is differentiable, let $\gamma > 0$, $x \in \mathcal{X}$, and $y \in \mathcal{Y}$. Then, the following statements are equivalent:*

$$(i)\ R_x^\gamma(y) = 0, \quad (ii)\ 0 \in \nabla F(y) + \partial G_x(y), \ \text{and}\ (iii)\ y \in S(x).$$

Let us emphasize that the previous proposition establishes the following implicit representation of the mapping S, that is, for a parameter $\gamma > 0$, we have that

$$S(x) := \{y \in \mathcal{Y} \mid R_x^\gamma(y) = 0\}. \tag{8}$$

3.2 A generalized composition reformulation of the NBHL problem

The results presented in the previous section allow us to reformulate the NBHL problem using the solution set of the lower-level problem, defined in (3b). Furthermore, the single-valuedness property established in Theorem 2 enables us to represent the bilevel optimization problem as an unconstrained problem with a locally Lipschitz continuous objective function via generalized composition, namely,

$$\min_{x \in \mathcal{X}} \Phi(x) := L(x, S(x)). \tag{9}$$

An important advantage of this single-level reformulation is that it allows us to apply the chain rule for the generalized composition of functions to effectively characterize the subdifferential of the upper-level function with respect to the parameter x. This characterization plays a pivotal role in the numerical solution of the bilevel problem using subgradient-based schemes. Specifically, to determine the limiting subdifferential, we rely on [31, Theorem 4.5], which provides the following result:

Lemma 4 *Under assumptions 2, 1 and 3, let L be strictly differentiable at $(\bar{x}, \bar{y})$ with $\bar{y} := S(\bar{x})$, then we have the equality:*

$$\partial \Phi(\bar{x}) = \nabla_x L(\bar{x}, \bar{y}) + D^* S(\bar{x})(\nabla_y L(\bar{x}, \bar{y})).$$

A key component in characterizing the limiting subdifferential of the bilevel problem is the coderivative of the solution mapping S. This task is nontrivial, as the mapping is given in an implicit form. However, we can use the results for coderivatives of implicit multifunctions [35, Corollary 4.34] to obtain the following estimate for the coderivative.

Lemma 5 *Under assumptions 1, 2 and 3, let* $\gamma > 0$, $\bar{x} \in \mathcal{X}$, $\bar{y} := S(\bar{x}) \in \mathcal{Y}$ *and* $R^\gamma : \mathcal{X} \times \mathcal{Y} \to \mathcal{Y}$. *Then,*

$$D^*S(\bar{x})(y^*) \subseteq \left\{x^* \in \mathcal{X} | (x^*, -y^*) \in D^*R^\gamma(\bar{x}, \bar{y})(z^*), \text{ for some } z^* \in \mathcal{Y}\right\}, \tag{10}$$

provided that

$$\ker D^*R^\gamma(\bar{x}, \bar{y}) = \{0\}. \tag{11}$$

Moreover, the inclusion in (10) *becomes an equality provided that* R^γ *is lower-regular at* $(\bar{x}, \bar{y})$.

Remark 1 It is important to mention that condition (11) is equivalent to metric regularity of R^γ around $(\bar{x}, \bar{y})$ (see, e.g., [36, Theorem 4C.2]).

Since the inclusion in Lemma 5 may be strict, it is convenient to name the right-hand side set explicitly, as it will play a central role in the subsequent analysis.

Definition 1 (Residual-enlarged coderivative and subdifferential) We define the *residual-enlarged coderivative* of S at $\bar{x}$ for $y^* \in \mathcal{Y}$ as the set:

$$D^*_{R^\gamma}S(\bar{x})(y^*) := \left\{x^* \in \mathcal{X} | (x^*, -y^*) \in D^*R^\gamma(\bar{x}, \bar{y})(z^*), \text{ for some } z^* \in \mathcal{Y}\right\},$$

and its corresponding *residual-enlarged subdifferential* is defined as:

$$\partial^\gamma_R\Phi(\bar{x}) := \nabla_x L(\bar{x}, \bar{y}) + D^*_{R^\gamma}S(\bar{x})(\nabla_y L(\bar{x}, \bar{y})).$$

By Lemmas 4 and 5, $\partial\Phi(\bar{x}) \subseteq \partial^\gamma_R\Phi(\bar{x})$, with equality when R^γ is lower-regular at $(\bar{x}, \bar{y})$.

Lemmas 4–5 and Definition 1 provide a tractable outer approximation $\partial^\gamma_R\Phi(\bar{x})$ of the limiting subdifferential of the NBHL problem. Elements of this set can be leveraged to design numerical algorithms for computing optimal solutions. Further development of these computations—particularly the coderivative calculus for R^γ and the verification of condition (11)—requires a more explicit representation of the NBHL problem, which will be addressed for the weighted ℓ_1 regularizer in the next section.

3.3 Subgradients for the NBHL problem with the weighted ℓ_1 regularizer at lower-level

In this section, we characterize the residual-enlarged subdifferential using the widely adopted weighted ℓ_1 regularizer. This regularizer is particularly important because it promotes sparsity in solutions while allowing for flexibility in penalizing different variables based on their importance or relevance. This makes it a powerful tool in applications such as feature selection, compressed sensing, and robust regression.

Specifically, we consider the setting $\mathcal{X} = \mathcal{Y} = \mathbb{R}^p$. Then, the weighted ℓ_1 regularizer $G_x \colon \mathbb{R}^p \to \mathbb{R}_+$ is defined as follows:

$$G_x(y) = \sum_{i=1}^{p} \psi(x_i)|y_i|, \tag{12}$$

where $x \in \mathbb{R}^p$ is the weight vector and $\psi \colon \mathbb{R} \to \mathbb{R}_+$ is a weight function, which is assumed to be continuously differentiable. It is straightforward that the function G_x defined in (12) satisfies Assumption 3.

To simplify the notation, let us set the vector weight mapping $\Psi \colon \mathbb{R}^p \to \mathbb{R}^p_+$ by $\Psi(x)_i := \psi(x_i)$. In this setting, the Forward-Backward (FB) operator and its associated residual–originally introduced in (6) and (7)–take on a concrete form tailored to our regularizer:

$$T^\gamma_x(y) = T^\gamma(x, y) = (\mathcal{T} \circ \mathcal{H}^\gamma)(x, y), \tag{13}$$

$$R^\gamma_x(y) = R^\gamma(x, y) = \gamma^{-1}(y - T^\gamma(x, y)), \tag{14}$$

where the mapping $\mathcal{H}^\gamma \colon \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}^p_+ \times \mathbb{R}^p$ is defined by $\mathcal{H}^\gamma(x, y) = (\gamma\Psi(x), y - \gamma\nabla F(y))$, and $\mathcal{T} \colon \mathbb{R}^p_+ \times \mathbb{R}^p \to \mathbb{R}^p$ corresponds to the component-wise soft-thresholding operator, defined for $(u, v) \in \mathbb{R}^p_+ \times \mathbb{R}^p$ as follows:

$$\mathcal{T}(u, v) = (\tau(u_i, v_i))_{i=1}^p = (\max\{|v_i| - u_i, 0\} \operatorname{sign}(v_i))_{i=1}^p . \tag{15}$$

Here, we have adapted the notation to explicitly reflect the dependence on both the parameter x and y, which will be crucial for analyzing their sensitivity in the remainder of this section.

The expressions for the solution mapping in (8), as well as for the Forward-Backward operator and residual in (13) and (14), respectively, allow for a precise characterization of the solution mapping S associated with the weighted ℓ_1 regularizer:

$$S(x) = \{y \in \mathbb{R}^p \mid y - (\mathcal{T} \circ \mathcal{H}^\gamma)(x, y) = 0\}.$$

This formulation is the starting point for computing the limiting subdifferential of the corresponding bilevel problem. As shown in Lemma 5, such analysis relies on the coderivative of the residual operator R^γ, which, in the case of the weighted ℓ_1 regularizer, further reduces to the coderivative of the component-wise soft-thresholding operator $\mathcal{T}$.

3.3.1 Coderivative calculus for the component-wise soft-thresholding operator $\mathcal{T}$

In this section, we derive more precise representations of the tangent and normal cones to the graph of the operator $\mathcal{T}$ introduced in (15). To that end, we introduce the sets of positive/negative inactive, active, and positive/negative biactive indices, respectively,

defined by

$$\begin{aligned}\mathcal{I}^+(u,v) &= \{i \in \{1,\dots,p\} \mid v_i > u_i\},\ \mathcal{I}^-(u,v) = \{i \in \{1,\dots,p\} \mid v_i < -u_i\},\\ \mathcal{B}^+(u,v) &= \{i \in \{1,\dots,p\} \mid v_i = u_i\},\ \mathcal{B}^-(u,v) = \{i \in \{1,\dots,p\} \mid v_i = -u_i\},\\ \mathcal{A}(u,v) &= \{i \in \{1,\dots,p\} \mid |v_i| < u_i\}.\end{aligned} \tag{16}$$

Using this partition of the index set $\{1,\dots,p\}$, we first obtain the following representation of the graph of $\mathcal{T}$ and the closedness of the graph of $\mathcal{T}$, which is presented formally in the following proposition.

Proposition 6 *Considering the operator $\mathcal{T}$, as defined in (15). Then, a vector $(u,v,w) \in \mathbb{R}^p_+ \times \mathbb{R}^p \times \mathbb{R}^p$ belongs to $\operatorname{gph}\mathcal{T}$ if and only if*

$$w_i = \tau(u_i,v_i) = \begin{cases} v_i - u_i & \text{if } i \in \mathcal{I}^+(u,v),\\ 0 & \text{if } i \in \mathcal{A}(u,v) \cup \mathcal{B}^+(u,v) \cup \mathcal{B}^-(u,v),\\ v_i + u_i & \text{if } i \in \mathcal{I}^-(u,v).\end{cases} \tag{17}$$

Furthermore, $\operatorname{gph}\mathcal{T}$ is a closed set.

Proof Let $\Omega := \mathbb{R}^p_+ \times \mathbb{R}^p \times \mathbb{R}^p$. First, a simple computation of the graph of $\mathcal{T}$ yields

$$\operatorname{gph}\mathcal{T} = \{(u,v,w) \in \Omega \mid w = \mathcal{T}(u,v)\} = \left\{ (u,v,w) \in \Omega \;\middle|\; w_i = \begin{cases} v_i - u_i & \text{if } v_i > u_i,\\ 0 & \text{if } |v_i| \le u_i,\\ v_i + u_i & \text{if } v_i < -u_i \end{cases} \right\}$$

Second, using the index sets defined in (16), we get that (17) holds. Figure 1 shows a graphical representation of $\operatorname{gph}\mathcal{T}$, clearly showcasing the piecewise linear structure of the operator $\mathcal{T}$ and its distinct behavior across the different index sets.

Now, let us focus on the closedness of $\operatorname{gph}\mathcal{T}$. To prove this, let us examine the soft-thresholding operator as a piecewise linear function. Specifically, the operator is defined as follows:

$$\tau(u_i,v_i) = \begin{cases} v_i - u_i & \text{if } v_i > u_i,\\ 0 & \text{if } |v_i| \le u_i,\\ v_i + u_i & \text{if } v_i < -u_i.\end{cases}$$

Now, consider a sequence $(u_i^k, v_i^k, w_i^k) \to (u_i, v_i, w_i)$. Using the continuity of τ we observe that $w_i^k = \tau(u_i^k, v_i^k) \to \tau(u_i, v_i)$, which implies $w_i^k \to w_i$. Since each component of $\mathcal{T}$ satisfies this property, the result follows directly. □

Using Proposition 6, we now focus on characterizing the limiting normal cone to $\operatorname{gph}\mathcal{T}$. To this end, we follow the constructive route suggested by the definitions: we begin by deriving an explicit expression for the tangent cone to the graph of the component-wise soft-thresholding operator $\mathcal{T}$. Next, we obtain the Fréchet normal cone by applying the polar operation to the tangent cone. Finally, passing to the limit, we compute the Mordukhovich normal cone.

We start by providing an explicit characterization of the tangent cone to the graph of the mapping $\mathcal{T}$, as defined in (15). The complete proof is given in Appendix A.

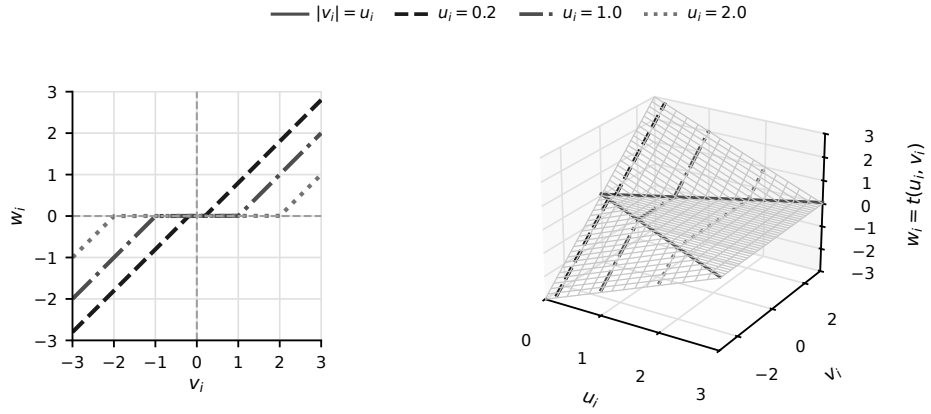

Fig. 1 Graphical description of $\operatorname{gph} \mathcal{T}$
The left plot shows the behavior of the soft-thresholding operator $\mathcal{T}$ for a single component i, demonstrating its piecewise linear structure. The right plot extends this to a 3D view, highlighting interactions between components, with red lines marking the biactive sets $\mathcal{B}^+(u,v) \cup \mathcal{B}^-(u,v)$.

Proposition 7 *The tangent cone for* $\operatorname{gph} \mathcal{T}$ *at* $(u,v,w) \in \operatorname{gph} \mathcal{T}$ *is given by:*

$$T((u,v,w), \operatorname{gph} \mathcal{T}) = \left\{ (\delta^u, \delta^v, \delta^w) \in \mathbb{R}^{3p} \;\middle|\; \begin{cases} \delta_i^w = \delta_i^v - \delta_i^u & \text{if } i \in \mathcal{I}^+(u,v), \\ \delta_i^w = \delta_i^v + \delta_i^u & \text{if } i \in \mathcal{I}^-(u,v), \\ \delta_i^w = 0 & \text{if } i \in \mathcal{A}(u,v), \\ \delta_i^w = \max\{\delta_i^v - \delta_i^u, 0\} & \text{if } i \in \mathcal{B}^+(u,v),\ u_i > 0, \\ \delta_i^w = \min\{\delta_i^v + \delta_i^u, 0\} & \text{if } i \in \mathcal{B}^-(u,v),\ u_i > 0 \end{cases} \right\} \tag{18}$$

Remark 2 The qualifier $u_i > 0$ in the biactive cases of (18) excludes the vertex configuration $u_i = v_i = 0$, at which i belongs simultaneously to $\mathcal{B}^+$ and $\mathcal{B}^-$. The tangent cone for that configuration is derived in Remark 5 of Appendix A. In the algorithmic context of this paper, $u_i = \gamma \Psi(x)_i > 0$ for all i by Assumption 3 and $\gamma > 0$, so the vertex case does not arise.

Using the characterization for the tangent cone, we can now provide a characterization for the limiting normal cone of the graph of the component-wise soft-thresholding operator. The proof can be found in Appendix B.

Proposition 8 *Let* $(u,v,w) \in \operatorname{gph} \mathcal{T}$*, then the limiting normal cone of the graph of the component-wise soft-thresholding operator at* $(u,v,w) \in \mathbb{R}^{3p}$ *is given by:*

$$N((u,v,w); \operatorname{gph} \mathcal{T}) =$$

$$\left\{ (\nu,\omega,-\zeta)\in\mathbb{R}^{3p} \;\middle|\; \begin{cases} \nu_i=-\zeta_i,\ \ \omega_i=\zeta_i & \text{if } i\in\mathcal{I}^+(u,v),\\ \nu_i=\ \ \zeta_i,\ \ \omega_i=\zeta_i & \text{if } i\in\mathcal{I}^-(u,v),\\ \nu_i=0,\ \ \omega_i=0 & \text{if } i\in\mathcal{A}(u,v),\\ \begin{cases}\nu_i=0,\ \ \omega_i=0,\\ \nu_i=-\omega_i,\ 0\le\omega_i\le\zeta_i,\\ \nu_i=-\zeta_i,\ \ \omega_i=\zeta_i\end{cases} & \text{if } i\in\mathcal{B}^+(u,v),\\ \begin{cases}\nu_i=0,\ \ \omega_i=0,\\ \nu_i=\ \ \omega_i,\ \zeta_i\le\omega_i\le 0,\\ \nu_i=\ \ \zeta_i,\ \ \omega_i=\zeta_i\end{cases} & \text{if } i\in\mathcal{B}^-(u,v)\end{cases}\right\}$$

Using Proposition 8, we obtain the coderivative of $\mathcal{T}$.

Corollary 9 *Let $(u,v)\in\mathbb{R}^p_+\times\mathbb{R}^p$. Then, for $z^*\in\mathbb{R}^p$, the coderivative of the component-wise soft-thresholding operator is given by:*

$$D^*\mathcal{T}(u,v)(z^*)=$$
$$\left\{ (u^*,v^*)\in\mathbb{R}^p_+\times\mathbb{R}^p \;\middle|\; \begin{cases} u_i^*=-z_i^*,\ v_i^*=z_i^*, & \text{if } i\in\mathcal{I}^+(u,v),\\ u_i^*=z_i^*,\ v_i^*=z_i^*, & \text{if } i\in\mathcal{I}^-(u,v),\\ u_i^*=v_i^*=0, & \text{if } i\in\mathcal{A}(u,v),\\ \left.\begin{array}{l} u_i^*=v_i^*=0\ \vee\\ u_i^*=-v_i^*,\ v_i^*\in[0,z_i^*]\ \vee\\ u_i^*=-z_i^*,\ v_i^*=z_i^*\end{array}\right\} & \text{if } i\in\mathcal{B}^+(u,v),\\ \left.\begin{array}{l} u_i^*=v_i^*=0\ \vee\\ u_i^*=v_i^*,\ v_i^*\in[z_i^*,0]\ \vee\\ u_i^*=z_i^*,\ v_i^*=z_i^*\end{array}\right\} & \text{if } i\in\mathcal{B}^-(u,v).\end{cases}\right\}. \tag{19}$$

Proof To start the proof, let us first consider the coderivative of $\mathcal{T}$:

$$D^*\mathcal{T}(u,v)(z^*):=\{(u^*,v^*):(u^*,v^*,-z^*)\in N((u,v,w),\operatorname{gph}\mathcal{T})\},$$

where $\operatorname{gph}\mathcal{T}:=\{(u,v,w):w=\mathcal{T}(u,v)\}$ and $N((u,v,w),\operatorname{gph}\mathcal{T})$ is the limiting normal cone to the $\operatorname{gph}\mathcal{T}$ at (u,v,w) described in Proposition 8. Then, using the provided characterization of the cone with $(\nu,\omega,-\zeta)=(u^*,v^*,-z^*)$, the result follows. □

3.3.2 Coderivative calculus of the FB residual R^γ and the solution mapping S

Next, we address the characterization of the subdifferential for the bilevel problem formulated as a generalized composition, as described in (9) with the weighted ℓ_1 regularizer in lower-level. With this goal in mind, we first provide a characterization for the residual operator coderivative in the following lemma.

Lemma 10 *Let $(x, y) \in \mathbb{R}^p \times \mathbb{R}^p$ and let Assumptions 2 and 1 hold. For $\gamma \in (0, \Lambda_F^{-1})$. Then, the coderivative of the residual operator R^γ defined for the weighted Lasso regularizer in (14) is:*

$$D^* R^\gamma(x, y)(z^*) = \left\{ \begin{pmatrix} \gamma J\Psi(x)u^* \\ z^* + (I - \gamma\nabla^2 F(y))v^* \end{pmatrix} \;\middle|\; (u^*, v^*) \in D^*\mathcal{T}(\mathcal{H}^\gamma(x, y))(-z^*) \right\}. \quad (20)$$

Proof Using the definition for the residual operator R^γ in (14), we can write the coderivative of the residual operator as follows:

$$D^* R^\gamma(x, y)(z^*) = D^*\left(I_y - \mathcal{T} \circ \mathcal{H}^\gamma\right)(x, y)(z^*) = (0, z^*) + D^*(\mathcal{T} \circ \mathcal{H}^\gamma)(x, y)(-z^*),$$

where we defined $I_y(x, y) = y$, and in the last equality we have Lemma 1 items (i) and (iii). Under the assumption that $\gamma \in (0, \Lambda_F^{-1})$, we can guarantee the matrix $I - \gamma\nabla^2 F(y)$ is strictly positive definite. Furthermore, the Jacobian of the vector-valued regularization mapping Ψ (cf. (12)) is a diagonal matrix. Specifically, given the element-wise structure $\Psi(x) = (\psi(x_1), \ldots, \psi(x_p))^\top$, the Jacobian $J\Psi(x) \in \mathbb{R}^{p\times p}$ is strictly diagonal, with its non-zero entries given by $[J\Psi(x)]_{ii} = \psi'(x_i)$ for all $i \in \{1, \ldots, p\}$. As a result, the coderivative of the composition is well-defined, and using item (ii) in Lemma 1, we obtain:

$$\begin{aligned} D^* R^\gamma(x, y)(z^*) &= (0, z^*) + J\mathcal{H}^\gamma(x, y)^\top \circ D^*\mathcal{T}(\mathcal{H}^\gamma(x, y))(-z^*) \\ &= (0, z^*) + \begin{pmatrix} \gamma J\Psi(x) & 0 \\ 0 & I - \gamma\nabla^2 F(y) \end{pmatrix}^\top \circ D^*\mathcal{T}(\mathcal{H}^\gamma(x, y))(-z^*). \end{aligned}$$

Now, taking $(u^*, v^*) \in D^*\mathcal{T}(\mathcal{H}^\gamma(x, y))(-z^*)$, we can write the coderivative as:

$$D^* R^\gamma(x, y)(z^*) = \{(x^*, y^*) |\; x^* = \gamma J\Psi(x)u^*, y^* = z^* + (I - \gamma\nabla^2 F(y))v^*\},$$

finishing the proof. □

A critical requirement for applying Lemma 5 to this problem is the satisfaction of the qualification condition (11) for the coderivative of the residual operator R^γ.

Lemma 11 *Let $(x, y) \in \mathbb{R}^p \times \mathbb{R}^p$ and the same assumptions required for Lemma 10. Then, the kernel of the coderivative of the residual operator R^γ defined for the weighted Lasso regularizer in (14) is given by:*

$$\ker D^* R^\gamma(x, y) = \{0\}. \quad (21)$$

Proof To prove this result, let us first consider that condition (21) is equivalent to:

$$(0, 0) \in D^* R^\gamma(x, y)(z^*) \implies z^* = 0.$$

Using the explicit geometric characterization from eq. (20) in Lemma 10, the condition $(0, 0) \in D^* R^\gamma(x, y)(z^*)$ implies there must exist a dual pair $(u^*, v^*) \in D^*\mathcal{T}(\mathcal{H}^\gamma(x, y))(-z^*)$ satisfying the system:

$$J\Psi(x)u^* = 0 \quad \wedge \quad z^* + (I - \gamma\nabla^2 F(y))v^* = 0.$$

By Assumption 3, $J\Psi(x)$ is a diagonal matrix with strictly positive entries, and therefore the first condition directly enforces $u^* = 0$. Subsequently, applying eq. (19) in Corollary 9 with $u^* = 0$, the structure of the soft-thresholding coderivative guarantees that $v^* = 0$ across all coordinate indices, regardless of their activity status (i.e., for both $i \in \mathcal{A}(\mathcal{H}^\gamma(x, y))$ and $i \notin \mathcal{A}(\mathcal{H}^\gamma(x, y))$). Substituting $v^* = 0$ into the second equality of our system yields:

$$z^* + (I - \gamma\nabla^2 F(y))0 = 0 \implies z^* = 0.$$

Hence $z^* = 0$. We conclude that $\ker D^* R^\gamma(x, y) = \{0\}$, which completes the proof. □

On a computable selection from the residual-enlarged subdifferential

As a direct consequence of Lemma 11, we can employ Lemma 5 to obtain a structural characterization of the residual-enlarged coderivative $D^*_{R^\gamma} S$ (Definition 1) associated with the weighted ℓ_1 regularizer. In particular, we define a computable selection from the residual-enlarged subdifferential. By introducing explicit selection masks to handle the non-uniqueness at biactive points, we arrive at the following result which covers both smooth and nonsmooth regimes.

Lemma 12 (Adjoint System for the Residual-Enlarged Coderivative) *Let the assumptions of Lemma 4 hold, $\bar{y} = S(\bar{x})$, and $z^* := \nabla_y L(\bar{x}, \bar{y})$. Since $\ker D^* R^\gamma(\bar{x}, \bar{y}) = \{0\}$ by Lemma 11, an element $x^* \in D^*_{R^\gamma} S(\bar{x})(z^*)$ is characterized by the existence of $q \in \mathbb{R}^p$ and a dual pair $(u^*, v^*) \in D^* \mathcal{T}(\mathcal{H}^\gamma(\bar{x}, \bar{y}))(-q)$ satisfying*

$$-z^* = q + (I - \gamma \nabla^2 F(\bar{y}))\, v^*, \tag{22a}$$

$$x^* = \gamma\, J\Psi(\bar{x})\, u^*. \tag{22b}$$

Proof To derive the adjoint component $x^* \in D^*_{R^\gamma} S(\bar{x})(z^*)$, we rely on the coderivative calculus for implicit mappings (see, e.g., [35, Corollary 4.34(ii)]). Because the qualification condition holds by Lemma 11, the cited result guarantees the existence of an adjoint vector q. Note that while q may not be strictly unique due to the set-valued nature of the nonsmooth mapping at the biactive coordinates, its existence is unconditionally guaranteed.

We now evaluate the inclusion $(x^*, -z^*) \in D^* R^\gamma(\bar{x}, \bar{y})(q)$ explicitly. Substituting the geometric structure of the residual coderivative from eq. (20) in Lemma 10 and matching the x- and y- components of the coderivative output yields the decoupled inclusion system:

$$x^* = \gamma J\Psi(\bar{x}) u^* \quad \text{and} \quad -z^* = q + (I - \gamma \nabla^2 F(\bar{y})) v^*,$$

where the dual multipliers must satisfy $(u^*, v^*) \in D^* \mathcal{T}(\mathcal{H}^\gamma(\bar{x}, \bar{y}))(-q)$. This corresponds exactly to the system defined in (22), completing the proof. □

Corollary 13 (Computable Element of the Residual-Enlarged Subdifferential) *Under the premises of Lemma 12, let $(\bar{u}, \bar{v}) := \mathcal{H}^\gamma(\bar{x}, \bar{y})$ and let $D_{supp} \in \mathbb{R}^{p \times p}$ be any diagonal signed support selection matrix with entries $(D_{supp})_{ii} \in \{-1, 0, +1\}$ satisfying:*

$$(D_{supp})_{ii} = \begin{cases} +1 & i \in \mathcal{I}^+(\bar{u}, \bar{v}), \\ -1 & i \in \mathcal{I}^-(\bar{u}, \bar{v}), \\ 0 & i \in \mathcal{A}(\bar{u}, \bar{v}), \\ +1 \text{ or } 0 & i \in \mathcal{B}^+(\bar{u}, \bar{v}), \\ -1 \text{ or } 0 & i \in \mathcal{B}^-(\bar{u}, \bar{v}). \end{cases}$$

Then, the reduced adjoint system

$$\Big[I - D^2_{supp} + \gamma \nabla^2_{yy} F(\bar{y})\, D^2_{supp}\Big] q = -\nabla_y L(\bar{x}, \bar{y}) \tag{23}$$

has a unique solution $q \in \mathbb{R}^p$, and the vector

$$h = \nabla_x L(\bar{x}, \bar{y}) + \gamma\, J\Psi(\bar{x})\, D_{supp}\, q \tag{24}$$

is an element of $\partial^\gamma_R \Phi(\bar{x})$.

Proof Set $z^* := \nabla_y L(\bar{x}, \bar{y})$ and let $\mathcal{S} := \{i : (D_{\text{supp}})_{ii} \neq 0\}$ denote the active support of D_{supp}. Since D_{supp} has entries in $\{-1, 0, +1\}$, its square D^2_{supp} is the diagonal $\{0,1\}$-projection onto $\mathcal{S}$.

Step 1 (Explicit selection). We construct a specific element of $D^*\mathcal{T}(\bar{u}, \bar{v})(-q)$ via Corollary 9. For each index i we choose:

- If $(D_{\text{supp}})_{ii} \neq 0$ (inactive or selected biactive index): apply option (iii) for $\mathcal{B}^+/\mathcal{I}^+$ and option (iii) for $\mathcal{B}^-/\mathcal{I}^-$. This gives, in every case,
$$u_i^* = (D_{\text{supp}})_{ii}\, q_i, \qquad v_i^* = -q_i.$$
Concretely: for $i \in \mathcal{I}^+$ (or $\mathcal{B}^+$ with $+1$ selected), Corollary 9 gives $u_i^* = q_i$ and $v_i^* = -q_i$; for $i \in \mathcal{I}^-$ (or $\mathcal{B}^-$ with -1 selected), it gives $u_i^* = -q_i$ and $v_i^* = -q_i$. Both are consistent with $u_i^* = (D_{\text{supp}})_{ii} q_i$ and $v_i^* = -(D^2_{\text{supp}})_{ii} q_i$.
- If $(D_{\text{supp}})_{ii} = 0$ (active or biactive with 0 selected): apply option (i), giving $u_i^* = v_i^* = 0$.

In compact form, the selection satisfies:
$$u^* = D_{\text{supp}}\, q, \qquad v^* = -D^2_{\text{supp}}\, q. \tag{25}$$

Step 2 (Deriving the linear system). Substituting $v^* = -D^2_{\text{supp}} q$ into the adjoint equation (22a) of Lemma 12:
$$-z^* = q + (I - \gamma \nabla^2_{yy} F(\bar{y}))\big(-D^2_{\text{supp}}\, q\big) = \big[I - D^2_{\text{supp}} + \gamma \nabla^2_{yy} F(\bar{y})\, D^2_{\text{supp}}\big] q,$$
which is exactly (23).

Step 3 (Unique solvability). Let $M := I - D^2_{\text{supp}} + \gamma \nabla^2_{yy} F(\bar{y})\, D^2_{\text{supp}}$. The block structure of M with respect to $\mathcal{S}$ and $\mathcal{S}^c$ is:
$$[Mr]_i = \begin{cases} r_i + \gamma \displaystyle\sum_{j \in \mathcal{S}} [\nabla^2_{yy} F(\bar{y})]_{ij}\, r_j & i \in \mathcal{S}^c, \\ \gamma \displaystyle\sum_{j \in \mathcal{S}} [\nabla^2_{yy} F(\bar{y})]_{ij}\, r_j & i \in \mathcal{S}. \end{cases}$$
Suppose $Mr = 0$. The equations for $i \in \mathcal{S}$ read $\gamma \nabla^2_{yy} F(\bar{y})\big|_{\mathcal{S}\times\mathcal{S}}\, r_{\mathcal{S}} = 0$. Since $\nabla^2_{yy} F(\bar{y})$ is positive definite by Assumption 2, every principal submatrix is positive definite, so $r_{\mathcal{S}} = 0$. Substituting $r_{\mathcal{S}} = 0$ into the equations for $i \in \mathcal{S}^c$ gives $r_i + \gamma \sum_{j \in \mathcal{S}} [\nabla^2_{yy} F(\bar{y})]_{ij} \cdot 0 = r_i = 0$. Hence $r = 0$, and M is invertible.

Step 4 (Membership in the residual-enlarged subdifferential). With $u^* = D_{\text{supp}} q$ as in (25), the equation (22b) of Lemma 12 yields $x^* = \gamma J\Psi(\bar{x})\, D_{\text{supp}}\, q$. By Lemma 12, $x^* \in D^*_{R^\gamma} S(\bar{x})(z^*)$, and by Definition 1,
$$h = \nabla_x L(\bar{x}, \bar{y}) + x^* \in \nabla_x L(\bar{x}, \bar{y}) + D^*_{R^\gamma} S(\bar{x})(z^*) = \partial^\gamma_R \Phi(\bar{x}).$$
This completes the proof. □

Corollary 13 characterizes admissible choices of the diagonal selector D_{supp} on biactive coordinates, but does not prescribe which one to use. The element of $\partial^\gamma_R \Phi(\bar{x})$ takes the form (24) $h = \nabla_x L(\bar{x}, \bar{y}) + \gamma\, J\Psi(\bar{x})\, D_{\text{supp}}\, q$, where q solves the reduced adjoint system $H_{\mathcal{S}}\, q_{\mathcal{S}} = -[z^\star]_{\mathcal{S}}$. For strict active coordinates the sign $\sigma_i := (D_{\text{supp}})_{ii} = \text{sign}(\bar{y}_i)$ is fixed by the lower-level solution; for biactive coordinates $\sigma_i \in \{-1, +1\}$ is a free design choice. Since $J\Psi(\bar{x})$ has strictly positive diagonal entries, the contribution of biactive coordinate $i \in \mathcal{S} \cap \mathcal{B}$ to h is proportional to $\sigma_i\, q_i$: the adjoint magnitude is determined by the linear system, and only the sign is under our control. A natural requirement is therefore $\sigma_i\, q_i > 0$ for every selected biactive coordinate — the adjoint corroborates the sign it was assigned. We call this the *sign-consistency condition*.

The following procedure constructs a selection satisfying it by initializing from a seed derived from z^* and iteratively pruning any biactive coordinate whose sign the adjoint contradicts.

Definition 2 (Sign-consistent biactive selection (`SC`)) Let $(\bar{x}, \bar{y})$ satisfy $\bar{y} = S(\bar{x})$, let $\mathcal{I}^+, \mathcal{I}^-, \mathcal{B}^+, \mathcal{B}^-$ be the coordinate partition, and set $z^* := \nabla_y L(\bar{x}, \bar{y})$.

(i) **Initialization.** Set $\sigma_i^{(0)} := +1$ for $i \in \mathcal{I}^+ \cup \{i \in \mathcal{B}^+ : z_i^* < 0\}$, $\sigma_i^{(0)} := -1$ for $i \in \mathcal{I}^- \cup \{i \in \mathcal{B}^- : z_i^* > 0\}$, and $\sigma_i^{(0)} := 0$ otherwise. Let $\mathcal{S}^{(0)} := \{i : \sigma_i^{(0)} \neq 0\}$.

(ii) **Pruning.** For $t = 0, 1, \ldots$, solve the reduced adjoint system

$$H_{\mathcal{S}^{(t)}}\, q^{(t)} = -[z^*]_{\mathcal{S}^{(t)}}, \qquad H_{\mathcal{S}} := \gamma\, [\nabla^2_{yy} F(\bar{y})]_{\mathcal{S},\mathcal{S}},$$

and remove from $\mathcal{S}^{(t)}$ every biactive index $i \in \mathcal{S}^{(t)} \cap \mathcal{B}$ satisfying $\sigma_i^{(t)}\, q_i^{(t)} \leq 0$.

(iii) **Termination.** Stop when no removal occurs; output $\mathcal{S} := \mathcal{S}^{(t)}$ and $(D_{\mathrm{supp}})_{ii} := \sigma_i^{(t)}$.

The procedure terminates in at most $|\mathcal{B}|$ steps since only biactive coordinates are removed. At termination, $\sigma_i\, q_i > 0$ for every $i \in \mathcal{S} \cap \mathcal{B}$, so the sign-consistency condition holds. When $\nabla^2_{yy} F(\bar{y})$ is diagonal, the initialization already satisfies sign-consistency and no pruning step is needed.

4 On the numerical solution of the NBHL problem

To solve the NBHL problem via any first-order scheme, the fundamental computational prerequisite is the evaluation of a residual-enlarged subgradient. The theoretical framework established in Corollary 13, specifically in eq. (24), dictates that extracting an element $h \in \partial_R^\gamma \Phi(x)$ requires solving a potentially dense $p \times p$ non-symmetric linear system. However, this naive computation does not scale to high-dimensional datasets.

To make the residual-enlarged subgradient practically computable, we use a support-reduced implementation of the coderivative formula. The key observation is that the generalized Jacobian structure vanishes outside the active and selected biactive coordinates, so the adjoint system can be restricted to a smaller working set. Indeed, in Algorithm 1, the key compression is in Phase 3: the adjoint solve shrinks from $\mathcal{O}(p^3)$ to $\mathcal{O}(|\mathcal{S}|^3)$, since $|\mathcal{S}| \ll p$ in any well-regularized sparse model. The only degree of freedom in the oracle is the biactive selection policy Π.

Remark 3 (Contrast with `Sparse-HO`) `Sparse-HO` [29] restricts the adjoint system to the primal support $\{i \mid \bar{y}_i \neq 0\}$, which coincides with our inactive index set $\mathcal{I}^+ \cup \mathcal{I}^-$. Under their non-degeneracy condition (Assumption 4 in [29]), the biactive sets $\mathcal{B}^+$ and $\mathcal{B}^-$ are empty, so the two adjoint systems are identical.

When $\mathcal{B}^+ \cup \mathcal{B}^- \neq \emptyset$, `Sparse-HO` assigns a zero subgradient component to every biactive coordinate because $\bar{y}_i = 0$ there, regardless of upper-level relevance. Our working set $\mathcal{S}$ is instead determined by the prox-argument pair $(\bar{u}, \bar{v}) = \mathcal{H}^\gamma(\bar{x}, \bar{y})$: biactive coordinates enter $\mathcal{S}$ whenever the selection masks $M_{\mathcal{B}^+}$ or $M_{\mathcal{B}^-}$ are non-trivial, allowing the adjoint system to capture non-zero generalized Jacobian contributions that a primal-support restriction would miss.

Algorithm 1 Support-Reduced Computation of a Residual-Enlarged Subgradient

Require: $\bar{x} \in \mathbb{R}^p$, step size $\gamma \in (0, \Lambda_F^{-1})$, biactive selection policy Π.

// Phase 1: Lower-Level Resolution & Support Identification

Compute a high-precision lower-level solution: $\bar{y} \approx S(\bar{x})$.

Set $(\bar{u}, \bar{v}) \leftarrow \big(\gamma\Psi(\bar{x}),\ \bar{y} - \gamma\nabla F(\bar{y})\big)$.

Partition $\{1, \dots, p\}$ based on $(\bar{u}, \bar{v})$ into $\mathcal{I}^+, \mathcal{I}^-, \mathcal{A}, \mathcal{B}^+, \mathcal{B}^-$.

// Phase 2: Selection Policy, Sign Mask & Subsystem Restriction

Compute upper-level partial gradients: $\nabla_x L(\bar{x}, \bar{y})$ and $z^* \leftarrow \nabla_y L(\bar{x}, \bar{y})$.

Apply policy Π to compute binary selection masks $M_{\mathcal{B}^+}, M_{\mathcal{B}^-} \in \{0,1\}^p$.

Define the active working set: $\mathcal{S} \leftarrow \mathcal{I}^+ \cup \mathcal{I}^- \cup \operatorname{supp}(M_{\mathcal{B}^+}) \cup \operatorname{supp}(M_{\mathcal{B}^-})$.

assert $\mathcal{S} \cap \mathcal{A} = \emptyset$ ▷ Ensure strict decoupling from active variables

Build the sign vector $\sigma \in \mathbb{R}^{|\mathcal{S}|}$:
$\sigma_i \leftarrow +1$ for $i \in \mathcal{I}^+ \cup \operatorname{supp}(M_{\mathcal{B}^+})$; $\sigma_i \leftarrow -1$ for $i \in \mathcal{I}^- \cup \operatorname{supp}(M_{\mathcal{B}^-})$.

// Phase 3: Reduced Adjoint System Solve

Extract the restricted dual vector: $[z^*]_{\mathcal{S}} \in \mathbb{R}^{|\mathcal{S}|}$.

Form the reduced Hessian block: $H_{\mathcal{S}} \leftarrow \gamma[\nabla^2 F(\bar{y})]_{\mathcal{S},\mathcal{S}}$.

Solve the reduced adjoint system: $H_{\mathcal{S}}\, q_{\mathcal{S}} = -[z^*]_{\mathcal{S}}$.

// Phase 4: Subgradient Reconstruction

Initialize $g_{\text{imp}} \leftarrow 0 \in \mathbb{R}^p$.

Apply sign correction and map onto the support: $[g_{\text{imp}}]_{\mathcal{S}} \leftarrow \gamma\,[J\Psi(\bar{x})]_{\mathcal{S},\mathcal{S}}\,(\sigma \odot q_{\mathcal{S}})$.

Compute the subgradient: $h \leftarrow \nabla_x L(\bar{x}, \bar{y}) + g_{\text{imp}}$.

return h.

With the residual-enlarged subgradient established in Algorithm 1, we now embed this routine within an outer-level optimization scheme. Algorithm 2 presents an adaptation of the bilevel approximation framework introduced by [5]. Adapted to our nonsmooth setting, this procedure utilizes a projected normalized subgradient method. Because the upper-level objective $\Phi(x)$ is generally non-convex and nonsmooth, we employ a fixed outer step size $\eta > 0$ and project the updates onto the feasible hyperparameter domain $\mathcal{X}$ to ensure model validity.

Alternatively, we propose a trust-region (TR) algorithm to govern the outer optimization loop. While traditional line-search techniques often stagnate in the presence of inexact oracles and non-smoothness, TR methods provide structural robustness by constructing a sequence of simplified local surrogate models. At each iteration, the hyperparameter update is strictly confined to a specified neighborhood—the trust region—where the surrogate is considered reliable. After computing the provisional step, the geometric fidelity of the model is evaluated by comparing the predicted reduction against the actual reduction in the true upper-level objective. Based on this agreement, the trust-region radius is dynamically expanded or contracted. This makes it well-suited for nonsmooth and potentially ill-conditioned upper-level objectives.

Algorithm 2 Nonsmooth Bilevel Approximation (`NBA`-$w\ell_1$)

Require: Initial hyperparameter $x_0 \in \mathcal{X} \subset \mathbb{R}^p$, tolerance tol > 0, $\gamma \in (0, \Lambda_F^{-1})$, outer step size $\eta > 0$, maximum iterations K.

for $k = 0, \dots, K-1$ **do**
  // Step 1: Residual-Enlarged Subdifferential Oracle
  Call Algorithm 1 with (x_k, γ) to obtain the subgradient $h_k \in \partial_R^\gamma \Phi(x_k)$.
  // Step 2: Stationarity Check
  if $\|h_k\| <$ tol **then**
    break
  end if
  // Step 3: Projected Normalized Subgradient Update
  $x_{k+1} \leftarrow P_{\mathcal{X}}\big(x_k - \eta\, h_k / \|h_k\|\big)$
end for
return x_k

Building upon the trust-region framework for nonsmooth optimization introduced by [37], Algorithm 3 presents the Nonsmooth Trust-Region Bilevel Approximation (`NTRBA`-$w\ell_1$) method. Designed for hyperparameter learning with a lower-level weighted ℓ_1 penalty, this approach embeds the residual-enlarged subdifferential oracle (Algorithm 1) within the TR subproblem. `NTRBA` uses a projected Cauchy step rather than a Hessian approximation, keeping the per-iteration cost low while enforcing the non-negativity of the regularization weights.

5 Numerical Experiments

The objective of this section is twofold. First, we demonstrate that feature-wise regularization provides strictly more expressive power than scalar regularization for structured sparse regression, but only if the vector hyperparameter can actually be optimized. Second, we show that the standard support-restricted adjoint (`Sparse-HO`; [29]) cannot optimize it on degenerate instances due to an exact, permanent gradient-starvation phenomenon at biactive coordinates—and that the self-consistent biactive policy introduced in Definition 2 resolves it.

The experiments are organized to make these two claims visible independently, in increasing order of realism and scale. *Experiment 1* (Section 5.1) establishes both motivations in the smallest possible illustrative instance: a two-panel proof showing (i) that the forward-backward reformulation (FB) preserves the solution set while classical smoothing techniques like Berkovier-Engelman (BE) do not, and (ii) that primal-support implicit differentiation assigns an identically zero subgradient to a validation-relevant biactive feature throughout the entire outer loop. *Experiment 2* (Section 5.2) asks whether feature-wise regularization is empirically worth the effort on synthetic overparameterized regression: we compare bilevel optimized per feature penalties against the best scalar elastic net tuned by grid search, and show that feature resolution rather than regularizer family is the binding constraint.

Experiment 3 (Section 5.3) isolates the two algorithmic contributions in a controlled 2×2 ablation: crossing the self-consistent (`SC`) and null biactive oracles against

Algorithm 3 Nonsmooth Trust-Region Bilevel Approximation (`NTRBA`-$w\ell_1$)

Require: Initial hyperparameter $x_0 \in \mathcal{X} \subset \mathbb{R}^p$
Require: Initial radius $\Delta_0 > 0$, radius bounds $0 < \Delta_{\min} \leq \Delta_{\max}$.
Require: Acceptance threshold $\eta_{\text{accept}} > 0$, expansion threshold $\eta_{\text{expand}} > \eta_{\text{accept}}$.
Require: Shrink factor $\beta_{\text{dec}} \in (0, 1)$, growth factor $\beta_{\text{inc}} > 1$.
Require: Tolerance tol > 0, maximum iterations K.

for $k = 0, \ldots, K-1$ **do**
  // Step 1: Residual-Enlarged Subdifferential Oracle
  Call Algorithm 1 with x_k to obtain $h_k \in \partial_R^\gamma \Phi(x_k)$.
  if $\|h_k\| <$ tol **then break** ▷ Heuristic termination near stationarity
  // Step 2: Projected Cauchy Step Formulation
  Compute the scaled steepest descent step: $d_k \leftarrow -\left(\Delta_k / \|h_k\|\right) h_k$
  Compute the effective step after projecting onto the feasible domain $\mathcal{X}$: $s_k \leftarrow P_{\mathcal{X}}(x_k + d_k) - x_k$
  if $\|s_k\| < \epsilon$ **then break** ▷ Safeguard: Trajectory pinned by active constraints
  // Step 3: Model Evaluation
  Compute the predicted linear reduction: $\text{pred}_k \leftarrow -\langle h_k, s_k \rangle$
  if $\text{pred}_k \leq 0$ **then goto** Step 4 (Reject) ▷ Safeguard: Non-descent direction
  Evaluate the actual outer objective reduction at the trial point: $\text{ared}_k \leftarrow \Phi(x_k) - \Phi(x_k + s_k)$
  Calculate the fidelity ratio: $\rho_k \leftarrow \text{ared}_k / \text{pred}_k$
  // Step 4: Step Acceptance and Radius Adaptation
  if $\rho_k < \eta_{\text{accept}}$ **then**
    $x_{k+1} \leftarrow x_k$ ▷ Reject the trial step
    $\Delta_{k+1} \leftarrow \max(\Delta_{\min}, \beta_{\text{dec}} \Delta_k)$ ▷ Contract the trust region
    if $\Delta_{k+1} = \Delta_{\min}$ **then break** ▷ Safeguard: Minimum radius reached
  else
    $x_{k+1} \leftarrow x_k + s_k$ ▷ Accept the trial step
    if $\rho_k > \eta_{\text{expand}}$ **and** $\|s_k\| \geq 0.95 \Delta_k$ **then**
      $\Delta_{k+1} \leftarrow \min(\Delta_{\max}, \beta_{\text{inc}} \Delta_k)$ ▷ Expand the trust region
    else
      $\Delta_{k+1} \leftarrow \Delta_k$ ▷ Maintain current radius
    end if
  end if
end for
return x_k

the normalized subgradient (`NBA`) and nonsmooth trust-region (`NTRBA`) outer optimizers on the same degenerate dataset. This distinguishes the effect of the oracle, which determines where the outer loop converges, from the effect of the optimizer, which determines how efficiently it reaches that point.

Experiment 4 (Section 5.4) is the critical test of the paper's central claim: it compares `NTRBA`-$w\ell_1$ against scalar ℓ_1 tuning and `Sparse-HO` on calibrated degenerate

instances, using the $\|\partial_{x_{\text{hid}}}\Phi\|_{k=0}$ column of Table 3 as a direct measurement of the gradient-starvation effect.

Experiment 5 (Section 5.5) studies high-dimensional binary classification benchmarks to assess whether the gradient-starvation phenomenon observed in the synthetic setting also appears in realistic correlated-feature regimes, and whether the self-consistent oracle remains effective there.

Implementation and reproducibility. Unless stated otherwise, all experiments were run with the same implementation principles. For bilevel runs, we store the full validation trajectory and report the model associated with the best validation objective visited during the run. The corresponding lower-level problem is then solved once more at that selected hyperparameter to obtain the final coefficient vector used for evaluation.

Lower-level solver. For all learning experiments, the inner problem is solved in the primal by proximal coordinate descent. In the regression experiments (Experiments 2–4) we use the weighted elastic-net model

$$\min_{y\in\mathbb{R}^p} \frac{1}{2n}\|b - Ay\|_2^2 + \frac{\alpha_{\ell_2}}{2}\|y\|_2^2 + \sum_{j=1}^{p} \exp(x_j)|y_j|,$$

whereas in Experiment 5 we use the weighted sparse logistic model

$$\min_{y\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n} \log\bigl(1 + \exp(-b_i a_i^\top y)\bigr) + \frac{\alpha_{\ell_2}}{2}\|y\|_2^2 + \sum_{j=1}^{p} \exp(x_j)|y_j|.$$

Each coordinate update is a soft-thresholding/proximal step with feature-wise Lipschitz scaling. For quadratic loss this yields the usual elastic-net coordinate update with precomputed $L_j = \|A_j\|_2^2/n$, while for logistic loss the update uses coordinate-wise curvature bounds of the smooth part. Dense matrices are handled in Fortran layout and sparse matrices in CSC format, so that the same solver can exploit efficient dense and sparse kernels. We stop the inner iterations using KKT-based residual checks with tolerance 10^{-8} rather than a fixed number of passes. Code and experiment scripts are available at https://github.com/dvillacis/sparse-ho-fb.

Remark 4 (The Role of the ℓ_2 Stabilizer) In our framework, the inner penalty $\alpha_{\ell_2} > 0$ is deliberately treated as a fixed Tikhonov stabilizer rather than a tunable hyperparameter. Analytically, it guarantees that the lower-level objective satisfies the strong convexity condition (Assumption 2) with modulus $\mu_F \geq \alpha_{\ell_2}$. This bounds the condition number of the reduced adjoint system, ensuring that the residual-enlarged subgradient remains well-posed and numerically stable even when the empirical design matrix is severely rank-deficient ($p \gg n$). Optimization is thus intentionally restricted to the non-smooth feature-wise weights x, which govern the critical sparse structure.

5.1 A minimal example of FB exactness and gradient starvation

This experiment is a toy example designed to expose two issues in the simplest possible setting: the exactness of the FB reformulation and the effect of the biactive oracle selection. Figure 2 consists of two complementary panels.

Panel (a): lower-level geometry. We consider the two-dimensional weighted ℓ_1 problem

$$y^\star(x) \in \arg\min_{y\in\mathbb{R}^2} \left\{ \frac{1}{2}\|y-d\|_2^2 + \sum_{j=1}^{2} \exp(x_j)|y_j| \right\},$$

with $d = (0.8,\, 0.5)$ and $\exp(x) = (0.6,\, 0.6)$. Since the problem is diagonal, the lower-level solution is available in closed form by componentwise soft-thresholding, i.e., $y^* = (0.2, 0)$. This panel highlights the geometric effect of smoothing at a nonsmooth minimizer. In this example, the true lower-level minimizer lies on the ℓ_1 boundary. The smoothing perturbs the landscape and shifts that minimizer into the interior, with a bias that increases with the smoothing parameter. By contrast, the FB reformulation preserves the exact minimizer for every $\gamma > 0$, so the induced outer objective retains the correct kink and minimizer.

Panel (b): outer-level trajectory. The second panel considers a three-dimensional diagonal instance with train target $d_{\text{train}} = (0.4,\, 1.2,\, 0)$ and validation target $d_{\text{val}} = (0,\, 1.2,\, 0)$. The bilevel problem is

$$\begin{aligned} \min_{x\,\in\,\mathbb{R}^3} \quad & L(y^\star(x)) := \frac{1}{2}\|y^\star(x) - d_{\text{val}}\|_2^2 \\ \text{s.t.} \quad & y^\star(x) \in \arg\min_{y\in\mathbb{R}^3} \left\{ \frac{1}{2}\|y-d_{\text{train}}\|_2^2 + \sum_{j=1}^{3} \exp(x_j)|y_j| \right\}. \end{aligned}$$

The diagonal structure again makes the lower-level solution explicit. To isolate the biactive mechanism, we update only the second hyperparameter.

$$x_2^{k+1} = x_2^k - \eta\, h_2(x^k), \qquad \eta = 1,$$

from the initialization $\exp(x^0) = (0.35,\, 1.20,\, 2.00)$, so that the second coordinate is biactive at the first outer iterate. We then compare the null and self-consistent policies over ten outer iterations. The panel shows that, under the null policy, the biactive coordinate receives an identically zero subgradient and the outer iteration stalls, whereas the self-consistent policy assigns a nonzero descent direction and escapes this gradient-starvation regime.

5.2 Feature-wise vs. scalar regularization

This experiment studies whether feature-wise regularization improves recovery in over-parameterized linear regression. We compare a scalar elastic-net baseline, where a

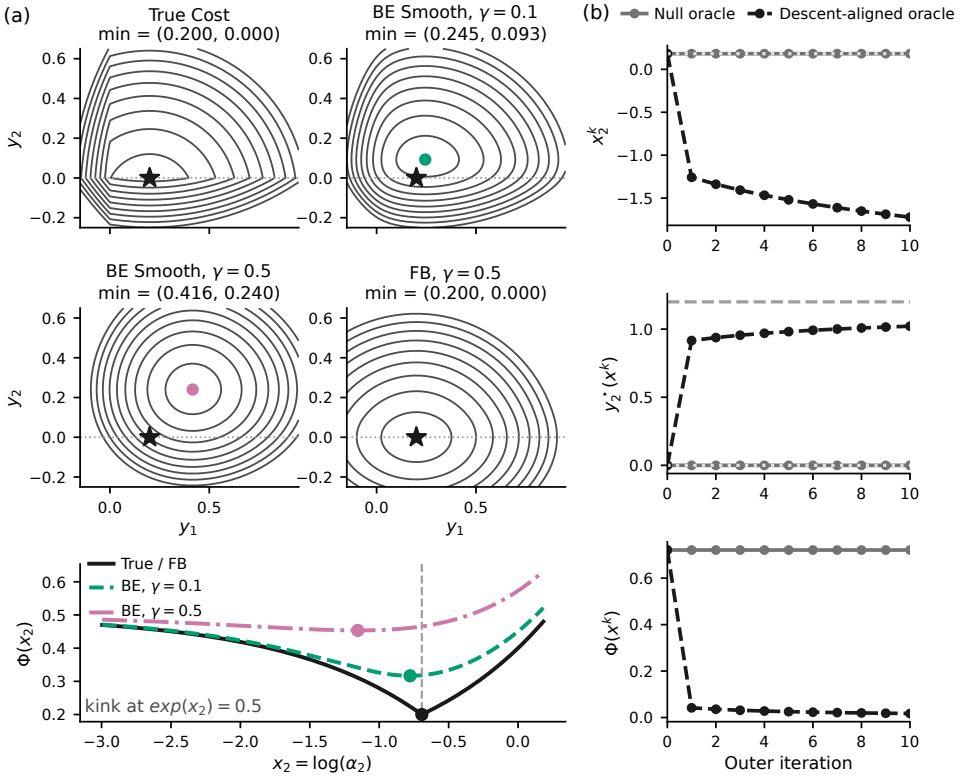

Fig. 2 **Left:** Berkovier–Engelman smoothing displaces the lower-level minimizer from the ℓ_1 boundary, whereas the forward-backward reformulation preserves it exactly; the same distortion is inherited by the outer objective. **Right:** At a biactive coordinate, the null oracle yields a zero subgradient and stalls the outer iteration, while the self-consistent oracle produces a nonzero descent signal and reduces the objective.

single penalty is selected by validation search, with our weighted elastic-net formulation, where each feature receives its own regularization parameter. The goal is to test whether the bilevel procedure can exploit heterogeneous feature structure, in particular in the presence of correlated nuisance variables.

We use synthetic three-group regression data with signal features, correlated noise features, and pure-noise features, and generate independent train/validation/test splits with proportions 60/20/20. The lower-level problem is the weighted elastic-net with feature-wise penalties and fixed ridge parameter $\alpha_{\ell_2} = 1/n_{\text{train}}$. For the scalar baseline, a single penalty is selected on a grid of 100 values using the validation MSE. For the weighted model, the upper-level objective is the validation MSE, the subgradient is computed using Algorithm 1, and the outer problem is optimized with `NTRBA-SC` for 50 outer iterations.

Figure 3 highlights a simple but important limitation of scalar regularization: when all features share the same penalty, the model cannot distinguish between informative

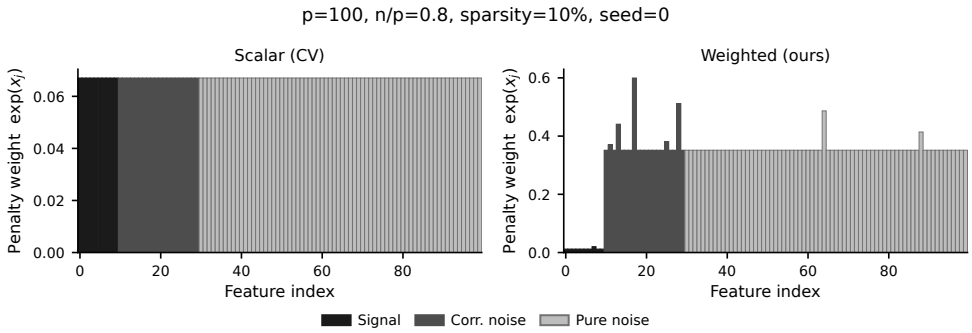

Fig. 3 Learned weights profile: comparison between scalar and feature-wise regularization on synthetic overparameterized regression. The weighted bilevel formulation yields clearer separation between signal and nuisance features, improving support recovery.

Table 1 Support-recovery F1 and test MSE (mean ± std over 10 seeds) for scalar elastic-net CV vs. weighted elastic-net (ours) across all synthetic configurations.

			Support Recovery F1 ↑		Test MSE ↓	
p	n/p	nnz	Scalar	Weighted (ours)	Scalar	Weighted (ours)
100	0.8	2%	0.282 ± 0.076	1.000 ± 0.000	0.020 ± 0.011	0.012 ± 0.006
		5%	0.408 ± 0.077	1.000 ± 0.000	0.108 ± 0.097	0.017 ± 0.006
		10%	0.464 ± 0.027	0.966 ± 0.054	0.488 ± 0.242	0.355 ± 0.649
	1	2%	0.291 ± 0.112	1.000 ± 0.000	0.020 ± 0.007	0.013 ± 0.004
		5%	0.377 ± 0.044	0.991 ± 0.027	0.053 ± 0.032	0.018 ± 0.006
		10%	0.452 ± 0.046	1.000 ± 0.000	0.133 ± 0.046	0.025 ± 0.007
200	0.8	2%	0.337 ± 0.108	1.000 ± 0.000	0.018 ± 0.005	0.012 ± 0.003
		5%	0.439 ± 0.048	1.000 ± 0.000	0.053 ± 0.021	0.019 ± 0.007
		10%	0.494 ± 0.038	0.942 ± 0.071	0.545 ± 0.325	0.652 ± 0.840
	1	2%	0.285 ± 0.072	1.000 ± 0.000	0.017 ± 0.003	0.013 ± 0.002
		5%	0.395 ± 0.045	1.000 ± 0.000	0.030 ± 0.011	0.014 ± 0.004
		10%	0.484 ± 0.033	0.986 ± 0.035	0.145 ± 0.068	0.181 ± 0.429

variables and correlated nuisance variables. In the overparameterized regimes considered here, this creates a structural compromise. A penalty small enough to preserve weak but relevant coordinates also tends to retain correlated noise, whereas a larger penalty may suppress nuisance features only at the cost of shrinking part of the true support.

By contrast, the weighted formulation resolves this tension by assigning each feature its own regularization level. The learned penalties discriminate between the three groups: signal variables receive smaller values, while correlated and pure-noise features are assigned stronger shrinkage, as illustrated in Figure 3.

As shown in Table 1, this finer resolution yields consistently higher F_1 support recovery across configurations and, at higher sparsity levels where the recovery problem is well-conditioned, also substantially reduces prediction error. At the lowest sparsity fraction tested (nnz=10%), the number of signal and correlated-noise features approaches the information-theoretic recovery threshold for the available training samples; in this near-threshold regime both methods produce high-variance test MSE estimates, and reliable prediction gains are not guaranteed even though the F_1 advantage is maintained. The support-recovery benefit is most pronounced when the true signal occupies a small fraction of the features, since it is precisely in these settings that a uniform penalty most severely conflates signal variables with correlated noise.

Calibrated degenerate design.

Experiments 3 and 4 use a calibrated four-group synthetic regression design whose purpose is to make gradient starvation visible at the initial hyperparameter. To avoid overloading the hyperparameter notation x, we denote the experimental design matrix by A. Its columns are partitioned as

$$A = [A_{\text{easy}}, A_{\text{dist}}, A_{\text{hid}}, A_{\text{noise}}],$$

where easy and hidden features have nonzero ground-truth coefficients, while distractor and noise features have zero ground-truth coefficients. The easy, distractor, and noise columns are sampled independently from a standard Gaussian distribution. Each hidden column is paired with a distractor column and generated as

$$A_{\text{hid},j} = \rho A_{\text{dist},\pi(j)} + \sqrt{1-\rho^2}\,\xi_j,$$

with ξ_j standard Gaussian and independent of the distractor columns. Thus ρ controls the correlation between hidden relevant variables and their nuisance distractors. The response is generated as

$$b = A\beta^\star + \sigma\varepsilon,$$

where $\beta^\star$ denotes the data-generating coefficient vector, and is supported only on the easy and hidden groups, with coefficients $\pm\beta_{\text{std}}$, and ε is standard Gaussian noise. In all synthetic degenerate experiments we use $\beta_{\text{std}} = 1$ and $\sigma = 0.05$.

The group sizes are chosen as follows. In Experiments 3 and 4, easy features and distractors each occupy 4% of the ambient dimension, with a minimum of five coordinates per group, and the number of hidden features equals the number of distractors. The remaining coordinates are background noise features. Thus

$$n_{\text{easy}} = \max\{\lfloor 0.04p \rfloor, 5\}, \qquad n_{\text{dist}} = \max\{\lfloor 0.04p \rfloor, 5\}, \qquad n_{\text{hid}} = n_{\text{dist}},$$

and $n_{\text{noise}} = p - n_{\text{easy}} - n_{\text{dist}} - n_{\text{hid}}$. We use a 60/20/20 train/validation/test split. If I_{tr}, I_{val}, and I_{test} denote the corresponding index sets, then

$$(A_{\text{tr}}, b_{\text{tr}}) = (A_{I_{\text{tr}}}, b_{I_{\text{tr}}}), \qquad (A_{\text{val}}, b_{\text{val}}) = (A_{I_{\text{val}}}, b_{I_{\text{val}}}), \qquad (A_{\text{test}}, b_{\text{test}}) = (A_{I_{\text{test}}}, b_{I_{\text{test}}}).$$

The lower-level weighted elastic-net is solved on $(A_{\mathrm{tr}}, b_{\mathrm{tr}})$ with fixed ridge parameter $\alpha_{\ell_2} = 1/n_{\mathrm{train}}$, while the upper-level objective is the validation MSE on $(A_{\mathrm{val}}, b_{\mathrm{val}})$.

The initial feature-wise penalties are calibrated in two passes. First, hidden features are assigned a large sentinel penalty and the lower-level problem is solved on $(A_{\mathrm{tr}}, b_{\mathrm{tr}})$, producing a coefficient vector β_S in which the hidden coordinates are inactive. The easy, distractor, and noise penalties are initialized as

$$\alpha_{\mathrm{easy}} = 0.5\,\beta_{\mathrm{std}}, \qquad \alpha_{\mathrm{dist}} = 0.5\,\frac{1-\rho^2}{\rho}\,\beta_{\mathrm{std}}, \qquad \alpha_{\mathrm{noise}} = 10\,\beta_{\mathrm{std}}.$$

Second, for each hidden coordinate j we compute the smooth training gradient at β_S. Since $(\beta_S)_j = 0$ on hidden coordinates, the ridge contribution vanishes there, and this gradient is

$$g_{\mathrm{hid},j} = \frac{1}{n_{\mathrm{train}}} A_{\mathrm{tr,hid},j}^{\top}\big(A_{\mathrm{tr}}\beta_S - b_{\mathrm{tr}}\big).$$

We then set the initial hidden penalty to

$$\alpha^0_{\mathrm{hid},j} = |g_{\mathrm{hid},j}|(1+\delta), \qquad \delta = 0.05.$$

Without the factor $(1+\delta)$ this places the hidden coordinate exactly at the weighted ℓ_1 threshold for the lower-level optimality condition. The small slack makes the hidden coordinates strictly inactive for the lower-level coordinate-descent solution while keeping them inside the relative biactive detection tolerance used by the variational oracle. Consequently, the standard support-restricted subgradient assigns zero signal to these hidden relevant coordinates at initialization, whereas the self-consistent oracle can select them as biactive coordinates and recover a nonzero descent direction.

5.3 Oracle–Optimizer Ablation

This experiment is a mechanism study, not a performance benchmark: its purpose is to show how gradient starvation arises and which components resolve it, under controlled conditions where the failure mode is visible by construction. Indeed, it is designed to isolate the two main ingredients of the proposed approach: the biactive selection rule used in the residual-enlarged subgradient computation, and the outer optimization scheme used to update the regularization parameters. To this end, we consider a controlled synthetic setting in which both effects can be separated cleanly. The experiment crosses two oracle choices, namely the null policy and the self-consistent policy, with two outer solvers, namely projected normalized subgradient descent and the trust-region method. This yields four variants and allows us to determine whether the gains come from the oracle, from the optimizer, or from their combination.

The data are generated so that a subset of relevant features is strictly inactive for the lower-level solver but lies within the biactive detection band at the initial hyperparameter. This makes the failure mode of the standard support-restricted oracle directly visible: under the null policy, these coordinates receive zero subgradient from the first iteration onward, even though they are relevant for the outer objective. The self-consistent policy is introduced precisely to recover useful descent information on such coordinates.

We use the calibrated degenerate synthetic generator introduced for this study, with four feature groups: easy signals, distractors, hidden relevant features, and background noise. In particular, $p \in \{250, 500, 1000\}$, $n = \lfloor (2/3)p \rfloor$, $\rho = 0.95$, and average over five seeds. The lower-level problem is the weighted elastic-net with feature-wise penalties and fixed ridge parameter $\alpha_{\ell_2} = 1/n_{\text{train}}$, and the upper-level objective is the validation MSE.

The four methods are obtained by combining two subgradient oracles and two outer solvers. The null oracle corresponds to the standard support-restricted rule, while the self-consistent oracle augments the working set with selected biactive coordinates. For the outer loop, we compare projected normalized subgradient updates with fixed step size and the proposed trust-region method with adaptive radius. All methods are run for the same outer-iteration budget, and the lower-level problems are solved by proximal coordinate descent with warm starts. We report validation loss, support-recovery scores, recovery of the hidden relevant features, and runtime per outer iteration.

The results in Figure 4 and Table 2 show that the main difficulty in this experiment is not merely slow convergence, but the loss of descent information caused by the null oracle. When the standard support-restricted rule is used, the hidden but relevant biactive coordinates receive zero subgradient at the initial point and therefore remain essentially invisible to the outer optimization. This effect is reflected in the poorer hidden-feature recovery and in the larger validation-loss gap observed for the null variants.

By contrast, the self-consistent oracle restores nonzero descent directions on selected biactive coordinates. As seen in Figure 4, this leads to a markedly improved optimization trajectory, with faster and more stable reduction of the validation objective. Table 2 shows that this improvement is not only qualitative: it translates into better support recovery and substantially higher recall on the hidden relevant features.

The comparison between outer solvers further indicates that the optimizer matters once informative subgradients are available. The trust-region scheme makes more effective use of the corrected oracle than fixed-step normalized subgradient updates, yielding the most favorable overall behavior in both the figure and the table. These results support the intended interpretation of this experiment: the self-consistent oracle addresses gradient starvation, and the trust-region method turns this additional information into more reliable outer-level progress.

5.4 Gradient Starvation: Comparison with Baselines

Experiment 4 compares the proposed method with two natural baselines on the calibrated degenerate instances introduced in Experiment 3, using the same calibrated degenerate synthetic generator with $(n, p) \in \{(100, 150), (200, 300), (500, 750)\}$, $\rho \in \{0.90, 0.95, 0.98\}$, and five seeds. The comparison is intended to separate two questions: does feature-wise regularization by itself provide a sufficient advantage over scalar tuning, and when biactive coordinates are present, is it necessary to modify both the subgradient oracle and the outer solver? The scalar baseline selects a single penalty by validation grid search. The weighted `Sparse-HO` baseline uses feature-wise penalties together with the standard support-restricted implicit subgradient and normalized subgradient outer updates. The proposed method replaces the oracle by the

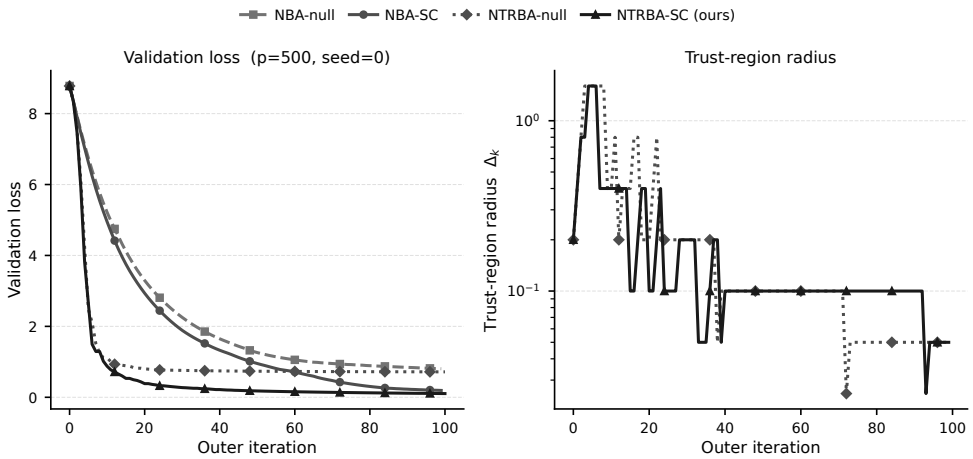

Fig. 4 Optimization trajectories for the oracle and optimizer ablation on the degenerate synthetic model: the self-consistent oracle improves the outer-level descent, and its combination with the trust-region method produces the most stable and effective reduction of the validation objective.

Table 2 Experiment 3 oracle×optimizer ablation (mean ± std over 5 seeds). Hidden features are biactive at initialization, so the null oracle starves them of subgradient signal while the self-consistent (`SC`) oracle restores that signal. The first two blocks show the gain from replacing the null oracle with `SC` under `NBA` and `NTRBA`; the third block shows the additional gain from replacing `NBA` with `NTRBA` once the oracle is fixed to `SC`. The final block anchors the comparison with the absolute outcome of `NTRBA-SC`. Better directions are: negative Δ loss, positive Δ recall, positive loss reduction, stationarity gain greater than 1, and positive recall gain.

Comparison	Metric	$p = 250$	$p = 500$	$p = 1000$
`SC` gain under `NBA`	Δ val. loss	−1.047 ± 1.048	−0.702 ± 0.126	−1.892 ± 0.680
	Δ recall	0.640 ± 0.136	0.420 ± 0.087	0.455 ± 0.080
`SC` gain under `NTRBA`	Δ val. loss	−1.031 ± 1.000	−0.817 ± 0.243	−1.413 ± 0.279
	Δ recall	0.660 ± 0.102	0.420 ± 0.163	0.440 ± 0.051
`NTRBA` gain under `SC`	loss red. (%)	68.0 ± 3.4	70.0 ± 18.8	86.9 ± 3.3
	stationarity gain (×)	4.2 ± 1.1	8.7 ± 8.7	7.0 ± 2.8
	recall gain	0.000 ± 0.000	0.000 ± 0.032	0.060 ± 0.044
`NTRBA-SC` outcome	best val. loss	0.016 ± 0.004	0.064 ± 0.034	0.130 ± 0.029
	F1	0.890 ± 0.028	0.889 ± 0.020	0.894 ± 0.018

self-consistent subgradient rule and updates the hyperparameters with the trust-region method. All bilevel methods are run with the same outer-iteration budget; the lower-level problems are solved by proximal coordinate descent with a weighted elastic-net objective and fixed ridge parameter $\alpha_{\ell_2} = 1/n_{\text{train}}$.

Table 3 shows that the proposed method achieves the best F_1 support-recovery score in every configuration. Validation loss and test MSE show a more nuanced pattern: at $\rho = 0.9$, the scalar baseline is competitive or superior on test MSE across all three scale settings ($(n, p) \in \{(100, 150), (200, 300), (500, 750)\}$), while the proposed method achieves substantially stronger F_1 support recovery in every case. This reflects that lower inter-feature correlation reduces the severity of biactive starvation, so scalar tuning retains reasonable predictive performance even without correct support identification.

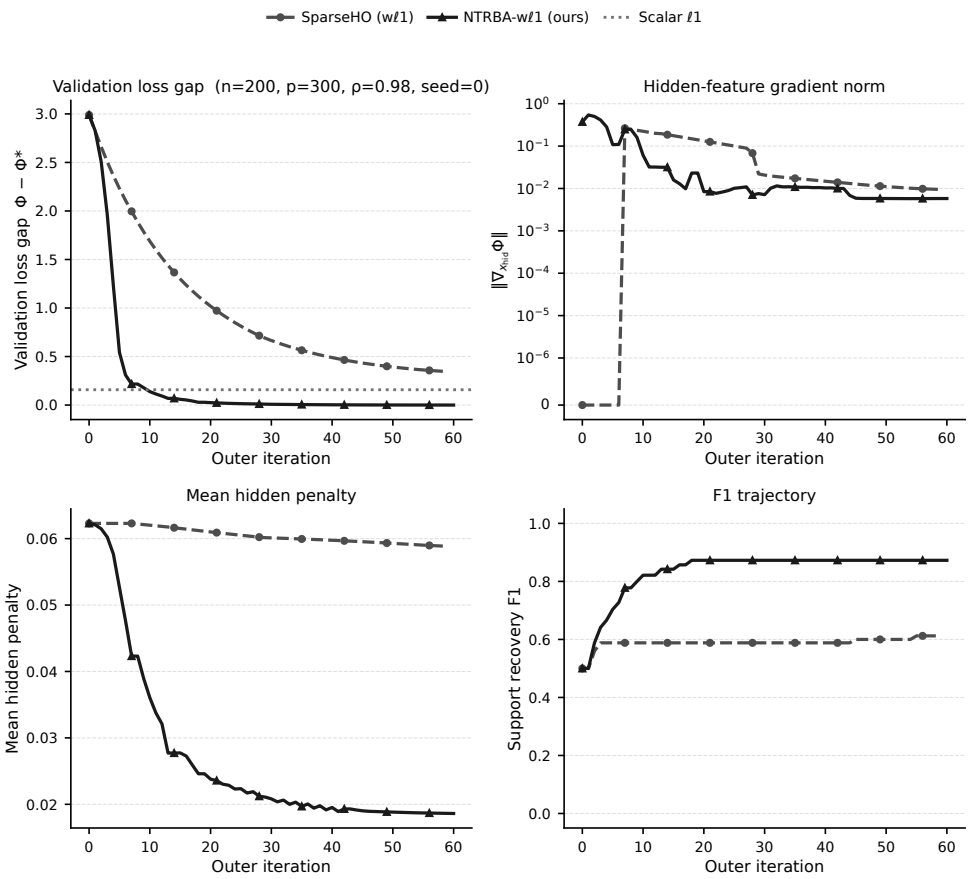

Fig. 5 Optimization trajectories on a representative calibrated degenerate instance. The panels compare scalar ℓ_1 tuning, weighted `Sparse-HO`, and the proposed `NTRBA-`$w\ell_1$ method for $(n, p) = (200, 300)$, $\rho = 0.98$, and seed 0. The proposed method rapidly reduces the validation-loss gap, decreases the penalties assigned to hidden relevant features, and improves support-recovery F_1. In contrast, the support-restricted `Sparse-HO` subgradient provides little useful signal on the hidden biactive coordinates, leading to slower progress and weaker recovery.

Figure 5 illustrates why feature-wise regularization alone does not fully resolve the difficulty of the problem. Although `Sparse-HO` uses individual penalties, its support-restricted subgradient assigns no initial signal to the hidden biactive coordinates, so these relevant variables remain weakly represented in the outer update. This explains its generally intermediate performance between scalar tuning and the proposed method. By restoring descent information on selected biactive coordinates, the proposed oracle yields a more stable validation trajectory and stronger hidden-support recovery.

The proposed method combines feature-wise regularization with self-consistent biactive selection and trust-region outer updates. In Figure 5, this is reflected in a more regular decrease of the validation objective on the representative instance. Table 3 shows that this behavior is accompanied by improved hidden-feature recovery and stronger overall support-recovery scores. Overall, the main benefit comes from the combination of weighted regularization with a subgradient oracle and outer solver that remain effective in the presence of biactive coordinates.

5.5 High-Dimensional Sparse Classification Benchmarks

This experiment extends the study to sparse classification problems and is divided into two complementary settings. The objective is to assess whether the behavior observed on synthetic degenerate instances remains relevant in more realistic regimes, where the feature geometry is induced by real data and the task is logistic classification rather than quadratic regression. In both settings, we compare scalar and feature-wise regularization strategies, with particular attention to the effect of biactive coordinates on hyperparameter optimization.

Setting 1 is semi-synthetic: it uses real sparse feature matrices to provide a realistic correlation structure, but injects a known sparse ground-truth signal. This makes it possible to evaluate not only predictive performance but also support recovery and recovery of hidden relevant features. Setting 2 uses binary classification datasets without artificial signal injection, so the evaluation focuses on predictive accuracy, sparsity, and optimization behavior. The datasets used in this experiment are summarized in Table 4.

5.5.1 Semi-Synthetic Benchmarks

We start from real sparse text matrices (RCV1, Real-sim, W8A, News20) and the same calibration principle is adapted to sparse logistic regression. We use 10 injected Gaussian easy features, 15 distractor/hidden pairs selected from the most highly correlated real features among the top-variance background coordinates, and up to 500 background features. Labels are generated from a sparse logistic model with coefficients ± 1 on the easy and hidden groups and additive Gaussian logit noise with standard deviation 0.20. The hidden penalties are calibrated by the same two-pass rule, replacing the squared-loss gradient by the smooth logistic gradient.

The lower-level problem is weighted sparse logistic regression with feature-wise penalties and fixed ridge parameter $\alpha_{\ell_2} = 1/n_{\text{train}}$. The upper-level objective is the held-out logistic loss. The initialization is calibrated in two passes so that the hidden

Table 3 Comparison on degenerate synthetic instances (mean ± std over 5 seeds for each $(\rho, (n, p))$ setting). $\|\partial_{x_{\text{hid}}} \Phi\|_{k=0}$ is the hidden-feature subgradient norm at the first outer iteration (not defined for the scalar baseline). Gray shading marks the best value in each $(\rho, (n, p), \text{metric})$ block.

ρ	(n, p)	Method	$\|\partial_{x_{\text{hid}}} \Phi\|_{k=0} \uparrow$	Best val. loss ↓	F1 ↑	Test MSE ↓
0.9	(100, 150)	Scalar ℓ_1	—	1.137 ± 1.099	0.468 ± 0.055	1.04 ± 1.26
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	1.612 ± 1.936	0.628 ± 0.129	1.96 ± 1.40
		`NTRBA`-$w\ell_1$ (ours)	1.69 ± 0.52	1.456 ± 2.835	0.772 ± 0.207	0.77 ± 1.36
	(200, 300)	Scalar ℓ_1	—	0.128 ± 0.047	0.510 ± 0.019	0.14 ± 0.03
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	1.442 ± 0.803	0.672 ± 0.068	3.15 ± 0.76
		`NTRBA`-$w\ell_1$ (ours)	2.77 ± 1.22	0.087 ± 0.034	0.883 ± 0.024	0.40 ± 0.29
	(500, 750)	Scalar ℓ_1	—	0.188 ± 0.107	0.513 ± 0.022	0.20 ± 0.11
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	5.109 ± 1.741	0.677 ± 0.053	8.31 ± 1.23
		`NTRBA`-$w\ell_1$ (ours)	5.12 ± 0.61	0.320 ± 0.073	0.893 ± 0.024	0.97 ± 0.38
0.95	(100, 150)	Scalar ℓ_1	—	1.312 ± 1.542	0.458 ± 0.057	1.18 ± 1.62
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	1.114 ± 1.649	0.670 ± 0.061	0.82 ± 0.57
		`NTRBA`-$w\ell_1$ (ours)	0.83 ± 0.30	0.025 ± 0.010	0.886 ± 0.055	0.04 ± 0.03
	(200, 300)	Scalar ℓ_1	—	0.139 ± 0.048	0.505 ± 0.028	0.16 ± 0.03
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	0.946 ± 0.680	0.673 ± 0.047	1.88 ± 0.82
		`NTRBA`-$w\ell_1$ (ours)	1.33 ± 0.63	0.043 ± 0.014	0.870 ± 0.027	0.19 ± 0.12
	(500, 750)	Scalar ℓ_1	—	0.226 ± 0.118	0.501 ± 0.019	0.23 ± 0.11
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	3.196 ± 1.237	0.669 ± 0.035	4.83 ± 1.28
		`NTRBA`-$w\ell_1$ (ours)	2.77 ± 0.34	0.155 ± 0.031	0.896 ± 0.026	0.44 ± 0.18
0.98	(100, 150)	Scalar ℓ_1	—	1.250 ± 1.649	0.437 ± 0.056	1.13 ± 1.71
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	0.138 ± 0.052	0.702 ± 0.061	0.21 ± 0.07
		`NTRBA`-$w\ell_1$ (ours)	0.30 ± 0.15	0.026 ± 0.014	0.855 ± 0.080	0.05 ± 0.05
	(200, 300)	Scalar ℓ_1	—	0.138 ± 0.045	0.480 ± 0.027	0.15 ± 0.02
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	0.784 ± 0.723	0.650 ± 0.032	1.27 ± 0.94
		`NTRBA`-$w\ell_1$ (ours)	0.44 ± 0.18	0.020 ± 0.007	0.855 ± 0.024	0.07 ± 0.03
	(500, 750)	Scalar ℓ_1	—	0.254 ± 0.111	0.493 ± 0.020	0.26 ± 0.10
		`Sparse-HO` ($w\ell_1$)	0.00 ± 0.00	2.220 ± 0.704	0.657 ± 0.025	3.23 ± 1.08
		`NTRBA`-$w\ell_1$ (ours)	1.31 ± 0.13	0.100 ± 0.055	0.869 ± 0.028	0.26 ± 0.15

relevant coordinates are inactive for the lower-level solution but lie within the biactive detection tolerance at the starting point.

The three methods and outer-iteration budget are identical to the baseline comparison in Section 5.4; the lower-level problem switches to weighted sparse logistic regression in Experiment 5.

The results in Table 5 show that the gradient-starvation phenomenon is not an artifact of synthetic Gaussian designs: it persists when the feature correlation structure comes from real text data and the loss function switches from quadratic to logistic. Hidden-feature recovery under the proposed oracle remains the strongest across all four datasets, while predictive loss stays competitive.

Table 4 Dataset characteristics for Experiment 5. Density is the percentage of non-zero entries in the feature matrix. Avg. nnz/sample denotes the average number of non-zero features per example.

Dataset	Used in	Samples	Features	Density (%)	Avg. nnz/sample	Pos. rate (%)
RCV1	S1+S2	20,242	47,236	0.157%	74.1	51.8%
Real-sim	S1+S2	72,309	20,958	0.245%	51.3	30.8%
W8A	S1	49,749	300	3.88%	11.7	2.97%
News20	S1+S2	19,996	1,355,191	0.034%	455	50.0%
Phishing	S2	11,055	68	44.1%	30.0	55.7%
Mnist (0/1)	S2	12,665	780	17.4%	135	53.2%

Table 5 Experiment 5 Setting 1: semi-synthetic sparse-text benchmark built from real sparse design matrices, reported as mean ± std over 5 train/val/test splits. Active features is the percentage of non-zero coefficients in the final inner solution; lower is better. Gray shading marks the best value within each dataset block.

		Recovery	Sparsity	Classification	Predictive loss	Efficiency
Dataset	Method	Hidden recall ↑	Act. feat. (%) ↓	F1	Log-loss	Runtime (s)
RCV1	Scalar ℓ_1 (grid)	0.133 ± 0.060	2.8 ± 0.2	0.609 ± 0.037	0.054 ± 0.001	92.7 ± 1.7
	`Sparse-HO` ($w\ell_1$)	0.120 ± 0.065	2.3 ± 0.2	0.640 ± 0.037	0.066 ± 0.001	16.9 ± 0.2
	`NTRBA-`$w\ell_1$ (ours)	0.387 ± 0.129	3.7 ± 0.9	0.717 ± 0.020	0.060 ± 0.010	47.7 ± 9.1
Real-sim	Scalar ℓ_1 (grid)	0.000 ± 0.000	2.0 ± 0.0	0.571 ± 0.000	0.053 ± 0.002	279.4 ± 3.2
	`Sparse-HO` ($w\ell_1$)	0.133 ± 0.094	2.4 ± 0.3	0.647 ± 0.052	0.066 ± 0.001	47.0 ± 0.8
	`NTRBA-`$w\ell_1$ (ours)	0.453 ± 0.204	3.8 ± 1.0	0.754 ± 0.076	0.056 ± 0.009	181.1 ± 28.1
W8A	Scalar ℓ_1 (grid)	0.640 ± 0.033	30.8 ± 1.1	0.325 ± 0.006	0.049 ± 0.001	107.7 ± 1.0
	`Sparse-HO` ($w\ell_1$)	0.013 ± 0.027	3.3 ± 0.1	0.579 ± 0.016	0.066 ± 0.002	11.5 ± 0.2
	`NTRBA-`$w\ell_1$ (ours)	0.667 ± 0.000	8.4 ± 1.6	0.792 ± 0.079	0.057 ± 0.011	21.7 ± 6.0
News20	Scalar ℓ_1 (grid)	0.107 ± 0.033	4.2 ± 0.4	0.498 ± 0.026	0.055 ± 0.003	99.2 ± 1.5
	`Sparse-HO` ($w\ell_1$)	0.160 ± 0.080	2.4 ± 0.2	0.662 ± 0.043	0.068 ± 0.003	15.8 ± 0.6
	`NTRBA-`$w\ell_1$ (ours)	0.240 ± 0.068	3.0 ± 0.5	0.673 ± 0.029	0.060 ± 0.010	36.8 ± 4.2

5.5.2 High-Dimensional Sparse Classification Benchmark

Setting 2 considers fully real binary classification tasks and removes the artificial signal injection used in Setting 1. Its purpose is to evaluate the proposed method in a standard predictive setting, where no ground-truth support is available and the quality of model selection must be assessed through predictive performance, sparsity, and optimization cost. This setting therefore complements the semi-synthetic benchmark by testing whether the weighted bilevel formulation remains useful when only observable task-level metrics can be measured.

The datasets cover a range of sparse and high-dimensional regimes, from moderate-scale classification problems to text-like representations with very large dimension. Setting 2 therefore tests both computational cost and model quality on real data.

Table 6 Experiment 5 Setting 2: real-world classification benchmarks (mean ± std over random splits). Active features is the fraction of features with non-zero weight at convergence. t/iter is wall-clock time per outer iteration. Gray shading marks the best value within each dataset block.

Dataset	Method	F1 ↑	Active features (%) ↓	t/iter (s) ↓
MNIST (0/1)	Scalar ℓ_1 (CV)	0.998 ± 0.001	34.530 ± 14.355	202.36 ± 1.55
	`Sparse-HO` ($w\ell_1$)	0.994 ± 0.001	1.239 ± 0.060	1.91 ± 0.01
	`NTRBA`-$w\ell_1$ (ours)	0.994 ± 0.001	2.991 ± 0.160	8.20 ± 0.52
NEWS20	Scalar ℓ_1 (CV)	0.957 ± 0.000	0.500 ± 0.006	808.33 ± 3.07
	`Sparse-HO` ($w\ell_1$)	0.821 ± 0.005	0.013 ± 0.000	4.06 ± 0.31
	`NTRBA`-$w\ell_1$ (ours)	0.830 ± 0.004	0.012 ± 0.000	15.18 ± 0.95
PHISHING	Scalar ℓ_1 (CV)	0.947 ± 0.004	81.373 ± 10.895	50.40 ± 0.24
	`Sparse-HO` ($w\ell_1$)	0.933 ± 0.003	14.706 ± 0.000	0.23 ± 0.00
	`NTRBA`-$w\ell_1$ (ours)	0.935 ± 0.002	14.706 ± 0.000	0.18 ± 0.04
RCV1	Scalar ℓ_1 (CV)	0.969 ± 0.002	6.390 ± 0.116	254.20 ± 3.62
	`Sparse-HO` ($w\ell_1$)	0.917 ± 0.004	0.230 ± 0.005	0.10 ± 0.01
	`NTRBA`-$w\ell_1$ (ours)	0.920 ± 0.003	0.227 ± 0.007	0.84 ± 0.04
REAL-SIM	Scalar ℓ_1 (CV)	0.951 ± 0.003	32.801 ± 0.170	2335.41 ± 15.25
	`Sparse-HO` ($w\ell_1$)	0.787 ± 0.005	0.286 ± 0.004	1.03 ± 0.11
	`NTRBA`-$w\ell_1$ (ours)	0.790 ± 0.006	0.283 ± 0.004	2.15 ± 0.07

The lower-level problem is weighted sparse logistic regression with feature-wise penalties and fixed ridge parameter $\alpha_{\ell_2} = 1/n_{\text{train}}$. The upper-level objective is the held-out logistic loss computed on a 60/20/20 train/validation/test split. Dense datasets are standardized using statistics computed on the training split only, while sparse datasets are kept in sparse CSC format. The initial hyperparameter is chosen as a uniform penalty equal to a fixed fraction of $\alpha_{\max}$, which provides a moderately sparse starting point.

The three methods and outer-iteration budget are identical to Setting 1. Since no ground-truth support is available in this setting, we report test F1 score, sparsity of the final model, and runtime per outer iteration.

The results in Table 6 indicate that the weighted bilevel methods remain competitive in fully real sparse classification problems, while producing models that are structurally different from those obtained by scalar tuning. The scalar baseline often provides a reasonable predictive reference, but its single global penalty offers limited control over how sparsity is distributed across features. This tends to restrict the range of models that can be selected.

Setting 2 does not involve controlled biactive initialization, so no recovery advantage is claimed. The question is whether the method degrades gracefully at scale. Table 6 shows that it does. Across all five datasets, `NTRBA`-$w\ell_1$ matches or marginally improves on `Sparse-HO` in F1. Sparsity is comparable on four of the five datasets; the exception is MNIST, where `NTRBA` retains roughly 2.4× more active features while achieving identical predictive performance. Per-iteration runtime overhead over `Sparse-HO` ranges from faster on PHISHING to roughly 2–8× on the larger datasets,

remaining well below scalar grid search in every case. This suggests that the self-consistent oracle does not hurt in regimes where biactive starvation is not the binding constraint.

From a computational viewpoint, Setting 2 also shows that the method remains practically usable on high-dimensional sparse data. The combination of proximal coordinate descent and trust-region outer updates makes it possible to handle large-scale problems without changing the underlying bilevel formulation.

6 Conclusions and future work

In this work, we studied a bilevel framework for learning feature-wise regularization parameters in sparse models. For the weighted ℓ_1 regularizer, we introduced an exact Forward–Backward reformulation of the lower-level problem that preserves the original solution set and yields a locally Lipschitz solution mapping. This reformulation allowed us to apply variational-analysis tools to characterize the coderivative of the solution map and to derive a computable element of the residual-enlarged subdifferential for the upper-level problem.

This analysis shows that biactive coordinates create a gradient-starvation phenomenon. To address this issue, we proposed a self-consistent biactive selection policy together with its support-reduced outer algorithms built with the resulting residual-enlarged subdifferential oracle.

The numerical results support the practical value of this approach. On synthetic regression problems, the proposed method shows that feature-wise regularization can substantially improve support recovery when compared with scalar tuning. On degenerate instances, the experiments illustrate that the proposed oracle resolves the gradient-starvation effect that limits support-restricted methods. Finally, on high-dimensional sparse classification benchmarks, the method remains effective in more realistic correlated-feature settings.

A natural next step for future work is to apply the present framework to other structured sparsity models, such as group penalties and related nonsmooth regularizers. Moreover, the derivation of convergence rate guarantees for `NTRBA` deserves a more detailed analysis.

Acknowledgements. The authors were supported by ANID Chile under grants Fondecyt Regular N° 1240120 (P. Pérez-Aros and E. Vilches), Fondecyt Regular N° 1220886 (P. Pérez-Aros and E. Vilches), Fondecyt Regular N° 1240335 (P. Pérez-Aros), Proyecto de Exploración N° 13220097 (P. Pérez-Aros and E. Vilches), CMM BASAL funds for Center of Excellence FB210005 (P. Pérez-Aros and E. Vilches), ECOS-ANID ECOS230027 (P. Pérez-Aros and E. Vilches), MATH-AMSUD AMSUD230036 (P. Pérez-Aros and E. Vilches), MATH-AMSUD AMSUD230018 (P. Pérez-Aros).

Declarations

Conflict of interest: The authors have no conflict of interest.

Appendix A Proof of Proposition 7

To characterize the Bouligand tangent cone, we begin with its formal definition. Since the entire proof operates at a fixed point $(u, v, w) \in \operatorname{gph} \mathcal{T}$ (the cone is empty by convention at points outside the graph), the tangent cone is given by:

$$T((u,v,w), \operatorname{gph}\mathcal{T}) = \big\{(\delta^u, \delta^v, \delta^w) : \exists\, t_k \downarrow 0,\ \exists\{(u_k, v_k, w_k)\} \subset \operatorname{gph}\mathcal{T},\\ \text{s.t. } (\delta^u, \delta^v, \delta^w) = \lim_{k\to\infty} t_k^{-1}[(u_k, v_k, w_k) - (u, v, w)]\big\}.$$

This definition relies on sequences within the graph of $\mathcal{T}$; membership of the base point (u, v, w) in the graph is the standing assumption, while the closedness of $\operatorname{gph}\mathcal{T}$ ensures that limits of convergent graph sequences remain in the graph.

Part 1: Necessary Condition

Let $(\delta^u, \delta^v, \delta^w)$ be a vector that belongs to the tangent cone $T((u, v, w); \operatorname{gph}\mathcal{T})$. By definition, this implies the existence of a sequence of scale parameters $t_k \downarrow 0$ and a sequence of directions $(\delta^u_k, \delta^v_k, \delta^w_k)$ converging to $(\delta^u, \delta^v, \delta^w)$ such that the graph inclusion holds for every k (throughout, $\delta^\bullet_{i,k}$ denotes the i-th component of the direction $\delta^\bullet_k$):

$$w_i + t_k\delta^w_{i,k} = \tau(u_i + t_k\delta^u_{i,k}, v_i + t_k\delta^v_{i,k}). \tag{A1}$$

We will show that the limit vector $(\delta^u, \delta^v, \delta^w)$ satisfies conditions (18) by the local geometry of $\mathcal{T}$. We analyze the behavior of the sequence according to their corresponding index set.

Case 1: Strictly Active/Inactive Regions *($i \in \mathcal{A}(u,v) \cup \mathcal{I}^+(u,v) \cup \mathcal{I}^-(u,v)$)*
In these regimes, the base point (u_i, v_i) lies strictly inside a region where the operator is linear (or zero). Since these sets are open, it holds that for a sufficiently large k (i.e., small enough t_k), the perturbed point $(u_i + t_k\delta^u_{i,k}, v_i + t_k\delta^v_{i,k})$ cannot escape the region.

- For $i \in \mathcal{I}^+(u, v), (v_i > u_i)$ and k sufficiently large, the perturbation satisfies $v_{i,k} > u_{i,k}$ and the inclusion (A1) becomes:

$$w_i + t_k\delta^w_{i,k} = (v_i + t_k\delta^v_{i,k}) - (u_i + t_k\delta^u_{i,k}).$$

  Subtracting the base equation $w_i = v_i - u_i$ and dividing by t_k, we obtain the exact relationship for the sequence: $\delta^w_{i,k} = \delta^v_{i,k} - \delta^u_{i,k}$. Passing to the limit as $k \to \infty$, we recover $\delta^w_i = \delta^v_i - \delta^u_i$.
- For $i \in \mathcal{I}^-(u, v), (v_i < -u_i)$, the points stay in the negative inactive region. The operator implies $w_i + t_k\delta^w_{i,k} = v_i + t_k\delta^v_{i,k} + u_i + t_k\delta^u_{i,k}$. Simplification yields $\delta^w_{i,k} = \delta^v_{i,k} + \delta^u_{i,k}$, which converges to $\delta^w_i = \delta^v_i + \delta^u_i$.
- For $i \in \mathcal{A}(u, v), (|v_i| < u_i)$ the points remain in the zone where the operator is identically zero. Thus, $0 + t_k\delta^w_{i,k} = 0$, forcing $\delta^w_{i,k} = 0$ for all large k, and implying $\delta^w_i = 0$.

Case 2: Positive Boundary *($i \in \mathcal{B}^+(u,v)$)* Throughout Cases 2 and 3 we explicitly exclude the vertex $u_i = v_i = 0$, at which i belongs simultaneously to $\mathcal{B}^+(u, v)$

and $\mathcal{B}^-(u,v)$ and the domain constraint $u \in \mathbb{R}^p_+$ imposes the additional requirement $\delta^u_i \geq 0$ on tangent directions; this configuration is addressed in Remark 5 below. Accordingly, the base point here satisfies $v_i = u_i > 0$ and $w_i = 0$. Since $u_i > 0$, entering $\mathcal{I}^-(u,v)$ along the approximating sequence would require $v_{i,k} < -u_{i,k}$, that is, $t_k(\delta^v_{i,k} + \delta^u_{i,k}) < -2u_i < 0$, which fails for all k large enough. Hence the perturbed points remain in $\mathcal{I}^+(u,v) \cup \mathcal{B}^+(u,v) \cup \mathcal{A}(u,v)$, where $\tau(u_{i,k}, v_{i,k}) = \max(v_{i,k} - u_{i,k}, 0)$, and (A1) becomes:

$$t_k \delta^w_{i,k} = \max\left((v_i + t_k\delta^v_{i,k}) - (u_i + t_k\delta^u_{i,k}), 0\right) = t_k \max(\delta^v_{i,k} - \delta^u_{i,k}, 0),$$

where we used $v_i - u_i = 0$. Dividing by t_k, we reveal that the sequence elements are explicitly bound by the max function: $\delta^w_{i,k} = \max(\delta^v_{i,k} - \delta^u_{i,k}, 0)$. By the continuity of the maximum function, taking the limit $k \to \infty$ yields: $\delta^w_i = \max(\delta^v_i - \delta^u_i, 0)$.

Case 3: Negative Boundary ($i \in \mathcal{B}^-(u,v)$) In this index set it holds $v_i = -u_i < 0$ and $w_i = 0$. Since $u_i > 0$, reaching the positive inactive region $\mathcal{I}^+(u,v)$ along the approximating sequence would require $v_{i,k} > u_{i,k}$, that is, $t_k(\delta^v_{i,k} - \delta^u_{i,k}) > 2u_i > 0$, which fails for all k large enough. Hence, the perturbed points stay in $\mathcal{I}^-(u,v) \cup \mathcal{B}^-(u,v) \cup \mathcal{A}(u,v)$, where the operator is correctly represented by $\tau(u_i, v_i) = -\max(-v_i - u_i, 0)$. Then, (A1) becomes:

$$t_k \delta^w_{i,k} = -\max\left(-(v_i + t_k\delta^v_{i,k}) - (u_i + t_k\delta^u_{i,k}), 0\right).$$

Using the equality $v_i + u_i = 0$, the argument simplifies to $-t_k(\delta^v_{i,k} + \delta^u_{i,k})$. Again, factoring out positive t_k, it yields: $\delta^w_{i,k} = -\max(-(\delta^v_{i,k} + \delta^u_{i,k}), 0)$. Taking the limit $k \to \infty$ and applying the identity $-\max(-a, 0) = \min(a, 0)$ recovers the condition stated in (18):

$$\delta^w_i = \min(\delta^v_i + \delta^u_i, 0).$$

Thus, any tangent vector must necessarily obey the conditions derived from the local structure of $\mathcal{T}$.

Part 2: Sufficient Condition

To establish the reverse inclusion, we must demonstrate that any vector $(\delta^u, \delta^v, \delta^w)$ satisfying the conditions (18) is indeed a valid tangent direction. By the definition of the Bouligand tangent cone, it suffices to construct a specific path inside the graph of $\mathcal{T}$ that originates from (u, v, w) with direction $(\delta^u, \delta^v, \delta^w)$. Let us fix the input direction sequences to be constant, i.e., $\delta^u_k \equiv \delta^u$ and $\delta^v_k \equiv \delta^v$ for all $k \in \mathbb{N}$. We then define the required response δ^w_k via the exact difference quotient of the operator along this ray:

$$\delta^w_{i,k} := \frac{\tau(u_i + t_k\delta^u_i, v_i + t_k\delta^v_i) - w_i}{t_k}. \tag{A2}$$

By this construction, the point $(u + t_k\delta^u, v + t_k\delta^v, w + t_k\delta^w_k)$ lies exactly on gph $\mathcal{T}$ for every step t_k. The proof reduces to showing that this sequence δ^w_k converges to our hypothesized direction δ^w as $t_k \downarrow 0$. We analyze the convergence behavior index by index, exploiting the geometry of the soft-thresholding operator:

Case 1: Strictly Active/Inactive Regions *($i \in \mathcal{A}(u,v) \cup \mathcal{I}^+(u,v) \cup \mathcal{I}^-(u,v)$)* In these regions, the point (u_i, v_i) lies in the interior of a linear piece of the operator. Since these sets define open neighborhoods, for any sufficiently small step size $t_k > 0$, the perturbed point $(u_i + t_k\delta_i^u, v_i + t_k\delta_i^v)$ remains strictly within the same region. Consequently:

- For $i \in \mathcal{I}^+(u,v), (v_i > u_i)$, the operator evaluates the difference of its arguments:

$$\delta_{i,k}^w = \frac{(v_i + t_k\delta_i^v) - (u_i + t_k\delta_i^u) - (v_i - u_i)}{t_k} = \delta_i^v - \delta_i^u = \delta_i^w.$$

- For $i \in \mathcal{I}^-(u,v), (v_i < -u_i)$ the operator evaluates to the sum, and reads:

$$\delta_{i,k}^w = \frac{(v_i + t_k\delta_i^v) + (u_i + t_k\delta_i^u) - (v_i + u_i)}{t_k} = \delta_i^v + \delta_i^u = \delta_i^w.$$

- For $i \in \mathcal{A}(u,v), (|v_i| < u_i)$ the operator returns zero locally. Then, the quotient vanishes, i.e., $\delta_{i,k}^w = \delta_i^w = 0$.

Case 2: Positive Boundary *($i \in \mathcal{B}^+(u,v)$)* Here, the point satisfies $v_i = u_i > 0$ and $w_i = 0$. Since the direction is fixed and $v_i = u_i > 0$, we have $v_i + t_k\delta_i^v + u_i + t_k\delta_i^u = 2u_i + t_k(\delta_i^v + \delta_i^u) \to 2u_i > 0$, so the perturbed point remains outside $\mathcal{I}^-(u,v)$ for all small enough t_k, and the operator evaluates as $\tau(u_i + t_k\delta_i^u, v_i + t_k\delta_i^v) = \max(v_i + t_k\delta_i^v - u_i - t_k\delta_i^u, 0)$. Substituting into (A2):

$$\delta_{i,k}^w = \frac{\max((v_i + t_k\delta_i^v) - (u_i + t_k\delta_i^u), 0) - 0}{t_k} = \max(\delta_i^v - \delta_i^u, 0),$$

where we used the fact that $v_i - u_i = 0$. This perfectly reproduces the conditional logic of (18): if $\delta_i^v > \delta_i^u$, the limit is $\delta_i^v - \delta_i^u$; otherwise, it is 0.

Case 3: Negative Boundary *($i \in \mathcal{B}^-(u,v)$)* Here, $v_i = -u_i < 0$ and $w_i = 0$. Since the direction is fixed and $v_i = -u_i < 0$, we have $v_i + t_k\delta_i^v - u_i - t_k\delta_i^u = -2u_i + t_k(\delta_i^v - \delta_i^u) \to -2u_i < 0$, so the perturbed point remains outside $\mathcal{I}^+(u,v)$ for all small enough t_k, and the operator evaluates as $\tau(u_i + t_k\delta_i^u, v_i + t_k\delta_i^v) = -\max(-(v_i + t_k\delta_i^v) - (u_i + t_k\delta_i^u), 0)$. The difference quotient (A2) then becomes:

$$\delta_{i,k}^w = \frac{-\max(-(v_i + t_k\delta_i^v) - (u_i + t_k\delta_i^u), 0)}{t_k}.$$

Using $v_i + u_i = 0$, this simplifies to: $\delta_{i,k}^w = -\max(-(\delta_i^v + \delta_i^u), 0) = \min(\delta_i^v + \delta_i^u, 0)$, where we used the identity $-\max(-a, 0) = \min(a, 0)$. Again, the t_k factors cancel out. This is precisely the last condition in (18): if $\delta_i^v + \delta_i^u < 0$, the value is $\delta_i^v + \delta_i^u$; otherwise, it is 0.

In every regime, the difference quotient sequence δ_k^w constructed from the graph constraint converges (or is locally constant) to the specific values mandated by conditions (18). Thus, the vector $(\delta^u, \delta^v, \delta^w)$ is the limit of valid secant directions, proving that it belongs to the tangent cone $T((u,v,w); \operatorname{gph} \mathcal{T})$. □

Remark 5 (Vertex case) At a vertex index i with $u_i = v_i = 0$, both $i \in \mathcal{B}^+(u,v)$ and $i \in \mathcal{B}^-(u,v)$ hold simultaneously, so the two biactive index sets are no longer disjoint. Moreover, since $u_i = 0$ lies on the boundary of the domain $\mathbb{R}^p_+$, the domain constraint forces $\delta_i^u \geq 0$ for any admissible tangent direction. A direct calculation combining the arguments of Cases 2 and 3 under this constraint shows that

$$\delta_i^w = \max(\delta_i^v - \delta_i^u, 0) + \min(\delta_i^v + \delta_i^u, 0), \quad \delta_i^u \geq 0,$$

which unifies both biactive contributions. Proposition 7 is therefore stated and proved under the standing assumption $u_i > 0$ for every $i \in \mathcal{B}^+(u,v) \cup \mathcal{B}^-(u,v)$, which excludes this degenerate configuration.

Appendix B Proof of Proposition 8

We proceed in two steps: first, by characterizing the Fréchet normal cone $\widehat{N}(\cdot)$, and then by using the limiting process to obtain the limiting normal cone $N(\cdot)$.

Lemma 14 *The Fréchet normal cone for the graph of the component-wise soft-thresholding operator at $(u,v,w) \in \mathbb{R}^{3p}$ is given by:*

$$\widehat{N}((u,v,w); \operatorname{gph} \mathcal{T}) = \left\{ (\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \;\middle|\; \begin{cases} \nu_i = -\zeta_i, \;\; \omega_i = \zeta_i & \text{if } i \in \mathcal{I}^+(u,v), \\ \nu_i = \;\; \zeta_i, \;\; \omega_i = \zeta_i & \text{if } i \in \mathcal{I}^-(u,v), \\ \nu_i = 0, \;\; \omega_i = 0 & \text{if } i \in \mathcal{A}(u,v), \\ \nu_i = -\omega_i, \quad 0 \leq \omega_i \leq \zeta_i, & \text{if } i \in \mathcal{B}^+(u,v), \\ \nu_i = \;\; \omega_i, \quad \zeta_i \leq \omega_i \leq 0, & \text{if } i \in \mathcal{B}^-(u,v) \end{cases} \right\}$$

Proof Using the definition for the Fréchet normal cone at (u,v,w) we have that:

$$\widehat{N}((u,v,w); \operatorname{gph} \mathcal{T}) = \left\{ (\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \,|\, \langle (\nu, \omega, -\zeta), (\delta^u, \delta^v, \delta^w) \rangle \leq 0, \forall (\delta^u, \delta^v, \delta^w) \in T((u,v,w); \operatorname{gph} \mathcal{T}) \right\}.$$

Now, using the characterization for the tangent cone, we proceed to analyze the conditions imposed by this inner product inequality across all index sets: active, positive/negative inactive, and positive/negative biactive components.

Case 1: $i \in \mathcal{I}^+(u,v)$: In this regime, the tangent cone is characterized by the relation $\delta_i^w = \delta_i^v - \delta_i^u$. For a normal vector $(\nu_i, \omega_i, -\zeta_i)$ to belong to the Fréchet normal cone, it must satisfy the following inequality:

$$\nu_i \delta_i^u + \omega_i \delta_i^v - \zeta_i \delta_i^w = \nu_i \delta_i^u + \omega_i \delta_i^v - \zeta_i (\delta_i^v - \delta_i^u) \leq 0, \quad \forall (\delta_i^u, \delta_i^v) \in \mathbb{R}^2.$$

Rewriting, we obtain:

$$(\nu_i + \zeta_i)\delta_i^u + (\omega_i - \zeta_i)\delta_i^v \leq 0, \quad \forall(\delta_i^u, \delta_i^v) \in \mathbb{R}^2.$$

Since this inequality must hold for both positive and negative directions of δ_i^u and δ_i^v, the respective coefficients must necessarily vanish to avoid contradiction. This implies the following equalities:

$$\nu_i + \zeta_i = 0, \quad \omega_i - \zeta_i = 0.$$

Thus, the Fréchet normal cone in this index set reduces to:

$$\widehat{N}((u, v, w); \operatorname{gph} \mathcal{T}) = \left\{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} | \nu_i = -\zeta_i,\, \omega_i = \zeta_i, \forall i \in \mathcal{I}^+(u, v)\right\}.$$

Case 2: $i \in \mathcal{I}^-(u, v)$: We use the same procedure applied to Case 1, but considering in this index set the tangent cone is characterized as $\delta_i^w = \delta_i^v + \delta_i^u$.
Case 3: $i \in \mathcal{A}(u, v)$: In this case, the tangent cone is given by $\delta_i^w = 0$. For the normal vector $(\nu_i, \omega_i, -\zeta_i)$ to belong to the Fréchet normal cone, the following inequality must hold:

$$\nu_i\delta_i^u + \omega_i\delta_i^v - \zeta_i\delta_i^w = \nu_i\delta_i^u + \omega_i\delta_i^v \leq 0, \quad \forall(\delta_i^u, \delta_i^v) \in \mathbb{R}^2.$$

As in Case 1, this inequality must hold for both positive and negative directions of δ_i^u and δ_i^v. Then, the inequality can only be satisfied for all admissible directions if both coefficients vanish, i.e., $\nu_i = \omega_i = 0$ and $\zeta_i \in \mathbb{R}$. Consequently, the Fréchet normal cone in this index set is given by:

$$\widehat{N}((u, v, w); \operatorname{gph} \mathcal{T}) = \left\{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} | \nu_i = \omega_i = 0, \zeta_i \in \mathbb{R}, \forall i \in \mathcal{A}(u, v)\right\}.$$

Case 4: $i \in \mathcal{B}^+(u, v)$: This case is more nuanced, as the tangent cone is the union of two convex cones, $T = K_1 \cup K_2$, the normal cone is the intersection of their respective polars: $\widehat{N} = K_1^\circ \cap K_2^\circ$.
(a) First representation: $\delta_i^w = 0$, $\delta_i^v \leq \delta_i^u$
Here, the inequality reduces to:

$$\nu_i\delta_i^u + \omega_i\delta_i^v \leq 0, \quad \forall(\delta_i^u, \delta_i^v) \text{ s.t. } \delta_i^v \leq \delta_i^u,$$

which can be alternatively written as:

$$\langle(\delta_i^u, \delta_i^v), (\nu_i, \omega_i)\rangle \leq 0, \forall(\delta_i^u, \delta_i^v) \text{ s.t. } \langle(\delta_i^u, \delta_i^v), (-1, 1)\rangle \leq 0. \tag{B3}$$

Defining $q_i = (-1, 1)$ and the cone of variations generated by q_i as:

$$\Pi(q_i) := \{(\delta_i^u, \delta_i^v) : \langle(\delta_i^u, \delta_i^v), q_i\rangle \leq 0\},$$

condition (B3) states that $\langle(\delta_i^u, \delta_i^v), (\nu_i, \omega_i)\rangle \leq 0$ for all $(\delta_i^u, \delta_i^v) \in \Pi(q_i)$. By definition, this is equivalent to requiring that (ν_i, ω_i) belongs to the polar cone of $\Pi(q_i)$. Since $\Pi(q_i)$ is a half-space, its polar is the ray generated by its normal; thus, the condition is equivalent to:

$$(\nu_i, \omega_i) \in \Pi^\circ(q_i) \iff (\nu_i, \omega_i) = c_1 q_i \text{ for some } c_1 \geq 0.$$

(b) Second representation: $\delta_i^w = \delta_i^v - \delta_i^u$, $\delta_i^u \leq \delta_i^v$
In this regime, the inequality for the Fréchet normal cone becomes:

$$\nu_i\delta_i^u + \omega_i\delta_i^v - \zeta_i\delta_i^w = \nu_i\delta_i^u + \omega_i\delta_i^v - \zeta_i(\delta_i^v - \delta_i^u) \leq 0,\ \forall(\delta_i^u, \delta_i^v) \text{ s.t. } \delta_i^u \leq \delta_i^v.$$

Rewriting:

$$\langle(\delta_i^u, \delta_i^v), (\nu_i + \zeta_i, \omega_i - \zeta_i)\rangle \leq 0, \forall(\delta_i^u, \delta_i^v) \text{ s.t. } \langle(\delta_i^u, \delta_i^v), (1, -1)\rangle \leq 0. \tag{B4}$$

Using the same argument as in the first representation, let $\tilde{q}_i := (1, -1)$ and $\Pi(\tilde{q}_i)$ be its corresponding cone of variations. Then, (B4) requires that the vector $(\nu_i + \zeta_i, \omega_i - \zeta_i)$ has

a non-positive inner product with all elements of $\Pi(\tilde{q}_i)$. This is exactly the definition of the polar cone, establishing the equivalence:

$$(\nu_i + \zeta_i, \omega_i - \zeta_i) \in \Pi^\circ(\tilde{q}_i) \iff (\nu_i + \zeta_i, \omega_i - \zeta_i) = -c_2 q_i \text{ for some } c_2 \geq 0,$$

where we used that $\tilde{q}_i = -q_i$.

Finally, a triplet $(\nu_i, \omega_i, -\zeta_i)$ belongs to the Fréchet normal cone if and only if it satisfies the conditions derived in both (a) and (b) simultaneously:

$$(\nu_i, \omega_i) = c_1(-1, 1) \quad \wedge \quad (\nu_i + \zeta_i, \omega_i - \zeta_i) = c_2(1, -1).$$

Solving this system yields the explicit characterization:

$$\widehat{N}((u, v, w); \operatorname{gph} \mathcal{T}) = \left\{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} | \nu_i = -\omega_i, 0 \leq \omega_i \leq \zeta_i, \forall i \in \mathcal{B}^+(u, v)\right\}.$$

Case 5: $i \in \mathcal{B}^-(u, v)$: Unlike Case 4, the tangent cone is $\delta_i^w = \min\{\delta_i^v + \delta_i^u, 0\}$.
(a) First representation: $\delta_i^w = 0$, $\delta_i^u + \delta_i^v \geq 0$.

Here, the inequality reduces to:

$$\nu_i \delta_i^u + \omega_i \delta_i^v \leq 0, \quad \forall\, (\delta_i^u, \delta_i^v) \text{ s.t. } \delta_i^u + \delta_i^v \geq 0,$$

which, defining $p_i = (1, 1)$, it can be written as:

$$\langle (\delta_i^u, \delta_i^v), (\nu_i, \omega_i) \rangle \leq 0, \quad \forall (\delta_i^u, \delta_i^v) \text{ s.t. } \langle (\delta_i^u, \delta_i^v), p_i \rangle \geq 0. \tag{B5}$$

The admissible set $\{x \in \mathbb{R}^2 : \langle x, p_i \rangle \geq 0\}$ has polar cone $\{-\lambda p_i : \lambda \geq 0\}$. Hence $(\nu_i, \omega_i) \in \{-\lambda_1 p_i : \lambda_1 \geq 0\}$, that is:

$$(\nu_i, \omega_i) = -\lambda_1(1, 1) \text{ for some } \lambda_1 \geq 0.$$

(b) Second representation: $\delta_i^w = \delta_i^u + \delta_i^v$, $\delta_i^u + \delta_i^v \leq 0$.

In this regime, the inequality for the Fréchet normal cone becomes:

$$\nu_i \delta_i^u + \omega_i \delta_i^v - \zeta_i \delta_i^w = \nu_i \delta_i^u + \omega_i \delta_i^v - \zeta_i(\delta_i^u + \delta_i^v) \leq 0, \ \forall (\delta_i^u, \delta_i^v) \text{ s.t. } \delta_i^u + \delta_i^v \leq 0.$$

Rewriting:

$$\langle (\delta_i^u, \delta_i^v), (\nu_i - \zeta_i, \omega_i - \zeta_i) \rangle \leq 0, \quad \forall (\delta_i^u, \delta_i^v) \text{ s.t. } \langle (\delta_i^u, \delta_i^v), p_i \rangle \leq 0. \tag{B6}$$

Defining $\Sigma(p_i) := \{x \in \mathbb{R}^2 : \langle x, p_i \rangle \leq 0\}$, condition (B6) states that $(\nu_i - \zeta_i, \omega_i - \zeta_i)$ belongs to the polar cone $\Sigma^\circ(p_i) = \{\lambda_2 p_i : \lambda_2 \geq 0\}$; therefore:

$$(\nu_i - \zeta_i, \omega_i - \zeta_i) = \lambda_2(1, 1) \text{ for some } \lambda_2 \geq 0.$$

Finally, a triplet $(\nu_i, \omega_i, -\zeta_i)$ belongs to the Fréchet normal cone if and only if both conditions hold simultaneously:

$$(\nu_i, \omega_i) = -\lambda_1(1, 1) \quad \wedge \quad (\nu_i - \zeta_i, \omega_i - \zeta_i) = \lambda_2(1, 1), \quad \lambda_1, \lambda_2 \geq 0.$$

From the first, $\nu_i = \omega_i = -\lambda_1$. Substituting into the second gives $-\lambda_1 - \zeta_i = \lambda_2 \geq 0$, which requires $\zeta_i \leq -\lambda_1 = \omega_i \leq 0$. Hence:

$$\widehat{N}((u, v, w); \operatorname{gph} \mathcal{T}) = \left\{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = \omega_i, \ \zeta_i \leq \omega_i \leq 0, \ \forall i \in \mathcal{B}^-(u, v)\right\}.$$

□

Limiting normal cone: To obtain the limiting normal cone, we apply the limiting process defined by the Mordukhovich normal cone construction. Indeed, this normal cone is defined as:

$$N((u, v, w); \operatorname{gph} \mathcal{T}) = \limsup_{(u', v', w') \to (u, v, w)} \widehat{N}((u', v', w'); \operatorname{gph} \mathcal{T})$$

$$= \left\{ \lim_{k\to\infty} (\nu^k, \omega^k, -\zeta^k) \;\middle|\; \begin{array}{r} (\nu^k, \omega^k, -\zeta^k) \in \widehat{N}((u_k, v_k, w_k); \operatorname{gph} \mathcal{T}), \\ (u_k, v_k, w_k) \xrightarrow{\operatorname{gph} \mathcal{T}} (u, v, w) \end{array} \right\}.$$

For indices i in the strictly active or inactive sets, the Fréchet normal cone is locally constant. Thus, the limit simply recovers the Fréchet cone derived in Lemma 14.

We focus on the biactive case $i \in \mathcal{B}^+(u, v)$, where $u_i = v_i > 0$ and $w_i = 0$. By the closedness of the graph, approximating sequences (u_k, v_k, w_k) approaching the biactive point must belong to either the strictly active regime ($|v_k| < u_k$), the strictly positive inactive regime ($v_k > u_k$), or staying on the biactive boundary ($v_k = u_k$). We analyze the limit set generated by each path:

- **Sequence from Active Region:** Consider a sequence with $|v_k| < u_k$. The Fréchet normals satisfy $\nu_k = 0, \omega_k = 0$. Passing to the limit, we obtain the subset of normal vectors:

$$S_1 = \{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = 0, \omega_i = 0\}.$$

- **Sequence from Inactive Region:** Consider a sequence with $v_k > u_k$. The Fréchet normals satisfy $\nu_k = -\zeta_k, \omega_k = \zeta_k$. Passing to the limit yields:

$$S_2 = \{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = -\zeta_i, \omega_i = \zeta_i\}.$$

- **Sequence on Biactive Boundary:** Consider a sequence with $v_k = u_k$ (always biactive). In this case, $(\nu_k, \omega_k, -\zeta_k)$ belongs to the biactive Fréchet cone derived in Lemma 14. The limit of this constant set is the set itself:

$$S_3 = \{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = -\omega_i,\ 0 \le \omega_i \le \zeta_i\}.$$

Consequently, the limiting normal cone is the set-theoretic union of these limits:

$$N((u, v, w); \operatorname{gph} \mathcal{T})_i =$$
$$\{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid (\nu_i = 0, \omega_i = 0) \vee (\nu_i = -\zeta_i, \omega_i = \zeta_i) \vee (\nu_i = -\omega_i, 0 \le \omega_i \le \zeta_i)\}.$$

Geometrically, this set consists of two linear subspaces (S_1 and S_2) connected by a convex cone (S_3) in the region where $\zeta_i \ge 0$. This union explicitly captures the non-convex nature of the operator at the singularity.

We now turn to the biactive case $i \in \mathcal{B}^-(u, v)$, where $v_i = -u_i < 0$ and $w_i = 0$. Approximating sequences (u_k, v_k, w_k) approaching the biactive-negative point belong to either the strictly active regime ($v_k > -u_k$), the strictly negative inactive regime ($v_k < -u_k$), or the biactive boundary ($v_k = -u_k$). The limit sets generated by each path are:

- **Sequence from Active Region:** $|v_k| < u_k$. The Fréchet normals satisfy $\nu_k = 0$, $\omega_k = 0$. In the limit:

$$S_1 = \{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = 0,\ \omega_i = 0\}.$$

- **Sequence from Negative Inactive Region:** $v_k < -u_k$. The Fréchet normals satisfy $\nu_k = \zeta_k$, $\omega_k = \zeta_k$. In the limit:

$$S_2 = \{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = \zeta_i,\ \omega_i = \zeta_i\}.$$

- **Sequence on Biactive Boundary:** $v_k = -u_k$. At each such point the Fréchet cone from Case 5 of Lemma 14 applies. In the limit:

$$S_3 = \{(\nu, \omega, -\zeta) \in \mathbb{R}^{3p} \mid \nu_i = \omega_i,\ \zeta_i \le \omega_i \le 0\}.$$

Then, the limiting normal cone is the union $S_1 \cup S_2 \cup S_3$, which agrees with the characterization stated in Proposition 8. □

References

[1] Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B.: Algorithms for hyper-parameter optimization. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 24. Curran Associates, Inc., Red Hook, NY, USA (2011). https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf

[2] Snoek, J., Larochelle, H., Adams, R.P.: Practical bayesian optimization of machine learning algorithms. In: Pereira, F., Burges, C.J., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 25. Curran Associates, Inc., Red Hook, NY, USA (2012). https://proceedings.neurips.cc/paper_files/paper/2012/file/05311655a15b75fab86956663e1819cd-Paper.pdf

[3] Wang, Z., Hutter, F., Zoghi, M., Matheson, D., De Feitas, N.: Bayesian optimization in a billion dimensions via random embeddings. Journal of Artificial Intelligence Research **55**, 361–387 (2016)

[4] Pedregosa, F.: Hyperparameter optimization with approximate gradient. In: International Conference on Machine Learning, pp. 737–746 (2016). PMLR

[5] Ghadimi, S., Wang, M.: Approximation Methods for Bilevel Programming (2018). https://arxiv.org/abs/1802.02246

[6] Ji, K., Yang, J., Liang, Y.: Bilevel optimization: Convergence analysis and enhanced design. In: International Conference on Machine Learning, pp. 4882–4892 (2021). PMLR

[7] Kunapuli, G., Bennett, K.P., Hu, J., Pang, J.S.: Classification model selection via bilevel programming. Optimization Methods and Software **23**(4), 475–489 (2008) https://doi.org/10.1080/10556780802102586

[8] Moore, G., Bergeron, C., Bennett, K.P.: Model selection for primal svm. Machine Learning **85**(1), 175–208 (2011) https://doi.org/10.1007/s10994-011-5246-7

[9] Lorraine, J., Vicol, P., Duvenaud, D.: Optimizing millions of hyperparameters by implicit differentiation. In: International Conference on Artificial Intelligence and Statistics, pp. 1540–1552 (2020). PMLR

[10] Rajeswaran, A., Finn, C., Kakade, S.M., Levine, S.: Meta-learning with implicit gradients. Advances in neural information processing systems **32** (2019)

[11] Franceschi, L., Donini, M., Frasconi, P., Pontil, M.: Forward and reverse gradient-based hyperparameter optimization. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1165–1173. PMLR, Sydney, Australia (2017). https://proceedings.mlr.press/v70/franceschi17a.html

[12] Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., Pontil, M.: Bilevel programming for hyperparameter optimization and meta-learning. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 1568–1577. PMLR, Stockholm, Sweden (2018). https://proceedings.mlr.press/v80/franceschi18a.html

[13] Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society Series B: Statistical Methodology **58**(1), 267–288 (1996)

[14] Zhang, C.-H.: Nearly unbiased variable selection under minimax concave penalty. The Annals of Statistics **38**(2), 894–942 (2010) https://doi.org/10.1214/09-AOS729

[15] Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American statistical Association **96**(456), 1348–1360 (2001)

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

[17] Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., Knight, K.: Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society Series B: Statistical Methodology **67**(1), 91–108 (2005)

[18] Kunisch, K., Pock, T.: A bilevel optimization approach for parameter learning in variational models. SIAM Journal on Imaging Sciences **6**(2), 938–983 (2013)

https://doi.org/10.1137/120882706

[19] De los Reyes, J.C., Schönlieb, C.-B.: Image denoising: Learning the noise model via nonsmooth pde-constrained optimization. Inverse Problems and Imaging **7**(4), 1183–1214 (2013) https://doi.org/10.3934/ipi.2013.7.1183

[20] De los Reyes, J.C., Villacís, D.: Interpretable model learning in variational imaging: a bilevel optimization approach. IMA Journal of Applied Mathematics **89**(1), 85–122 (2023) https://doi.org/10.1093/imamat/hxad024

[21] De los Reyes, J.C., Villacís, D.: Optimality conditions for bilevel imaging learning problems with total variation regularization. SIAM Journal on Imaging Sciences **15**(4), 1646–1689 (2022) https://doi.org/10.1137/21M143412X

[22] Gao, L., Ye, J., Yin, H., Zeng, S., Zhang, J.: Value function based difference-of-convex algorithm for bilevel hyperparameter selection problems. In: Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S. (eds.) Proceedings of the 39th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 162, pp. 7164–7182. PMLR, Baltimore, Maryland, USA (2022). https://proceedings.mlr.press/v162/gao22j.html

[23] Gao, L.L., Ye, J.J., Yin, H., Zeng, S., Zhang, J.: Moreau Envelope Based Difference-of-weakly-Convex Reformulation and Algorithm for Bilevel Programs (2024). https://arxiv.org/abs/2306.16761

[24] Li, Q., Li, Z., Zemkoho, A.: Bilevel hyperparameter optimization for support vector classification: Theoretical analysis and a solution method. Mathematical Methods of Operations Research **96**(3), 315–350 (2022) https://doi.org/10.1007/s00186-022-00798-6

[25] Dempe, S., Zemkoho, A.B.: The bilevel programming problem: Reformulations, constraint qualifications and optimality conditions. Mathematical Programming **138**(1), 447–473 (2013) https://doi.org/10.1007/s10107-011-0508-5

[26] Moore, G.M., Bergeron, C., Bennett, K.P.: Nonsmooth bilevel programming for hyperparameter selection. In: 2009 IEEE International Conference on Data Mining Workshops, pp. 374–381 (2009). https://doi.org/10.1109/ICDMW.2009.74

[27] Ochs, P., Ranftl, R., Brox, T., Pock, T.: Bilevel optimization with nonsmooth lower level problems. In: Aujol, J.-F., Nikolova, M., Papadakis, N. (eds.) Scale Space and Variational Methods in Computer Vision, pp. 654–665. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-18461-6_52

[28] Solodov, M.V.: A bundle method for a class of bilevel nonsmooth convex minimization problems. SIAM Journal on Optimization **18**(1), 242–259 (2007) https://doi.org/10.1137/050647566

[29] Bertrand, Q., Klopfenstein, Q., Massias, M., Blondel, M., Vaiter, S., Gramfort, A., Salmon, J.: Implicit differentiation for fast hyperparameter selection in non-smooth convex learning. Journal of Machine Learning Research **23**(149), 1–43 (2022)

[30] Beck, A.: First-order Methods in Optimization. MOS-SIAM Ser. Optim., vol. 25, p. 475. Society for Industrial and Applied Mathematics, Philadelphia, PA (2017). https://doi.org/10.1137/1.9781611974997

[31] Mordukhovich, B.S.: Variational Analysis and Applications. Springer Monographs in Mathematics, p. 622. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-92775-6

[32] Rockafellar, R.T., Wets, R.J.-B.: Variational Analysis. Grundlehren Math. Wiss., vol. 317, p. 733. Springer, Berlin (1998). https://doi.org/10.1007/978-3-642-02431-3

[33] Stella, L., Themelis, A., Patrinos, P.: Forward–backward quasi-newton methods for nonsmooth optimization problems. Computational Optimization and Applications **67**(3), 443–487 (2017)

[34] Bauschke, H., Combettes, P.: Convex analysis and monotone operator theory in Hilbert spaces. CMS books in mathematics **10**, 978–1 (2011) https://doi.org/10.1007/978-3-319-48311-5

[35] Mordukhovich, B.S.: Variational Analysis and Generalized Differentiation. I. Grundlehren Math. Wiss., vol. 330, p. 579. Springer, Heidelberg (2006). https://doi.org/10.1007/3-540-31247-1

[36] Dontchev, A.L., Rockafellar, R.T.: Implicit Functions and Solution Mappings, 2nd edn. Springer Ser. Oper. Res. Financ. Eng., p. 466. Springer, New York (2014). https://doi.org/10.1007/978-1-4939-1037-3

[37] Qi, L., Sun, J.: A trust region algorithm for minimization of locally Lipschitzian functions. Mathematical Programming **66**(1), 25–43 (1994) https://doi.org/10.1007/BF01581136