

Extrapolated Plug-and-Play Three-Operator Splitting Methods for Nonconvex Optimization with Applications to Image Restoration*

Zhongming Wu[†], Chaoyan Huang[‡], and Tieyong Zeng[§]

Abstract. This paper investigates the convergence properties and applications of the three-operator splitting method, also known as Davis-Yin splitting (DYS) method, integrated with extrapolation and Plug-and-Play (PnP) denoiser within a nonconvex framework. We first propose an extrapolated DYS method to effectively solve a class of structural nonconvex optimization problems that involve minimizing the sum of three possible nonconvex functions. Our approach provides an algorithmic framework that encompasses both extrapolated forward-backward splitting and extrapolated Douglas-Rachford splitting methods. To establish the convergence of the proposed method, we rigorously analyze its behavior based on the Kurdyka-Łojasiewicz property, subject to some tight parameter conditions. Moreover, we introduce two extrapolated PnP-DYS methods with convergence guarantee, where the traditional regularization prior is replaced by a gradient step-based denoiser. This denoiser is designed using a differentiable neural network and can be reformulated as the proximal operator of a specific nonconvex functional. We conduct extensive experiments on image deblurring and image super-resolution problems, where our results showcase the advantage of the extrapolation strategy and the superior performance of the learning-based model that incorporates the PnP denoiser in terms of achieving high-quality recovery images.

Key words. Plug-and-Play, three-operator splitting method, nonconvex optimization, denoising prior, convergence guarantee

AMS subject classifications. 90C26, 90C30, 90C90, 65K05

1. Introduction. In this paper, we consider the following type of structural nonconvex optimization problem:

$$(1.1) \quad \min_{\mathbf{x} \in \mathbb{R}^n} F(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x}) + h(\mathbf{x}),$$

where f_1 and h are continuously differentiable and potentially nonconvex, and f_2 is a proper closed (possibly nonconvex) function. The model (1.1) captures a rich number of applications in fields of deep learning, signal and image processing, and statistical learning, see e.g., [7, 14, 15, 29, 74, 75, 76]. In particular, the smooth term includes the least squares or logistic loss functions, and the nonsmooth term can be represented as regularizers, e.g., to promote potential behavior such as sparsity and low-rank.

*Submitted to the editors October 22, 2023.

Funding: This work was supported by Grant NSFC/RGC N_CUHK 415/19, Grant ITF ITS/173/22FP, Grant RGC 14300219, 14302920, 14301121, and CUHK Direct Grant for Research, the National Natural Science Foundation of China Grant 12001286, and the China Postdoctoral Science Foundation Grant 2022M711672.

[†]Co-first author. School of Management Science and Engineering, Nanjing University of Information Science and Technology, Nanjing, China (wuzm@nuist.edu.cn).

[‡]Co-first author. Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong, China (cychuang@math.cuhk.edu.hk).

[§]Corresponding author. Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong, China (zeng@math.cuhk.edu.hk).

Splitting methods, which fully leverage the inherent separable structure, is a class of popular and state-of-the-art approaches for effectively addressing structural optimization problems. A generic way to solve the type of problem (1.1) is the three-operator splitting method, also known as Davis-Yin splitting (DYS) method which was first studied in [16] for convex optimization, i.e., all the involved functions in (1.1) are convex. The concrete iterative scheme of DYS method can be read as

$$(1.2) \quad \begin{cases} \mathbf{y}^{k+1} \in \arg \min_{\mathbf{y} \in \mathbb{R}^n} \left\{ f_1(\mathbf{y}) + \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{x}^k\|^2 \right\}, \\ \mathbf{z}^{k+1} \in \arg \min_{\mathbf{z} \in \mathbb{R}^n} \left\{ f_2(\mathbf{z}) + \frac{1}{2\gamma} \|\mathbf{z} - (2\mathbf{y}^{k+1} - \gamma \nabla h(\mathbf{y}^{k+1}) - \mathbf{x}^k)\|^2 \right\}, \\ \mathbf{x}^{k+1} = \mathbf{x}^k + (\mathbf{z}^{k+1} - \mathbf{y}^{k+1}), \end{cases}$$

where $\gamma > 0$ is a proximal parameter. DYS method (1.2) includes two proximal subproblems with respect to \mathbf{y} and \mathbf{z} , which extends various previous splitting schemes such as the forward-backward splitting (FBS) method [3], Douglas-Rachford splitting (DRS) method [34], alternating direction method of multipliers (ADMM) algorithm [23] and the generalized forward-backward splitting method [51]. Later on, some variants and extensions of DYS method are explored for convex optimization [32, 56, 57, 62]. However, for the nonconvex setting as that in (1.1), convergence properties of the DYS method (1.2) are less understood. In contrast, the FBS method and DRS method, two special cases of the DYS method, have been well studied for nonconvex optimization, see e.g., [3, 34, 63]. Indeed, splitting methods are widely employed in image processing because numerous problems in image restoration can be addressed through variational methods. The resulting image is obtained as a minimizer of a suitable energy functional, typically exhibiting a separable structure. For recent applications in this field, we refer to [17, 40, 58, 62].

Another captivating and intriguing topic within the realm of splitting methods is the incorporation of acceleration techniques. Since the pioneering work of Polyak [50] on the heavy-ball method approach to gradient descent, extrapolation, as well as named inertial strategy, has been adapted to various optimization schemes to achieve accelerated convergence. Notable examples include the accelerated proximal point algorithm [12] for variational inequality problems and the accelerated FBS [4, 68, 6] for convex optimization. Over the past decade, the extrapolation technique has also been extended to various splitting methods for solving nonconvex optimization problems and expediting convergence based on Kurdyka–Łojasiewicz framework (see [Definition 2.2](#)), as demonstrated in studies such as [45, 33, 69, 37, 49, 73, 72, 48]. In this paper, our first focus is to investigate the convergence properties of the DYS method (1.2) when combined with extrapolation technique for solving (1.1). This endeavor will result in the development of a versatile framework encompassing extrapolated (or named inertial) FBS and extrapolated DRS methods as specialized schemes tailored for nonconvex optimization.

Recently, Plug-and-Play (PnP) methods combine splitting algorithm with denoising priors are widely used in solving many practical problems [19, 70, 71, 35]. PnP method offers a concise yet adaptable approach for integrating statistical priors into a problem, eliminating the requirement to explicitly construct an objective function. The first PnP method was the PnP-ADMM developed in [67] to address a range of imaging problems, which simply replaces the proximal subproblem with the denoising prior. Since then, many PnP-based methods

such as PnP-FBS [59, 66], PnP-DRS [10, 28] and PnP-primal dual [46] approaches, reported empirical success on a large variety of applications, but with scarce theoretical guarantees. In several recent studies, the convergence of PnP methods has been achieved through the utilization of contractive fixed-point iterations. For example, the convergence of various proximal algorithms has been established by assuming properties such as denoiser averaging [60], firm nonexpansiveness [61], or simple nonexpansiveness [41, 52]. However, it is important to note that off-the-shelf deep denoisers often lack 1-Lipschitz continuity, which is equivalent to nonexpansiveness. The imposition of strict Lipschitz constraints on the network adversely affects its denoising performance [24, 28].

To address the challenge of nonexpansiveness in deep denoisers, Ryu et al. [55] proposed a method where each layer is individually normalized using its spectral norm. However, this approach imposes limitations on the utilization of residual skip connections, which are widely employed in deep denoisers. In a recent study, Hurault et al. [27] tackled this issue by training a deep image denoiser using a gradient-based PnP prior. By replacing the regularization step with the constructed denoiser, they demonstrated that the resulting gradient step PnP prior corresponds to the proximal operator of a specific nonconvex functional [28]. Under this condition, they successfully established the convergence of PnP-FBS, PnP-ADMM, and PnP-DRS iterates towards stationary points of explicit functions. Inspired by this research direction, it is worth exploring the convergence guarantees and potential applications of combining PnP methods with the DYS algorithm (1.2) in the form of (1.1).

1.1. Our contribution. This paper provides a generic algorithm framework that combines splitting methods, extrapolation strategy, and deep prior. The main contributions of this paper are threefold:

- We propose an extrapolated DYS method for solving the type of structural nonconvex optimization problem (1.1), which provides a generic algorithm framework including extrapolated FBS and extrapolated DRS methods. Under the tight parameter conditions, the convergence of the generated iterates is established based on Kurdyka–Łojasiewicz framework.
- By replacing the regularization step with the gradient step-based denoiser, we propose two extrapolated PnP-DYS methods. The denoiser is constructed by a differentiable neural network and can be reformulated as the proximal operator of a specific nonconvex functional. The convergence of both PnP-DYS algorithms is also established.
- Extensive experiments on image deblurring and image super-resolution problems are conducted to evaluate the performance of the proposed schemes. The numerical results illustrate the advantages and efficiency of the extrapolation strategy. Moreover, the experiments reveal the superiority of the PnP-based model with deep denoiser in terms of the quality of the recovered images.

1.2. Organization. The remainder of this paper is organized as follows. Some related methods and preliminaries are reviewed in Section 2. An extrapolated DYS method with convergence analysis is developed in Section 3. Section 4 combines PnP approach and produces two extrapolated PnP-DYS methods with convergence guarantee. Some experimental results are reported in Section 5, and the conclusions follow in Section 6.

1.3. Notation. We use \mathbb{R}^n to denote the n -dimensional Euclidean space, \mathbb{R}_+ to denote the set of nonnegative real numbers, $\langle \cdot, \cdot \rangle$ to denote the inner product, and $\|\cdot\|$ to denote the norm induced from the inner product. For an extended real-valued function f , the domain of f is defined as $\text{dom}f := \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) < \infty\}$. We say that the function f is proper if $\text{dom}f \neq \emptyset$ and $f(\mathbf{x}) > -\infty$ for any $\mathbf{x} \in \text{dom}f$, and is closed if it is lower semicontinuous. For any subset $S \subseteq \mathbb{R}^n$ and any point $\mathbf{x} \in \mathbb{R}^n$, the distance from \mathbf{x} to S is defined by $\text{dist}(\mathbf{x}, S) := \inf \{\|\mathbf{y} - \mathbf{x}\| \mid \mathbf{y} \in S\}$, and $\text{dist}(\mathbf{x}, S) = \infty$ for all \mathbf{x} when $S = \emptyset$.

2. Preliminaries. In this section, we review the definitions of subdifferential and Kurdyka-Łojasiewicz (KL) property for further analysis.

Definition 2.1. [3, 8] (Subdifferentials) Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a proper and lower semicontinuous function.

- (i) For a given $\mathbf{x} \in \text{dom}f$, the Fréchet subdifferential of f at \mathbf{x} , written by $\widehat{\partial}f(\mathbf{x})$, is the set of all vectors $\mathbf{u} \in \mathbb{R}^n$ satisfying

$$\liminf_{\mathbf{y} \neq \mathbf{x}, \mathbf{y} \rightarrow \mathbf{x}} \frac{f(\mathbf{y}) - f(\mathbf{x}) - \langle \mathbf{u}, \mathbf{y} - \mathbf{x} \rangle}{\|\mathbf{y} - \mathbf{x}\|} \geq 0,$$

and we set $\widehat{\partial}f(\mathbf{x}) = \emptyset$ when $\mathbf{x} \notin \text{dom}f$.

- (ii) The limiting-subdifferential, or simply the subdifferential, of f at \mathbf{x} , written by $\partial f(\mathbf{x})$, is defined by

$$(2.1) \quad \partial f(\mathbf{x}) := \{\mathbf{u} \in \mathbb{R}^n \mid \exists \mathbf{x}^k \rightarrow \mathbf{x}, \text{ s.t. } f(\mathbf{x}^k) \rightarrow f(\mathbf{x}) \text{ and } \widehat{\partial}f(\mathbf{x}^k) \ni \mathbf{u}^k \rightarrow \mathbf{u}\}.$$

- (iii) A point \mathbf{x}^* is called (limiting-)critical point or stationary point of f if it satisfies $0 \in \partial f(\mathbf{x}^*)$, and the set of critical points of f is denoted by $\text{crit}f$.

Definition 2.1 implies that the property $\widehat{\partial}f(\mathbf{x}) \subseteq \partial f(\mathbf{x})$ holds immediately, and $\widehat{\partial}f(\mathbf{x})$ is closed and convex while $\partial f(\mathbf{x})$ is closed. Indeed, the subdifferential (2.1) reduces to the gradient of f denoted by ∇f if f is continuously differentiable. Furthermore, as described in [53], if g is a continuously differentiable function, it holds that $\partial(f + g) = \partial f + \nabla g$.

Next, we recall the KL property [2, 8], which is important in the convergence analysis.

Definition 2.2. (KL property and KL function) Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a proper and lower semicontinuous function.

- (a) The function f is said to have KL property at $\mathbf{x}^* \in \text{dom}(\partial f)$ if there exist $\eta \in (0, +\infty]$, a neighborhood U of \mathbf{x}^* and a continuous and concave function $\varphi : [0, \eta) \rightarrow \mathbb{R}_+$ such that

- (i) $\varphi(0) = 0$ and φ is continuously differentiable on $(0, \eta)$ with $\varphi' > 0$;
(ii) for all $\mathbf{x} \in U \cap \{\mathbf{z} \in \mathbb{R}^n \mid f(\mathbf{x}^*) < f(\mathbf{z}) < f(\mathbf{x}^*) + \eta\}$, the following KL inequality holds:

$$(2.2) \quad \varphi'(f(\mathbf{x}) - f(\mathbf{x}^*))\text{dist}(0, \partial f(\mathbf{x})) \geq 1.$$

- (b) If f satisfies the KL property at each point of $\text{dom}(\partial f)$, then f is called a KL function.

Denote Φ_η as the set of functions φ which satisfy the involved conditions in **Definition 2.2**(a). Then, we give an uniformized KL property which was established in [8] in the following, it will be useful for further convergence analysis.

Lemma 2.3. [8] (Uniformized KL property) Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a proper and lower semicontinuous function and Ω be a compact set. Assume that f is a constant on Ω and satisfies the KL property at each point of Ω . Then, there exist $\varsigma > 0$, $\eta > 0$ and $\varphi \in \Phi_\eta$ such that

$$(2.3) \quad \varphi'(f(\mathbf{x}) - f(\bar{\mathbf{x}})) \text{dist}(0, \partial f(\mathbf{x})) \geq 1,$$

for all $\bar{\mathbf{x}} \in \Omega$ and each \mathbf{x} satisfying $\text{dist}(\mathbf{x}, \Omega) < \varsigma$ and $f(\bar{\mathbf{x}}) < f(\mathbf{x}) < f(\bar{\mathbf{x}}) + \eta$.

Below we give a well-known descent lemma for smooth functions in the literature and the detailed proof can be found in [44, Lemma 1.2.3].

Lemma 2.4. [44] Let $h : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuously differentiable function with gradient ∇h assumed L_h -Lipschitz continuous. Then, we have

$$(2.4) \quad \left| h(\mathbf{u}) - h(\mathbf{v}) - \langle \mathbf{u} - \mathbf{v}, \nabla h(\mathbf{v}) \rangle \right| \leq \frac{L_h}{2} \|\mathbf{u} - \mathbf{v}\|^2, \quad \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^n.$$

Lemma 2.5. [9] Let $\{a_n\}$ and $\{b_n\}$ be two nonnegative sequences satisfying $\sum_{n \in \mathbb{N}} b_n < \infty$ and $a_{n+1} \leq a \cdot a_n + b \cdot a_{n-1} + b_n$ for all $n \geq 1$, where $a \in \mathbb{R}$, $b \geq 0$ and $a + b < 1$. Then, we have $\sum_{n \in \mathbb{N}} a_n < \infty$.

3. Extrapolated DYS method with convergence analysis. In this section, we propose a general extrapolated DYS method and conduct the convergence analysis.

3.1. The extrapolated DYS method. We propose an extrapolated DYS algorithm to solve the general nonconvex optimization problem (1.1), where an extrapolation step is incorporated to accelerate the convergence speed. Note that for any $\gamma > 0$, the proximal operator of the function f is defined by

$$\text{Prox}_{\gamma f}(\mathbf{x}) = \arg \min_{\mathbf{y} \in \mathbb{R}^n} \left\{ f(\mathbf{y}) + \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{x}\|^2 \right\}.$$

We say that f is prox-bounded if $f + \frac{1}{2\gamma} \|\cdot\|^2$ is lower bounded for some $\gamma > 0$. The supremum of all such γ is the threshold of prox-boundedness of f , denoted as γ_f . If f is lower semicontinuous, then $\text{Prox}_{\gamma f}$ is nonempty and compact for all $\gamma \in (0, \gamma_f)$ [53, Theorem 1.25].

Algorithm 3.1 An extrapolated DYS method

Choose the parameters $\alpha \geq 0$ and $\gamma > 0$. Given \mathbf{x}^0 and $\mathbf{x}^{-1} = \mathbf{x}^0$, set $k = 0$.

while the stopping criteria is not satisfied, **do**

$$(3.1) \quad \begin{cases} \mathbf{w}^k = \mathbf{x}^k + \alpha(\mathbf{x}^k - \mathbf{x}^{k-1}), \\ \mathbf{y}^{k+1} = \text{Prox}_{\gamma f_1}(\mathbf{w}^k), \\ \mathbf{z}^{k+1} = \text{Prox}_{\gamma f_2}(2\mathbf{y}^{k+1} - \gamma \nabla h(\mathbf{y}^{k+1}) - \mathbf{w}^k), \\ \mathbf{x}^{k+1} = \mathbf{w}^k + (\mathbf{z}^{k+1} - \mathbf{y}^{k+1}). \end{cases}$$

end while

The concrete iterative scheme is summarized in [Algorithm 3.1](#), which provides a versatile algorithmic framework that encompasses both (extrapolated) forward-backward splitting and (extrapolated) Douglas-Rachford splitting methods. In particular, when the extrapolation step vanishes, i.e., $\alpha = 0$, [Algorithm 3.1](#) simplifies to the classical three-operator splitting method studied in [\[7, 16\]](#). When $f_1 = 0$ in [\(1.1\)](#), [Algorithm 3.1](#) reduces to the extrapolated (or named inertial) forward-backward splitting method, also known as inertial proximal gradient method, studied in [\[4, 43, 38, 73\]](#). [Algorithm 3.1](#) also recovers extrapolated Douglas-Rachford splitting method when $h = 0$.

Besides, when $\alpha = 0$ and the function h vanishes, [Algorithm 3.1](#) reduces to the classical DRS algorithm. The convergence of DRS method for nonconvex optimization was first discussed in [\[34\]](#), and then refined in [\[63\]](#). Some other variants and extensions of DRS method for nonconvex optimization can refer to [\[21, 22, 36, 39, 65\]](#). When $\alpha = 0$ and f_1 vanishes, the DYS algorithm becomes another very popular approach, namely, the forward-backward splitting (FBS) or proximal gradient method. We refer to [\[1, 3, 9, 64, 73\]](#) for the extension studies of FBS method in the nonconvex setting.

Next we present some assumptions for problem [\(1.1\)](#) to facilitate convergence analysis.

Assumption 3.1. The functions f_1, f_2 , and g in [\(1.1\)](#) satisfy the following conditions:

(i) f_1 has a Lipschitz continuous gradient, i.e., there exists a constant $L_{f_1} > 0$ such that

$$\|\nabla f_1(\mathbf{y}_1) - \nabla f_1(\mathbf{y}_2)\| \leq L_{f_1} \|\mathbf{y}_1 - \mathbf{y}_2\|, \quad \forall \mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^n.$$

(ii) h has a Lipschitz continuous gradient, i.e., there exists a constant $L_h > 0$ such that

$$\|\nabla h(\mathbf{y}_1) - \nabla h(\mathbf{y}_2)\| \leq L_h \|\mathbf{y}_1 - \mathbf{y}_2\|, \quad \forall \mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^n.$$

(iii) $f_2 : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ is a proper closed function, and the objective function F is bounded from below.

Let $l \in \mathbb{R}$ be a constant such that $f_1 + \frac{l}{2}\|\cdot\|^2$ is convex. It should be noted that the existence of such an l can be guaranteed by the Lipschitz continuity of ∇f_1 . Specifically, one can always choose $l = L_{f_1}$. In addition, it follows from the convexity of $f_1 + \frac{l}{2}\|\cdot\|^2$ that

$$f_1(\mathbf{y}_1) - f_1(\mathbf{y}_2) - \langle \nabla f_1(\mathbf{y}_2), \mathbf{y}_1 - \mathbf{y}_2 \rangle \geq -\frac{l}{2} \|\mathbf{y}_1 - \mathbf{y}_2\|^2, \quad \forall \mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^n.$$

Then, according to the Lipschitz continuity of ∇f_1 and [Lemma 2.4](#), it must holds that $l \geq -L_{f_1}$. Hence, there must exist a constant $l \in [-L_{f_1}, L_{f_1}]$ such that $f_1 + \frac{l}{2}\|\cdot\|^2$ is convex. Note that $l < 0$ implies that f_1 is strongly convex. Define

$$(3.2) \quad \Lambda(\gamma) := \frac{1 - \gamma l - 2\gamma L_h}{2 + \gamma L_h} - \gamma^2 L_{f_1}^2.$$

Now we give the parameter conditions for [Algorithm 3.1](#) in the following assumption.

Assumption 3.2. The parameters α and γ should be chosen such that $0 < \gamma < \frac{1}{L_{f_1} + L_h}$ and $0 \leq \alpha < \Lambda(\gamma)$.

Remark 3.3. Note that for given $L_{f_1} > 0$ and $L_h \geq 0$, $\Lambda(\gamma) > 0$ always holds if $\gamma > 0$ is sufficiently small. Moreover, for the case of $L_h = 0$, i.e., when $h = 0$, it is easy to determine that $\Lambda(\gamma) > 0$ if the following threshold for γ is satisfied:

$$(3.3) \quad 0 < \gamma < \frac{-l + \sqrt{l^2 + 8L_{f_1}^2}}{4L_{f_1}^2}.$$

The above relation implies that $\gamma < \frac{1}{L_{f_1}}$, since the maximum value of the upper bound can be attained when $l = -L_{f_1}$ for every fixed value of L_{f_1} . Indeed, when $h = 0$ and $\alpha = 0$, the extrapolated DYS algorithm (3.1) reduces to the classical DRS algorithm in [34, 63]. In this case, the range of γ specified in (3.3) is tighter compared to that in [34], particularly in terms of the larger upper bound. For $L_h > 0$, we can also provide a computable threshold for γ to ensure that **Assumption 3.2** holds, i.e., $0 < \gamma < \frac{1}{L_{f_1} + L_h}$ and $\Lambda(\gamma) > 0$, as follows:

$$(3.4) \quad 0 < \gamma < \min \left\{ \frac{1}{L_{f_1} + L_h}, \gamma_0 \right\},$$

$$\text{where } \gamma_0 := \frac{-(2L_h + L_{f_1} + l) + \sqrt{(2L_h + L_{f_1} + l)^2 + 4(L_h L_{f_1} + L_{f_1}^2)}}{2(L_h L_{f_1} + L_{f_1}^2)}.$$

Remark 3.4. When $\alpha = 0$, the extrapolated DYS algorithm (3.1) reduces to the method studied in [7, 42]. However, in this case, the range of γ based on **Assumption 3.2** is different from the result in [7] for the fixed L_{f_1} , L_h and l . Especially the upper bound of γ is different due to the distinct construction of $\Lambda(\gamma)$ in (3.2). In other words, as a byproduct, this paper provides an improved parameter condition for γ to ensure the convergence of the DYS method in the nonconvex setting. In addition, the lower boundedness of the energy function for the DYS method, and a certain sublinear convergence rate are established under some common conditions, which will be detailed later.

3.2. Convergence analysis. In this subsection, we prove the convergence of **Algorithm 3.1**, i.e., the extrapolated DYS algorithm, for the general nonconvex optimization problem (1.1) under **Assumption 3.1** and **Assumption 3.2**.

For convenience, we first present the corresponding first-order optimality conditions for the \mathbf{y} - and \mathbf{z} -subproblems in (3.1), which will be frequently utilized in the subsequent convergence analysis. Specifically, the optimality condition for \mathbf{y} -subproblem in (3.1) is

$$(3.5) \quad 0 = \nabla f_1(\mathbf{y}^{k+1}) + \frac{1}{\gamma} (\mathbf{y}^{k+1} - \mathbf{w}^k),$$

and that for \mathbf{z} -subproblem in (3.1) is

$$(3.6) \quad 0 \in \partial f_2(\mathbf{z}^{k+1}) + \frac{1}{\gamma} (\mathbf{z}^{k+1} + \gamma \nabla h(\mathbf{y}^{k+1}) - 2\mathbf{y}^{k+1} + \mathbf{w}^k).$$

To simplify the notations in our analysis, we denote

$$(3.7) \quad \mathbf{v}^k = (\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)^\top, \quad \mathbf{u}^k = (\mathbf{v}^k, \mathbf{x}^{k-1}, \mathbf{x}^{k-2})^\top, \quad \forall k \geq 1,$$

and

$$(3.8) \quad \Delta_{\mathbf{x}}^k = \mathbf{x}^k - \mathbf{x}^{k-1}, \quad \Delta_{\mathbf{y}}^k = \mathbf{y}^k - \mathbf{y}^{k-1}, \quad \forall k \geq 1.$$

Next, for $\gamma > 0$, we define an auxiliary function \mathcal{H}_γ as follows:

$$(3.9) \quad \begin{aligned} \mathcal{H}_\gamma(\mathbf{y}, \mathbf{z}, \mathbf{x}) &= f_1(\mathbf{y}) + f_2(\mathbf{z}) + h(\mathbf{y}) + \frac{1}{2\gamma} \|2\mathbf{y} - \mathbf{z} - \mathbf{x} - \gamma \nabla h(\mathbf{y})\|^2 \\ &\quad - \frac{1}{2\gamma} \|\mathbf{x} - \mathbf{y} + \gamma \nabla h(\mathbf{y})\|^2 - \frac{1}{\gamma} \|\mathbf{y} - \mathbf{z}\|^2 \\ &= f_1(\mathbf{y}) + f_2(\mathbf{z}) + h(\mathbf{y}) + \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{x} - \gamma \nabla h(\mathbf{y})\|^2 - \frac{1}{2\gamma} \|\mathbf{z} - \mathbf{x} - \gamma \nabla h(\mathbf{y})\|^2, \end{aligned}$$

which is motivated by the DYS envelope studied in [42] and also utilized in [7]. Based on the definition of \mathcal{H}_γ , we define the energy function associated with extrapolated DYS method (3.1) as follows:

$$(3.10) \quad \Theta_{\alpha, \gamma}(\mathbf{y}, \mathbf{z}, \mathbf{x}, \mathbf{x}_1, \mathbf{x}_2) = \mathcal{H}_\gamma(\mathbf{y}, \mathbf{z}, \mathbf{x}) + \frac{\alpha^2}{2\gamma} \|\mathbf{x}_1 - \mathbf{x}_2\|^2,$$

where $\alpha \geq 0$ is a constant parameter that remains consistent with that in [Algorithm 3.1](#).

We first show that the sequence $\{\Theta_{\alpha, \gamma}(\mathbf{u}^k)\}_{k \geq 1}$ is monotonically nonincreasing.

Lemma 3.5. *Suppose that [Assumption 3.1](#) and [Assumption 3.2](#) hold. Let the sequence $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ be generated by (3.1), and $\{\mathbf{u}^k\}$, $\{\mathbf{v}^k\}$ and $\{\Delta_{\mathbf{x}}^k\}$, $\{\Delta_{\mathbf{y}}^k\}$ are defined in (3.7) and (3.8), respectively. Then, for a given $\tau \in (\alpha, \Lambda(\gamma))$, the sequence $\{\Theta_{\alpha, \gamma}(\mathbf{u}^k)\}_{k \geq 1}$ is monotonically nonincreasing. In particular, for any $k \geq 1$, we have*

$$(3.11) \quad \Theta_{\alpha, \gamma}(\mathbf{u}^k) - \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) \geq (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \|\Delta_{\mathbf{y}}^{k+1}\|^2 + \xi(\alpha, \gamma) \|\Delta_{\mathbf{x}}^k\|^2,$$

where $\Lambda(\gamma)$ is defined in (3.2), and $\xi(\alpha, \gamma) := \frac{\alpha}{\gamma} + \alpha L_h - \frac{\alpha^2 L_h}{2} - \frac{\gamma \alpha^2 L_h^2}{2} - \frac{\alpha^2 L_h}{2\tau} - \frac{\alpha^2}{\tau\gamma} > 0$.

Proof. It follows from (3.9) that

$$(3.12) \quad \begin{aligned} \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^{k+1}, \mathbf{x}^k) - \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^{k+1}, \mathbf{x}^{k+1}) &= \frac{1}{\gamma} \langle -\Delta_{\mathbf{x}}^{k+1}, \mathbf{z}^{k+1} - \mathbf{y}^{k+1} \rangle \\ &= -\frac{1}{\gamma} \|\mathbf{y}^{k+1} - \mathbf{z}^{k+1}\|^2 - \frac{\alpha}{\gamma} \langle \mathbf{z}^{k+1} - \mathbf{y}^{k+1}, \Delta_{\mathbf{x}}^k \rangle, \end{aligned}$$

where the last equality follows from the first and last relations in (3.1). Since \mathbf{z}^{k+1} is a minimizer of the \mathbf{z} -subproblem according to the third equality in (3.1), we have

$$\begin{aligned} f_2(\mathbf{z}^k) + \frac{1}{2\gamma} \|2\mathbf{y}^{k+1} - \mathbf{z}^k - \mathbf{w}^k - \gamma \nabla h(\mathbf{y}^{k+1})\|^2 \\ \geq f_2(\mathbf{z}^{k+1}) + \frac{1}{2\gamma} \|2\mathbf{y}^{k+1} - \mathbf{z}^{k+1} - \mathbf{w}^k - \gamma \nabla h(\mathbf{y}^{k+1})\|^2. \end{aligned}$$

This together with (3.9), we have

$$\begin{aligned}
& \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^k, \mathbf{x}^k) - \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^{k+1}, \mathbf{x}^k) \\
&= f_2(\mathbf{z}^k) + \frac{1}{2\gamma} \left\| 2\mathbf{y}^{k+1} - \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^{k+1}) \right\|^2 - \frac{1}{\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{z}^k \right\|^2 \\
(3.13) \quad & - f_2(\mathbf{z}^{k+1}) - \frac{1}{2\gamma} \left\| 2\mathbf{y}^{k+1} - \mathbf{z}^{k+1} - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^{k+1}) \right\|^2 + \frac{1}{\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{z}^{k+1} \right\|^2 \\
&\geq \frac{1}{\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{z}^{k+1} \right\|^2 - \frac{1}{\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{z}^k \right\|^2 + \frac{1}{\gamma} \left\langle \mathbf{z}^{k+1} - \mathbf{z}^k, \mathbf{w}^k - \mathbf{x}^k \right\rangle \\
&= \frac{1}{\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{z}^{k+1} \right\|^2 - \frac{1}{\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{z}^k \right\|^2 + \frac{\alpha}{\gamma} \left\langle \mathbf{z}^{k+1} - \mathbf{z}^k, \Delta_{\mathbf{x}}^k \right\rangle,
\end{aligned}$$

where the last equality follows from the relation $\mathbf{w}^k = \mathbf{x}^k + \alpha \Delta_{\mathbf{x}}^k$ in (3.1). Since $f_1 + \frac{1}{2\gamma} \|\mathbf{w}^k - \cdot\|^2$ is a strongly convex function with modulus $\frac{1}{\gamma} - l$, and recall the optimality condition $0 = \nabla f_1(\mathbf{y}^{k+1}) + \frac{1}{\gamma}(\mathbf{y}^{k+1} - \mathbf{w}^k)$ for the \mathbf{y} -subproblem in (3.5), we obtain

$$f_1(\mathbf{y}^k) + \frac{1}{2\gamma} \left\| \mathbf{y}^k - \mathbf{w}^k \right\|^2 \geq f_1(\mathbf{y}^{k+1}) + \frac{1}{2\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{w}^k \right\|^2 + \frac{1}{2} \left(\frac{1}{\gamma} - l \right) \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2.$$

This implies that

$$\begin{aligned}
& f_1(\mathbf{y}^k) + \frac{1}{2\gamma} \left\| \mathbf{y}^k - \mathbf{x}^k \right\|^2 \\
&\geq f_1(\mathbf{y}^{k+1}) + \frac{1}{2\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{x}^k \right\|^2 - \frac{\alpha}{\gamma} \left\langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \right\rangle + \frac{1}{2} \left(\frac{1}{\gamma} - l \right) \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2.
\end{aligned}$$

Therefore, it follows from (3.9) that

$$\begin{aligned}
& \mathcal{H}_\gamma(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k) - \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^k, \mathbf{x}^k) \\
&= f_1(\mathbf{y}^k) + h(\mathbf{y}^k) + \frac{1}{2\gamma} \left\| \mathbf{y}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^k) \right\|^2 - \frac{1}{2\gamma} \left\| \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^k) \right\|^2 \\
&\quad - f_1(\mathbf{y}^{k+1}) - h(\mathbf{y}^{k+1}) - \frac{1}{2\gamma} \left\| \mathbf{y}^{k+1} - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^{k+1}) \right\|^2 + \frac{1}{2\gamma} \left\| \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^{k+1}) \right\|^2 \\
&\geq h(\mathbf{y}^k) - \left\langle \mathbf{y}^k - \mathbf{x}^k, \nabla h(\mathbf{y}^k) \right\rangle + \frac{\gamma}{2} \left\| \nabla h(\mathbf{y}^k) \right\|^2 - \frac{1}{2\gamma} \left\| \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^k) \right\|^2 \\
&\quad - h(\mathbf{y}^{k+1}) + \left\langle \mathbf{y}^{k+1} - \mathbf{x}^k, \nabla h(\mathbf{y}^{k+1}) \right\rangle - \frac{\gamma}{2} \left\| \nabla h(\mathbf{y}^{k+1}) \right\|^2 + \frac{1}{2\gamma} \left\| \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^{k+1}) \right\|^2 \\
&\quad + \frac{1}{2} \left(\frac{1}{\gamma} - l \right) \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2 - \frac{\alpha}{\gamma} \left\langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \right\rangle.
\end{aligned}$$

Then, expanding the squares and combining the terms in the right-hand side of the above

inequality, we have

$$\begin{aligned}
& \mathcal{H}_\gamma(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k) - \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^k, \mathbf{x}^k) \\
& \geq h(\mathbf{y}^k) + \langle \mathbf{z}^k - \mathbf{y}^k, \nabla h(\mathbf{y}^k) \rangle - \frac{1}{2\gamma} \|\mathbf{z}^k - \mathbf{x}^k\|^2 - h(\mathbf{y}^{k+1}) + \langle \mathbf{y}^{k+1} - \mathbf{z}^k, \nabla h(\mathbf{y}^{k+1}) \rangle \\
(3.14) \quad & + \frac{1}{2\gamma} \|\mathbf{z}^k - \mathbf{x}^k\|^2 + \frac{1}{2} \left(\frac{1}{\gamma} - l \right) \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \frac{\alpha}{\gamma} \langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \rangle \\
& = h(\mathbf{y}^k) + \langle \mathbf{z}^k - \mathbf{y}^k, \nabla h(\mathbf{y}^k) \rangle - h(\mathbf{y}^{k+1}) - \langle \mathbf{z}^k - \mathbf{y}^{k+1}, \nabla h(\mathbf{y}^{k+1}) \rangle \\
& + \frac{1}{2} \left(\frac{1}{\gamma} - l \right) \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \frac{\alpha}{\gamma} \langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \rangle,
\end{aligned}$$

Next, according to [Lemma 2.4](#) and the L_h -Lipschitz continuity of ∇h , we have

$$\begin{aligned}
& h(\mathbf{y}^k) - h(\mathbf{y}^{k+1}) + \langle \mathbf{z}^k - \mathbf{y}^k, \nabla h(\mathbf{y}^k) \rangle - \langle \mathbf{z}^k - \mathbf{y}^{k+1}, \nabla h(\mathbf{y}^{k+1}) \rangle \\
& = h(\mathbf{y}^k) - h(\mathbf{y}^{k+1}) + \langle \Delta_{\mathbf{y}}^{k+1}, \nabla h(\mathbf{y}^k) \rangle - \langle \mathbf{z}^k - \mathbf{y}^{k+1}, \nabla h(\mathbf{y}^{k+1}) - \nabla h(\mathbf{y}^k) \rangle \\
(3.15) \quad & \geq -\frac{L_h}{2} \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \langle \mathbf{z}^k - \mathbf{y}^{k+1}, \nabla h(\mathbf{y}^{k+1}) - \nabla h(\mathbf{y}^k) \rangle \\
& \geq -\frac{L_h}{2} \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \frac{L_h}{2} \|\mathbf{y}^{k+1} - \mathbf{z}^k\|^2 - \frac{L_h}{2} \|\Delta_{\mathbf{y}}^{k+1}\|^2.
\end{aligned}$$

Substituting [\(3.15\)](#) into [\(3.14\)](#), we obtain

$$\begin{aligned}
& \mathcal{H}_\gamma(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k) - \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^k, \mathbf{x}^k) \\
(3.16) \quad & \geq \frac{1}{2} \left(\frac{1}{\gamma} - l \right) \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \frac{\alpha}{\gamma} \langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \rangle - L_h \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \frac{L_h}{2} \|\mathbf{y}^{k+1} - \mathbf{z}^k\|^2.
\end{aligned}$$

Summing [\(3.12\)](#), [\(3.13\)](#) and [\(3.16\)](#) yields

$$\begin{aligned}
& \mathcal{H}_\gamma(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k) - \mathcal{H}_\gamma(\mathbf{y}^{k+1}, \mathbf{z}^{k+1}, \mathbf{x}^{k+1}) \\
& \geq \frac{1 - \gamma l - 2\gamma L_h}{2\gamma} \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \|\mathbf{y}^{k+1} - \mathbf{z}^k\|^2 + \frac{\alpha}{\gamma} \langle \mathbf{y}^k - \mathbf{z}^k, \Delta_{\mathbf{x}}^k \rangle \\
(3.17) \quad & = \frac{1 - \gamma l - 2\gamma L_h}{2\gamma} \|\Delta_{\mathbf{y}}^{k+1}\|^2 - \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \|\mathbf{y}^{k+1} - \mathbf{z}^k\|^2 \\
& - \frac{\alpha}{\gamma} \|\Delta_{\mathbf{x}}^k\|^2 + \frac{\alpha^2}{\gamma} \langle \Delta_{\mathbf{x}}^{k-1}, \Delta_{\mathbf{x}}^k \rangle,
\end{aligned}$$

where the last equality holds due to the fact $\mathbf{y}^k - \mathbf{z}^k = \mathbf{w}^{k-1} - \mathbf{x}^k = \mathbf{x}^{k-1} - \mathbf{x}^k + \alpha(\mathbf{x}^{k-1} - \mathbf{x}^{k-2})$ by [\(3.1\)](#). Our further aim is to analyze the negative term $\|\mathbf{y}^{k+1} - \mathbf{z}^k\|^2$. It follows from the

second equality in (3.1) that $0 = \nabla f_1(\mathbf{y}^{k+1}) + \frac{1}{\gamma}(\mathbf{y}^{k+1} - \mathbf{w}^k)$. Further, we obtain

$$\begin{aligned}
 (3.18) \quad & \|\mathbf{y}^{k+1} - \mathbf{z}^k\|^2 \\
 &= \left\| \mathbf{y}^{k+1} - \left(\mathbf{x}^k - \mathbf{w}^{k-1} + \mathbf{y}^k \right) \right\|^2 \\
 &= \left\| (\mathbf{y}^{k+1} - \mathbf{w}^k) - (\mathbf{y}^k - \mathbf{w}^{k-1}) + (\mathbf{w}^k - \mathbf{x}^k) \right\|^2 \\
 &= \gamma^2 \left\| \nabla f_1(\mathbf{y}^k) - \nabla f_1(\mathbf{y}^{k+1}) \right\|^2 + \|\mathbf{w}^k - \mathbf{x}^k\|^2 + 2\langle (\mathbf{y}^{k+1} - \mathbf{w}^k) - (\mathbf{y}^k - \mathbf{w}^{k-1}), \mathbf{w}^k - \mathbf{x}^k \rangle \\
 &\leq \gamma^2 L_{f_1}^2 \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2 + \alpha^2 \left\| \Delta_{\mathbf{x}}^k \right\|^2 + 2\alpha \langle (\mathbf{y}^{k+1} - \mathbf{w}^k) - (\mathbf{y}^k - \mathbf{w}^{k-1}), \Delta_{\mathbf{x}}^k \rangle \\
 &= \gamma^2 L_{f_1}^2 \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2 + 2\alpha \langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \rangle + 2\alpha^2 \langle \Delta_{\mathbf{x}}^{k-1}, \Delta_{\mathbf{x}}^k \rangle - \alpha(2 + \alpha) \left\| \Delta_{\mathbf{x}}^k \right\|^2,
 \end{aligned}$$

where the last equality follows from the relation $\mathbf{w}^{k-1} - \mathbf{w}^k = \alpha(\mathbf{x}^{k-1} - \mathbf{x}^{k-2}) + (1 + \alpha)(\mathbf{x}^{k-1} - \mathbf{x}^k)$ by (3.1). Substituting (3.18) into (3.17) yields

$$\begin{aligned}
 (3.19) \quad & \mathcal{H}_\gamma \left(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k \right) - \mathcal{H}_\gamma \left(\mathbf{y}^{k+1}, \mathbf{z}^{k+1}, \mathbf{x}^{k+1} \right) \\
 &\geq \left(\frac{1 - \gamma l - 2\gamma L_h}{2\gamma} - \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \gamma^2 L_{f_1}^2 \right) \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2 \\
 &\quad + \left(\frac{\alpha + \alpha^2}{\gamma} + \alpha L_h + \frac{\alpha^2 L_h}{2} \right) \left\| \Delta_{\mathbf{x}}^k \right\|^2 \\
 &\quad - \left(\frac{2}{\gamma} + L_h \right) \alpha \langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \rangle - \left(\frac{1}{\gamma} + L_h \right) \alpha^2 \langle \Delta_{\mathbf{x}}^{k-1}, \Delta_{\mathbf{x}}^k \rangle.
 \end{aligned}$$

Note that for any $\tau > 0$, it holds that

$$\alpha \langle \Delta_{\mathbf{y}}^{k+1}, \Delta_{\mathbf{x}}^k \rangle \leq \frac{\tau}{2} \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2 + \frac{\alpha^2}{2\tau} \left\| \Delta_{\mathbf{x}}^k \right\|^2,$$

and

$$\left(\frac{1}{\gamma} + L_h \right) \alpha^2 \langle \Delta_{\mathbf{x}}^{k-1}, \Delta_{\mathbf{x}}^k \rangle \leq \frac{\alpha^2}{2\gamma} \left\| \Delta_{\mathbf{x}}^{k-1} \right\|^2 + \frac{\gamma\alpha^2}{2} \left(\frac{1}{\gamma} + L_h \right)^2 \left\| \Delta_{\mathbf{x}}^k \right\|^2.$$

Substituting the above inequalities into (3.19), we get

$$\begin{aligned}
 (3.20) \quad & \mathcal{H}_\gamma \left(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k \right) - \mathcal{H}_\gamma \left(\mathbf{y}^{k+1}, \mathbf{z}^{k+1}, \mathbf{x}^{k+1} \right) \\
 &\geq (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \left\| \Delta_{\mathbf{y}}^{k+1} \right\|^2 - \frac{\alpha^2}{2\gamma} \left\| \Delta_{\mathbf{x}}^{k-1} \right\|^2 \\
 &\quad + \left(\frac{\alpha}{\gamma} + \frac{\alpha^2}{2\gamma} + \alpha L_h - \frac{\alpha^2 L_h}{2} - \frac{\gamma\alpha^2 L_h^2}{2} - \frac{\alpha^2 L_h}{2\tau} - \frac{\alpha^2}{\tau\gamma} \right) \left\| \Delta_{\mathbf{x}}^k \right\|^2,
 \end{aligned}$$

where $\Lambda(\gamma)$ is defined in (3.2) and τ is an auxiliary parameter assumed to satisfy $\alpha < \tau < \Lambda(\gamma)$, which must exist according to Assumption 3.2. Then, according to the definition of $\Theta_{\alpha,\gamma}$ in (3.10) and (3.20), the conclusion (3.11) can be obtained directly.

Now we show that $\xi(\alpha, \gamma) := \frac{\alpha}{\gamma} + \alpha L_h - \frac{\alpha^2 L_h}{2} - \frac{\gamma \alpha^2 L_h^2}{2} - \frac{\alpha^2 L_h}{2\tau} - \frac{\alpha^2}{\tau\gamma} > 0$. Since $\tau > \alpha$, we can easily obtain that $\frac{\alpha}{\gamma} - \frac{\alpha^2}{\tau\gamma} > 0$. It leaves to show $\alpha L_h - \frac{\alpha^2 L_h}{2} - \frac{\gamma \alpha^2 L_h^2}{2} - \frac{\alpha^2 L_h}{2\tau} > 0$. According to [Assumption 3.2](#), we know that

$$(3.21) \quad \alpha < \Lambda(\gamma) < \frac{1 - \gamma l - 2\gamma L_h}{2 + \gamma L_h} < \frac{1}{1 + \gamma L_h}.$$

This together with $\tau > \alpha$, we have $\alpha L_h - \frac{\alpha^2 L_h}{2} - \frac{\gamma \alpha^2 L_h^2}{2} - \frac{\alpha^2 L_h}{2\tau} > \alpha L_h - \frac{\alpha^2 L_h}{2} - \frac{\gamma \alpha^2 L_h^2}{2} - \frac{\alpha^2 L_h}{2\alpha} = \frac{\alpha L_h}{2}(1 - \alpha - \alpha\gamma L_h) > 0$. This completes the proof. \blacksquare

The following lemma presents that the sequences $\{\Delta_{\mathbf{x}}^k\}$, $\{\Delta_{\mathbf{y}}^k\}$ and $\{\mathbf{y}^k - \mathbf{z}^k\}$ vanish with certain sublinear convergence rate.

Lemma 3.6. *Suppose that [Assumption 3.1](#) and [Assumption 3.2](#) hold. Let the sequence $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ be generated by [\(3.1\)](#) which is assumed to be bounded, and the sequences $\{\mathbf{u}^k\}$, $\{\mathbf{v}^k\}$ are defined in [\(3.7\)](#), respectively. Then,*

- (i) *it holds that $\sum_{k=0}^{\infty} \|\Delta_{\mathbf{x}}^k\|^2 < +\infty$ and $\sum_{k=0}^{\infty} \|\Delta_{\mathbf{y}}^k\|^2 < +\infty$. Furthermore, we have $\lim_{k \rightarrow \infty} \|\Delta_{\mathbf{x}}^k\| = 0$, $\lim_{k \rightarrow \infty} \|\Delta_{\mathbf{y}}^k\| = 0$, and $\lim_{k \rightarrow \infty} \|\mathbf{y}^k - \mathbf{z}^k\| = 0$.*
- (ii) *it holds that $\min_{k \leq K} \|\Delta_{\mathbf{x}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, $\min_{k \leq K} \|\Delta_{\mathbf{y}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, and $\min_{k \leq K} \|\mathbf{y}^k - \mathbf{z}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$.*

Proof. We now prove (i). We first show that $\Theta_{\alpha, \gamma}(\mathbf{u}^k)$ is lower bounded for all k . It follows from the definition of $\Theta_{\alpha, \gamma}$ in [\(3.10\)](#) that

$$(3.22) \quad \begin{aligned} \Theta_{\alpha, \gamma}(\mathbf{u}^k) &= \mathcal{H}_{\gamma}(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k) + \frac{\alpha}{2\gamma} \|\Delta_{\mathbf{x}}^{k-1}\|^2 \\ &= f_1(\mathbf{y}^k) + f_2(\mathbf{z}^k) + h(\mathbf{y}^k) + \frac{1}{2\gamma} \|2\mathbf{y}^k - \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^k)\|^2 \\ &\quad - \frac{1}{2\gamma} \|\mathbf{x}^k - \mathbf{y}^k + \gamma \nabla h(\mathbf{y}^k)\|^2 - \frac{1}{\gamma} \|\mathbf{y}^k - \mathbf{z}^k\|^2 + \frac{\alpha}{2\gamma} \|\Delta_{\mathbf{x}}^{k-1}\|^2. \end{aligned}$$

Since ∇f_1 and ∇h are both Lipschitz continuous with moduli L_{f_1} and L_h , then

$$f_1(\mathbf{y}^k) \geq f_1(\mathbf{z}^k) - \langle \nabla f_1(\mathbf{y}^k), \mathbf{z}^k - \mathbf{y}^k \rangle - \frac{L_{f_1}}{2} \|\mathbf{y}^k - \mathbf{z}^k\|^2,$$

and

$$h(\mathbf{y}^k) \geq h(\mathbf{z}^k) - \langle \nabla h(\mathbf{y}^k), \mathbf{z}^k - \mathbf{y}^k \rangle - \frac{L_h}{2} \|\mathbf{y}^k - \mathbf{z}^k\|^2.$$

Substituting them into [\(3.22\)](#), and togethering with $\nabla f_1(\mathbf{y}^k) = -\frac{1}{\gamma}(\mathbf{y}^k - \mathbf{w}^{k-1})$ from [\(3.5\)](#), we have

$$(3.23) \quad \begin{aligned} \Theta_{\alpha, \gamma}(\mathbf{u}^k) &\geq f_1(\mathbf{z}^k) + f_2(\mathbf{z}^k) + h(\mathbf{z}^k) + \left(\frac{1}{2\gamma} - \frac{L_{f_1} + L_h}{2} \right) \|\mathbf{y}^k - \mathbf{z}^k\|^2 \\ &\quad - \frac{1}{\gamma} \langle \mathbf{x}^k - \mathbf{w}^{k-1}, \mathbf{y}^k - \mathbf{z}^k \rangle - \frac{1}{\gamma} \|\mathbf{y}^k - \mathbf{z}^k\|^2 + \frac{\alpha}{2\gamma} \|\Delta_{\mathbf{x}}^{k-1}\|^2 \\ &\geq F(\mathbf{z}^k) + \left(\frac{1}{2\gamma} - \frac{L_{f_1} + L_h}{2} \right) \|\mathbf{y}^k - \mathbf{z}^k\|^2, \end{aligned}$$

where the first inequality follows from $\frac{1}{2\gamma} \|2\mathbf{y}^k - \mathbf{z}^k - \mathbf{x}^k - \gamma \nabla h(\mathbf{y}^k)\|^2 = \frac{1}{2\gamma} \|\mathbf{y}^k - \mathbf{z}^k\|^2 + \frac{1}{\gamma} \langle \mathbf{y}^k - \mathbf{x}^k, \mathbf{y}^k - \mathbf{z}^k \rangle - \langle \mathbf{y}^k - \mathbf{z}^k, \nabla h(\mathbf{y}^k) \rangle + \frac{1}{2\gamma} \|\mathbf{x}^k - \mathbf{y}^k + \gamma \nabla h(\mathbf{y}^k)\|^2$, and the second one follows from $\frac{\alpha}{2\gamma} \geq 0$ and $\mathbf{x}^k = \mathbf{w}^{k-1} + (\mathbf{z}^k - \mathbf{y}^k)$ by (3.1). This implies that $\Theta_{\alpha,\gamma}(\mathbf{u}^k)$ for all $k \geq 1$ is bounded from below due to the fact that $0 < \gamma < \frac{1}{L_{f_1} + L_h}$ and the boundedness of F and $\{\mathbf{u}^k\}_{k \geq 1}$. Summing (3.11) from $k = 1$ to $N - 1 \geq 0$, we get

$$(3.24) \quad \Theta_{\alpha,\gamma}(\mathbf{u}^1) - \Theta_{\alpha,\gamma}(\mathbf{u}^N) \geq (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \sum_{k=2}^N \|\Delta_{\mathbf{y}}^k\|^2 + \xi(\alpha, \gamma) \sum_{k=1}^{N-1} \|\Delta_{\mathbf{x}}^k\|^2.$$

Therefore, letting $N \rightarrow +\infty$ and following the lower boundedness of $\{\Theta_{\alpha,\gamma}(\mathbf{u}^k)\}_{k \geq 1}$, we have

$$(3.25) \quad (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \sum_{k=2}^{\infty} \|\Delta_{\mathbf{y}}^k\|^2 + \xi(\alpha, \gamma) \sum_{k=1}^{\infty} \|\Delta_{\mathbf{x}}^k\|^2 < +\infty.$$

This implies that $\sum_{k=0}^{\infty} \|\Delta_{\mathbf{x}}^k\|^2 < +\infty$ and $\sum_{k=0}^{\infty} \|\Delta_{\mathbf{y}}^k\|^2 < +\infty$. Therefore, it holds that $\lim_{k \rightarrow \infty} \|\Delta_{\mathbf{x}}^k\| = 0$ and $\lim_{k \rightarrow \infty} \|\Delta_{\mathbf{y}}^k\| = 0$. Since $\mathbf{y}^k - \mathbf{z}^k = \mathbf{w}^{k-1} - \mathbf{x}^k = \mathbf{x}^{k-1} - \mathbf{x}^k + \alpha(\mathbf{x}^{k-1} - \mathbf{x}^{k-2})$ from (3.1), we further have $\lim_{k \rightarrow \infty} \|\mathbf{y}^k - \mathbf{z}^k\| = 0$.

We turn to prove (ii). According to (3.24) and recalling $\xi(\alpha, \gamma) > 0$ and the lower boundedness of $\{\Theta_{\alpha,\gamma}(\mathbf{u}^k)\}_{k \geq 1}$, we know that there exists a constant C_0 such that

$$(3.26) \quad K \cdot \min_{1 \leq k \leq K} \|\Delta_{\mathbf{x}}^k\|^2 \leq \sum_{k=1}^K \|\Delta_{\mathbf{x}}^k\|^2 \leq \frac{1}{\xi(\alpha, \gamma)} (\Theta_{\alpha,\gamma}(\mathbf{u}^1) - \Theta_{\alpha,\gamma}(\mathbf{u}^{K+1})) \leq C_0.$$

This implies that $\min_{k \leq K} \|\Delta_{\mathbf{x}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$. Similarly, we can obtain $\min_{k \leq K} \|\Delta_{\mathbf{y}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$ and $\min_{k \leq K} \|\mathbf{y}^k - \mathbf{z}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$. This completes the proof. \blacksquare

Note that in Lemma 3.6, we show the lower boundedness of $\Theta_{\alpha,\gamma}$, as well as \mathcal{H}_{γ} , for the generated sequences, relying on Assumption 3.1(iii) and Assumption 3.2. The lower boundedness plays a crucial role in establishing both the sublinear convergence rate and the convergence of the generated sequence. Some similar results have also been discussed in [42], which demonstrates the consistency between the lower bound and the minimizer of \mathcal{H}_{γ} and F .

In the following, we give the subsequential convergence result for Algorithm 3.1.

Theorem 3.7. *Suppose that Assumption 3.1 and Assumption 3.2 hold. Let the sequence $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ be generated by (3.1) which is assumed to be bounded, and the sequences $\{\mathbf{u}^k\}$, $\{\mathbf{v}^k\}$ are defined in (3.7). Then,*

- (i) *any cluster point $\mathbf{u}^* := (\mathbf{y}^*, \mathbf{z}^*, \mathbf{x}^*, \mathbf{x}^*, \mathbf{x}^*)$ of the sequence $\{\mathbf{u}^k\}_{k \geq 1}$ is a critical point of the problem (1.1), i.e., it holds that $0 \in \partial F(\mathbf{y}^*)$.*
- (ii) *The limit $\lim_{k \rightarrow \infty} \Theta_{\alpha,\gamma}(\mathbf{u}^k)$ exists and for any cluster point \mathbf{u}^* of the sequence $\{\mathbf{u}^k\}_{k \geq 1}$, we have*

$$(3.27) \quad \Theta^* := \lim_{k \rightarrow \infty} \Theta_{\alpha,\gamma}(\mathbf{u}^k) = \Theta_{\alpha,\gamma}(\mathbf{u}^*).$$

Proof. We first prove (i). It follows from (3.1) and Lemma 3.6(i) that

$$\lim_{k \rightarrow \infty} \|\mathbf{z}^{k+1} - \mathbf{z}^k\| = 0.$$

Let \mathbf{u}^* be a cluster point of $\{\mathbf{u}^k\}_{k \geq 1}$, and assume that $\{\mathbf{u}^{k_j}\}$ is a convergent subsequence such that $\lim_{k \rightarrow \infty} \mathbf{u}^{k_j} = \mathbf{u}^*$. Then

$$(3.28) \quad \lim_{j \rightarrow \infty} \mathbf{u}^{k_j} = \lim_{j \rightarrow \infty} \mathbf{u}^{k_j-1} = \mathbf{u}^*.$$

Summing (3.5) and (3.6) and taking the limit along the convergent subsequence $\{\mathbf{u}^{k_j}\}$, and applying (2.1) and (3.28), we have

$$0 \in \nabla f_1(\mathbf{y}^*) + \partial f_2(\mathbf{y}^*) + \nabla h(\mathbf{y}^*).$$

Now we prove (ii). Suppose that $\{\mathbf{u}^{k_j}\}$ is a subsequence which converges to \mathbf{u}^* as $j \rightarrow \infty$. It follows from Lemma 3.5 and Lemma 3.6 that $\Theta_{\alpha, \gamma}$ is nonincreasing and bounded from below by Assumption 3.2. Therefore, $\Theta^* := \lim_{k \rightarrow \infty} \Theta_{\alpha, \gamma}(\mathbf{u}^k)$ exists. It follows from (3.1) that \mathbf{z}^k is the minimizer of \mathbf{z} -subproblem, we have

$$\begin{aligned} & f_2(\mathbf{z}^k) + \frac{1}{2\gamma} \left\| \mathbf{z}^k - \left(2\mathbf{y}^k - \gamma \nabla h(\mathbf{y}^k) - \mathbf{x}^{k-1} \right) \right\|^2 \\ & \leq f_2(\mathbf{z}^*) + \frac{1}{2\gamma} \left\| \mathbf{z}^* - \left(2\mathbf{y}^k - \gamma \nabla h(\mathbf{y}^k) - \mathbf{x}^{k-1} \right) \right\|^2. \end{aligned}$$

Replacing k by k_j in the above inequality and taking the limit on both sides, it follows from (3.28) yields $\lim_{j \rightarrow \infty} f_2(\mathbf{z}^{k_j}) \leq f_2(\mathbf{z}^*)$. On the other hand, since f_2 is proper and closed, we have $\liminf_{j \rightarrow \infty} f_2(\mathbf{z}^{k_j}) \geq f_2(\mathbf{z}^*)$. Hence

$$\lim_{j \rightarrow \infty} f_2(\mathbf{z}^{k_j}) = f_2(\mathbf{z}^*).$$

This together with the properties of f_1 and g in Assumption 3.1 and (3.28), and the boundedness of the sequence $\{\mathbf{u}^k\}$, we claim that

$$\Theta^* := \lim_{k \rightarrow \infty} \Theta_{\alpha, \gamma}(\mathbf{u}^k) = \Theta_{\alpha, \gamma}(\mathbf{u}^*).$$

This completes the proof. ■

Remark 3.8. Note that the boundedness of the sequence $\{\mathbf{x}^k\}$ is a standard assumption for the nonconvex optimization algorithms. It is documented in [2, Remark 3.3] that the boundedness assumption on the sequence $\{\mathbf{x}^k\}$ automatically holds when the corresponding lower level set $\{\mathbf{x} \mid F(\mathbf{x}) \leq F_0\}$ is compact for some $F_0 \in \mathbb{R}$.

We present an inequality characterizing the upper bound of the subdifferential of $\Theta_{\alpha, \gamma}$, which plays a key role in further convergence analysis.

Lemma 3.9. Suppose that [Assumption 3.1](#) and [Assumption 3.2](#) hold. Let h be a twice continuously differentiable function with a bounded Hessian, i.e., there exists a constant $M > 0$ such that $\|\nabla^2 h(\mathbf{y})\| \leq M, \forall \mathbf{y} \in \mathbb{R}^n$. Let $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ be the sequence generated by [\(3.1\)](#), and $\{\mathbf{u}^k\}$, $\{\mathbf{v}^k\}$ are defined in [\(3.7\)](#). Then, for any $k \geq 1$, there exists a constant $b > 0$ such that

$$(3.29) \quad \text{dist} \left(0, \partial \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) \right) \leq b \left(\|\Delta_{\mathbf{x}}^{k+1}\| + \|\Delta_{\mathbf{x}}^k\| \right).$$

Proof. Firstly, from the definition of $\Theta_{\alpha, \gamma}$ in [\(3.10\)](#), we have

$$(3.30) \quad \begin{aligned} \nabla_{\mathbf{y}} \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) &= \nabla f_1(\mathbf{y}^{k+1}) + \frac{1}{\gamma} \left(\mathbf{y}^{k+1} - \mathbf{x}^{k+1} \right) + \nabla^2 h(\mathbf{y}^{k+1})^{\top} \left(\mathbf{z}^{k+1} - \mathbf{y}^{k+1} \right) \\ &= \frac{1}{\gamma} \left(\mathbf{x}^k - \mathbf{x}^{k+1} \right) + \frac{\alpha}{\gamma} \left(\mathbf{x}^k - \mathbf{x}^{k-1} \right) + \nabla^2 h(\mathbf{y}^{k+1})^{\top} \left(\mathbf{z}^{k+1} - \mathbf{y}^{k+1} \right), \end{aligned}$$

where the last equality follows from [\(3.5\)](#). Secondly, we compute the subgradient of $\Theta_{\alpha, \gamma}$ with respect to \mathbf{z} as follows:

$$(3.31) \quad \begin{aligned} \partial_{\mathbf{z}} \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) &= \partial f_2(\mathbf{z}^{k+1}) + \frac{1}{\gamma} \left(\mathbf{z}^{k+1} - 2\mathbf{y}^{k+1} + \gamma \nabla h(\mathbf{y}^{k+1}) + \mathbf{x}^{k+1} \right) - \frac{2}{\gamma} \left(\mathbf{z}^{k+1} - \mathbf{y}^{k+1} \right) \\ &\ni -\frac{1}{\gamma} \left(\mathbf{x}^{k+1} - \mathbf{x}^k \right) - \frac{\alpha}{\gamma} \left(\mathbf{x}^k - \mathbf{x}^{k-1} \right), \end{aligned}$$

where the inclusion follows from [\(3.1\)](#) and [\(3.6\)](#). Thirdly, from the definition of $\Theta_{\alpha, \gamma}$ in [\(3.10\)](#), it is easy to obtain

$$(3.32) \quad \nabla_{\mathbf{x}} \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) = \frac{1}{\gamma} \left(\mathbf{z}^{k+1} - \mathbf{y}^{k+1} \right) = \frac{1}{\gamma} \left(\mathbf{x}^{k+1} - \mathbf{x}^k \right) + \frac{\alpha}{\gamma} \left(\mathbf{x}^k - \mathbf{x}^{k-1} \right),$$

where the last equality follows from [\(3.1\)](#). Finally, it follows from [\(3.10\)](#) that

$$(3.33) \quad \nabla_{\mathbf{x}_1} \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) = \frac{\alpha^2}{\gamma} \left(\mathbf{x}^k - \mathbf{x}^{k-1} \right) \quad \text{and} \quad \nabla_{\mathbf{x}_2} \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) = \frac{\alpha^2}{\gamma} \left(\mathbf{x}^{k-1} - \mathbf{x}^k \right).$$

Besides, by the boundedness of $\|\nabla^2 h(\cdot)\|$, we get

$$(3.34) \quad \begin{aligned} &\left\| \nabla_{\mathbf{y}} \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) \right\| \\ &\leq \frac{1}{\gamma} \left\| \mathbf{x}^k - \mathbf{x}^{k+1} \right\| + \frac{\alpha}{\gamma} \left\| \mathbf{x}^k - \mathbf{x}^{k-1} \right\| + M \left\| \mathbf{z}^{k+1} - \mathbf{y}^{k+1} \right\| \\ &\leq \left(\frac{1}{\gamma} + M \right) \left\| \mathbf{x}^k - \mathbf{x}^{k+1} \right\| + \left(\frac{\alpha}{\gamma} + \alpha M \right) \left\| \mathbf{x}^k - \mathbf{x}^{k-1} \right\|, \end{aligned}$$

where the last inequality follows from the first and last relations in [\(3.1\)](#). Combining [\(3.30\)](#), [\(3.31\)](#), [\(3.32\)](#), [\(3.33\)](#), and [\(3.34\)](#), we can obtain the conclusion [\(3.29\)](#) immediately. This completes the proof. \blacksquare

Now we establish the global convergence for [Algorithm 3.1](#) based on the uniformized KL property. We will show that the sequence $\{\mathbf{u}^k\}_{k \geq 1}$ has finite length and thus is convergent. Especially, the sequence $\{\mathbf{y}^k\}_{k \geq 1}$ converges to a stationary point in $\text{crit}F$.

Theorem 3.10. *Suppose that [Assumption 3.1](#) and [Assumption 3.2](#) hold. Let h be a twice continuously differentiable function with a bounded Hessian, i.e., there exists a constant $M > 0$ such that $\|\nabla^2 h(\mathbf{y})\| \leq M, \forall \mathbf{y} \in \mathbb{R}^n$. Let $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ be the sequence generated by [\(3.1\)](#) which is assumed to be bounded, and $\{\mathbf{u}^k\}$, $\{\mathbf{v}^k\}$ are defined in [\(3.7\)](#). If F in [\(1.1\)](#) is a KL function, then the sequence $\{\mathbf{u}^k\}_{k \geq 1}$ has finite length, that is,*

$$\sum_{k=1}^{\infty} \|\mathbf{y}^{k+1} - \mathbf{y}^k\| < +\infty, \quad \sum_{k=1}^{\infty} \|\mathbf{z}^{k+1} - \mathbf{z}^k\| < +\infty, \quad \sum_{k=1}^{\infty} \|\mathbf{x}^{k+1} - \mathbf{x}^k\| < +\infty.$$

Hence, the whole sequence $\{\mathbf{u}^k\}_{k \geq 1}$ is convergent.

Proof. We use $\theta(\mathbf{u}^\infty)$ to denote the cluster point set of the sequence $\{\mathbf{u}^k\}$. Since $\{\mathbf{u}^k\}$ is bounded, $\theta(\mathbf{u}^\infty)$ is a nonempty compact set, and it holds that

$$\lim_{k \rightarrow \infty} \text{dist}(\mathbf{u}^k, \theta(\mathbf{u}^\infty)) = 0.$$

From [Lemma 3.6\(i\)](#), [Theorem 3.7\(i\)](#) and [\(3.1\)](#), we know that $\theta(\mathbf{u}^\infty) \subseteq \text{crit}F \times \text{crit}F \times \text{crit}F \times \text{crit}F \times \text{crit}F$. Hence, for any $\mathbf{u}^* := (\mathbf{y}^*, \mathbf{z}^*, \mathbf{x}^*, \mathbf{x}^*, \mathbf{x}^*) \in \theta(\mathbf{u}^\infty)$, there exists a subsequence $\{\mathbf{u}^{k_i}\}$ of $\{\mathbf{u}^k\}$ converging to \mathbf{u}^* .

It follows from [Theorem 3.7\(ii\)](#) that $\lim_{k \rightarrow \infty} \Theta_{\alpha, \gamma}(\mathbf{u}^k) = \Theta_{\alpha, \gamma}(\mathbf{u}^*)$. If there exists an integer \bar{k} such that $\Theta_{\alpha, \gamma}(\mathbf{u}^k) = \Theta_{\alpha, \gamma}(\mathbf{u}^*)$, then from [Lemma 3.5](#), we have

$$\begin{aligned} & (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \|\Delta_{\mathbf{y}}^{k+1}\|^2 + \xi(\alpha, \gamma) \|\Delta_{\mathbf{x}}^k\|^2 \\ & \leq \Theta_{\alpha, \gamma}(\mathbf{u}^k) - \Theta_{\alpha, \gamma}(\mathbf{u}^{k+1}) \\ & \leq \Theta_{\alpha, \gamma}(\mathbf{u}^{\bar{k}}) - \Theta_{\alpha, \gamma}(\mathbf{u}^*) \\ & = 0 \quad \forall k > \bar{k}. \end{aligned}$$

Thus, we have $\mathbf{y}^{k+1} = \mathbf{y}^k$ and $\mathbf{x}^{k+1} = \mathbf{x}^k$ for any $k > \bar{k}$. Together with [\(3.1\)](#), we also have $\mathbf{z}^{k+1} = \mathbf{z}^k$, and thus the assertion $\sum_{k=1}^{\infty} \|\mathbf{x}^{k+1} - \mathbf{x}^k\| < +\infty$, $\sum_{k=1}^{\infty} \|\mathbf{y}^{k+1} - \mathbf{y}^k\| < +\infty$, and $\sum_{k=1}^{\infty} \|\mathbf{z}^{k+1} - \mathbf{z}^k\| < +\infty$ hold trivially. Otherwise, since $\Theta_{\alpha, \gamma}(\mathbf{u}^k)$ is nonincreasing from [Lemma 3.5](#), we have $\Theta_{\alpha, \gamma}(\mathbf{u}^k) > \Theta_{\alpha, \gamma}(\mathbf{u}^*)$ for all k . Again from $\lim_{k \rightarrow \infty} \Theta_{\alpha, \gamma}(\mathbf{u}^k) = \Theta_{\alpha, \gamma}(\mathbf{u}^*)$, we know that for any $\eta > 0$, there exists a nonnegative integer k_0 such that $\Theta_{\alpha, \gamma}(\mathbf{u}^k) < \Theta_{\alpha, \gamma}(\mathbf{u}^*) + \eta$ for any $k > k_0$. In addition, for any $\varsigma > 0$ there exists a positive integer k_1 such that $\text{dist}(\mathbf{u}^k, \theta(\mathbf{u}^\infty)) < \varsigma$ for all $k > k_1$. Consequently, for any $\eta, \varsigma > 0$, when $k > k_2 := \max\{k_0, k_1\}$, we have

$$\text{dist}(\mathbf{u}^k, \theta(\mathbf{u}^\infty)) < \varsigma \quad \text{and} \quad \Theta_{\alpha, \gamma}(\mathbf{u}^k) < \Theta_{\alpha, \gamma}(\mathbf{u}^*) + \eta.$$

Since $\theta(\mathbf{u}^\infty)$ is a nonempty and compact set, and $\Theta_{\alpha, \gamma}$ is a constant on $\theta(\mathbf{u}^\infty)$, we can apply [Lemma 2.3](#) with $\Omega := \theta(\mathbf{u}^\infty)$. Therefore, for any $k > k_2$, we have

$$(3.35) \quad \varphi'(\Theta_{\alpha, \gamma}(\mathbf{u}^k) - \Theta_{\alpha, \gamma}(\mathbf{u}^*)) \text{dist}(0, \partial\Theta_{\alpha, \gamma}(\mathbf{u}^k)) \geq 1.$$

From the concavity of φ , we have

$$\begin{aligned} & \varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)) - \varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^{k+1}) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)) \\ & \geq \varphi'(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^*))(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^{k+1})). \end{aligned}$$

Then, associated with $\text{dist}(0, \partial\Theta_{\alpha,\gamma}(\mathbf{u}^k)) \leq b(\|\mathbf{x}^k - \mathbf{x}^{k-1}\| + \|\mathbf{x}^{k-1} - \mathbf{x}^{k-2}\|)$ in [Lemma 3.9](#), [\(3.35\)](#), and $\varphi'(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)) > 0$, we get

$$\begin{aligned} \Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^{k+1}) & \leq \frac{\varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)) - \varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^{k+1}) - \Theta_{\alpha,\gamma}(\mathbf{u}^*))}{\varphi'(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^*))} \\ & \leq b(\|\mathbf{x}^k - \mathbf{x}^{k-1}\| + \|\mathbf{x}^{k-1} - \mathbf{x}^{k-2}\|) \\ & \quad \times [\varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)) - \varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^{k+1}) - \Theta_{\alpha,\gamma}(\mathbf{u}^*))]. \end{aligned}$$

For convenience, for all $p, q \in \mathbb{N}$, we define

$$\zeta_{p,q} := \varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^p) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)) - \varphi(\Theta_{\alpha,\gamma}(\mathbf{u}^q) - \Theta_{\alpha,\gamma}(\mathbf{u}^*)).$$

Combining [\(3.11\)](#) and the above relation, it yields that for any $k > k_2$,

$$\begin{aligned} & (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right) \|\Delta_{\mathbf{y}}^{k+1}\|^2 + \xi(\alpha, \gamma) \|\Delta_{\mathbf{x}}^k\|^2 \\ & \leq \Theta_{\alpha,\gamma}(\mathbf{u}^k) - \Theta_{\alpha,\gamma}(\mathbf{u}^{k+1}) \\ & \leq b \left(\|\Delta_{\mathbf{x}}^k\| + \|\Delta_{\mathbf{x}}^{k-1}\| \right) \zeta_{k,k+1}. \end{aligned}$$

This implies that

$$\|\Delta_{\mathbf{y}}^{k+1}\| \leq \sqrt{\frac{1}{2}(\|\Delta_{\mathbf{x}}^k\| + \|\Delta_{\mathbf{x}}^{k-1}\|)} \sqrt{\frac{2b}{\rho_1} \zeta_{k,k+1}},$$

and

$$\|\Delta_{\mathbf{x}}^k\| \leq \sqrt{\frac{1}{2}(\|\Delta_{\mathbf{x}}^k\| + \|\Delta_{\mathbf{x}}^{k-1}\|)} \sqrt{\frac{2b}{\rho_2} \zeta_{k,k+1}},$$

where $\rho_1 := (\Lambda(\gamma) - \tau) \left(\frac{1}{\gamma} + \frac{L_h}{2} \right)$ and $\rho_2 := \xi(\alpha, \gamma)$. Further, using the fact that $\sqrt{\mu_1 \mu_2} \leq \mu_1/2 + \mu_2/2$ with $\mu_1 = (\|\Delta_{\mathbf{x}}^k\| + \|\Delta_{\mathbf{x}}^{k-1}\|)/2$ and $\mu_2 = 2b\zeta_{k,k+1}/\rho_1$ or $\mu_2 = 2b\zeta_{k,k+1}/\rho_2$, we get

$$(3.36) \quad \|\Delta_{\mathbf{y}}^{k+1}\| \leq \frac{1}{4} \left(\|\Delta_{\mathbf{x}}^k\| + \|\Delta_{\mathbf{x}}^{k-1}\| \right) + \frac{b}{\rho_1} \zeta_{k,k+1},$$

and

$$(3.37) \quad \|\Delta_{\mathbf{x}}^k\| \leq \frac{1}{4} \left(\|\Delta_{\mathbf{x}}^k\| + \|\Delta_{\mathbf{x}}^{k-1}\| \right) + \frac{b}{\rho_2} \zeta_{k,k+1}.$$

Then, it follows from [Lemma 2.5](#) and [\(3.37\)](#) that $\sum_{k=1}^{\infty} \|\Delta_{\mathbf{x}}^{k+1}\| < +\infty$, and we further have $\sum_{k=1}^{\infty} \|\Delta_{\mathbf{y}}^{k+1}\| < +\infty$ due to [\(3.36\)](#). Again from [\(3.1\)](#), we know that $\sum_{k=1}^{\infty} \|\Delta_{\mathbf{z}}^{k+1}\| < +\infty$. Thus, $\{\mathbf{u}^k\}_{k \geq 1}$ is a Cauchy sequence and hence it is convergent. Applying [Theorem 3.7\(i\)](#), there exists a $\mathbf{y}^* \in \text{crit}F$ such that $\lim_{k \rightarrow \infty} \mathbf{y}^k = \mathbf{y}^*$. This completes the proof. \blacksquare

Remark 3.11. KL functions exhibit remarkable versatility and are extensively applied in various domains, including semi-algebraic analysis, subanalytic analysis, and log-exp functions. Concrete examples of KL functions can be found in [2, 3, 8]. These examples encompass many common instances such as ℓ_p -norm (where $p \geq 0$), indicator functions of semi-algebraic sets, and a majority of convex functions.

4. Extrapolated PnP-DYS methods. In this section, we focus on the development of a class of Plug-and-Play Davis-Yin splitting (PnP-DYS) algorithms with convergence guarantee. The PnP approach is a versatile methodology primarily utilized for addressing inverse problems involving large-scale measurements through the integration of statistical priors defined as denoisers. This approach draws inspiration from well-established proximal algorithms commonly employed in nonsmooth composite optimization, such as FBS, DRS, and ADMM. The rise in the popularity of deep learning has resulted in the widespread adoption of PnP for effectively utilizing learned priors defined through pre-trained deep neural networks. This adoption has propelled PnP to achieve state-of-the-art performance across a range of applications. For instance, by replacing the proximal operator of f_2 with a learned denoiser \mathcal{D}_σ in (1.2), we can obtain a PnP-DYS method as follows:

$$\begin{cases} \mathbf{y}^{k+1} = \text{Prox}_{\gamma f_1}(\mathbf{x}^k), \\ \mathbf{z}^{k+1} = \mathcal{D}_\sigma(2\mathbf{y}^{k+1} - \gamma \nabla h(\mathbf{y}^{k+1}) - \mathbf{x}^k), \\ \mathbf{x}^{k+1} = \mathbf{x}^k + (\mathbf{z}^{k+1} - \mathbf{y}^{k+1}). \end{cases}$$

To guarantee the theoretical convergence, we consider the Gradient Step (GS) Denoiser developed in [13, 27] as follows:

$$(4.1) \quad \mathcal{D}_\sigma = I - \nabla g_\sigma,$$

which is obtained from a scalar function:

$$g_\sigma = \frac{1}{2} \|\mathbf{x} - N_\sigma(\mathbf{x})\|^2,$$

where the mapping $N_\sigma(\mathbf{x})$ is realized as a differentiable neural network, enabling the explicit computation of g_σ and ensuring that g_σ has a Lipschitz gradient with a constant L ($L < 1$). Originally, the denoiser \mathcal{D}_σ in (4.1) is trained to denoise images degraded with Gaussian noise of level σ . In [27], it is shown that, although constrained to be an exact conservative field, it can realize state-of-the-art denoising. Remarkably, the denoiser \mathcal{D}_σ in (4.1) takes the form of a proximal mapping of a weakly convex function, as stated in the next proposition.

Proposition 4.1. [28, Proposition 3.1] $\mathcal{D}_\sigma(\mathbf{x}) = \text{prox}_{\phi_\sigma}(\mathbf{x})$, where ϕ_σ is defined by

$$(4.2) \quad \phi_\sigma(\mathbf{x}) = g_\sigma(\mathcal{D}_\sigma^{-1}(\mathbf{x})) - \frac{1}{2} \|\mathcal{D}_\sigma^{-1}(\mathbf{x}) - \mathbf{x}\|^2$$

if $\mathbf{x} \in \text{Im}(\mathcal{D}_\sigma)$, and $\phi_\sigma(\mathbf{x}) = +\infty$ otherwise. Moreover, ϕ_σ is $\frac{L}{L+1}$ -weakly convex and $\nabla \phi_\sigma$ is $\frac{L}{1-L}$ -Lipschitz on $\text{Im}(\mathcal{D}_\sigma)$, and $\phi_\sigma(\mathbf{x}) \geq g_\sigma(\mathbf{x})$, $\forall \mathbf{x} \in \mathbb{R}^n$.

Drawing upon the [Proposition 4.1](#), we are interested in developing the extrapolated PnP-DYS algorithm, with a plugged denoiser \mathcal{D}_σ in [\(4.1\)](#) that corresponds to the proximal operator of a nonconvex functional ϕ_σ in [\(4.2\)](#). To do so, we turn to target the optimization problems as follows:

$$(4.3) \quad \min F_{\gamma, \sigma}(\mathbf{x}) = f(\mathbf{x}) + \frac{1}{\gamma} \phi_\sigma(\mathbf{x}) + h(\mathbf{x}),$$

where f is a (possibly nonconvex) data-fidelity term, h is differential with Lipschitz continuous gradient, γ is a regularization parameter and ϕ_σ is defined as in [Proposition 4.1](#) from the function g_σ satisfying $\mathcal{D}_\sigma = I - \nabla g_\sigma$. In our analysis, to use [Proposition 4.1](#), g_σ is assumed \mathcal{C}^2 with L -Lipschitz continuous gradient ($L < 1$). We also assume f and g_σ bounded from below. From [Proposition 4.1](#), we get that ϕ_σ and thus $F_{\lambda, \sigma}$ are also bounded from below. In the following, we develop two extrapolated PnP-DYS methods depending on whether f in [\(4.3\)](#) exhibits smoothness and discuss their theoretical convergence.

According to [\[25, Lemma 1\]](#), $\phi_\sigma(\mathbf{x})$ in [\(4.2\)](#) satisfies the Kurdyka-Łojasiewicz (KL) property if g_σ is real analytic [\[31\]](#) in a neighborhood of $\mathbf{x} \in \mathbb{R}^n$ and its Jacobian matrix $Jg_\sigma(\mathbf{x})$ is nonsingular. Note that the real analytic property of g_σ can be ensured for a broader range of deep neural networks. Meanwhile, the nonsingularity of $Jg_\sigma(\mathbf{x})$ can be guaranteed by assuming $L < 1$ as discussed in [\[25\]](#). For more discussions on general conditions under which the KL property holds for deep neural networks, we refer to [\[5, 11, 77\]](#). Therefore, selecting a neural network for g_σ that guarantees the KL property of $\phi_\sigma(\mathbf{x})$ during implementation is not a difficult task.

4.1. When f is smooth with Lipschitz continuous gradient. In this subsection, we consider the case that f in [\(4.3\)](#) is differentiable with Lipschitz continuous gradient. In this case, we replace the second proximal subproblem with a learned denoiser \mathcal{D}_σ in [\(4.1\)](#), and produce a smooth extrapolated PnP-DYS method detailed in [Algorithm 4.1](#). Actually, [Algorithm 4.1](#) reduces to the extrapolated versions, i.e., the accelerated versions, of PnP-DRS and PnP-FBS methods when $h = 0$ and $f = 0$, respectively. Notably, these specific cases have not been explored in previous literature to the best of our knowledge.

Algorithm 4.1 A smooth extrapolated PnP-DYS method

Choose the parameters $\alpha \geq 0$ and $\gamma > 0$. Given \mathbf{x}^0 and $\mathbf{x}^{-1} = \mathbf{x}^0$, set $k = 0$.

while the stopping criteria is not satisfied, **do**

$$\begin{cases} \mathbf{w}^k = \mathbf{x}^k + \alpha(\mathbf{x}^k - \mathbf{x}^{k-1}), \\ \mathbf{y}^{k+1} = \text{Prox}_{\gamma f}(\mathbf{w}^k), \\ \mathbf{z}^{k+1} = \mathcal{D}_\sigma(2\mathbf{y}^{k+1} - \gamma \nabla h(\mathbf{y}^{k+1}) - \mathbf{w}^k), \\ \mathbf{x}^{k+1} = \mathbf{w}^k + (\mathbf{z}^{k+1} - \mathbf{y}^{k+1}). \end{cases}$$

end while

Next, we discuss the convergence property of [Algorithm 4.1](#) for the explicit optimization

problem (4.3). Before the analysis, we define

$$\tilde{\mathcal{H}}_\gamma(\mathbf{y}, \mathbf{z}, \mathbf{x}) = f(\mathbf{y}) + \frac{1}{\gamma}\phi_\sigma(\mathbf{z}) + h(\mathbf{y}) + \frac{1}{2\gamma}\|\mathbf{y} - \mathbf{x} - \gamma\nabla h(\mathbf{y})\|^2 - \frac{1}{2\gamma}\|\mathbf{z} - \mathbf{x} - \gamma\nabla h(\mathbf{y})\|^2,$$

and

$$\tilde{\Theta}_{\alpha, \gamma}(\mathbf{y}, \mathbf{z}, \mathbf{x}, \mathbf{x}_1, \mathbf{x}_2) = \tilde{\mathcal{H}}_\gamma(\mathbf{y}, \mathbf{z}, \mathbf{x}) + \frac{\alpha^2}{2\gamma}\|\mathbf{x}_1 - \mathbf{x}_2\|^2.$$

In the following, we present the convergence results of [Algorithm 4.1](#).

Theorem 4.2. *Let $g_\sigma : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ of class \mathcal{C}^2 with L -Lipschitz continuous gradient with $L < 1$, and $\mathcal{D}_\sigma = I - \nabla g_\sigma$. Let $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ and h be differentiable with L_f - and L_h -Lipschitz continuous gradient, and let l_f be a constant such that $f + \frac{l_f}{2}\|\cdot\|$ is convex. Suppose that f , g_σ and h are bounded from below, Then, for α and γ satisfying [Assumption 3.2](#) with $L_{f_1} := L_f$ and $l := l_f$, the sequence $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ generated by [Algorithm 4.1](#) which is assumed to be bounded verify that*

- (i) $\{\tilde{\Theta}_{\alpha, \gamma}(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k, \mathbf{x}^{k-1}, \mathbf{x}^{k-2})\}_{k \geq 1}$ is nonincreasing and converges.
- (ii) the sequences $\{\Delta_{\mathbf{x}}^k\}$, $\{\Delta_{\mathbf{y}}^k\}$ and $\{\mathbf{y}^k - \mathbf{z}^k\}$ vanish with rate $\min_{k \leq K} \|\Delta_{\mathbf{x}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, $\min_{k \leq K} \|\Delta_{\mathbf{y}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, and $\min_{k \leq K} \|\mathbf{y}^k - \mathbf{z}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, respectively.
- (iii) any cluster point $\mathbf{u}^* := (\mathbf{y}^*, \mathbf{z}^*, \mathbf{x}^*, \mathbf{x}^*, \mathbf{x}^*)$ of sequence $\{\mathbf{u}^k\}_{k \geq 1}$ is a critical point of the problem (4.3), i.e., it holds that $0 \in \partial F_{\gamma, \sigma}(\mathbf{y}^*)$.
- (iv) if h is a twice continuously differentiable function with a bounded Hessian, i.e., there exists a constant $M > 0$ such that $\|\nabla^2 h(\mathbf{y})\| \leq M$, $\forall \mathbf{y} \in \mathbb{R}^n$, and $F_{\gamma, \sigma}$ in (4.3) is a KL function. Then, the whole sequence $\{\mathbf{u}^k\}_{k \geq 1}$ is convergent.

Proof. Since f and h are differentiable with L_f - and L_h -Lipschitz continuous gradient, the problem (4.3) is a special form of (1.1) with $f_1 := f$ and $f_2 := \frac{1}{\gamma}\phi_\sigma$. Therefore, it follows from [Lemma 3.5](#) and [Lemma 3.6](#) that (i) and (ii) hold. The assertion (iii) can be obtained according to [Theorem 3.7](#), and the conclusion (iv) can be derived from [Theorem 3.10](#). This completes the proof. \blacksquare

4.2. When f is nonsmooth. To cope with the problem (4.3) with a possibly nondifferentiable function f , we propose a nonsmooth extrapolated PnP-DYS method in [Algorithm 4.2](#). In this case, we replace the first proximal subproblem in [Algorithm 4.2](#) by a learned denoiser \mathcal{D}_σ defined in (4.1) to guarantee the theoretical convergence.

In order to analyze the convergence of [Algorithm 4.2](#), we define

$$\hat{\mathcal{H}}_\gamma(\mathbf{y}, \mathbf{z}, \mathbf{x}) = \frac{1}{\gamma}\phi_\sigma(\mathbf{y}) + f(\mathbf{z}) + h(\mathbf{y}) + \frac{1}{2\gamma}\|\mathbf{y} - \mathbf{x} - \gamma\nabla h(\mathbf{y})\|^2 - \frac{1}{2\gamma}\|\mathbf{z} - \mathbf{x} - \gamma\nabla h(\mathbf{y})\|^2,$$

and

$$\hat{\Theta}_{\alpha, \gamma}(\mathbf{y}, \mathbf{z}, \mathbf{x}, \mathbf{x}_1, \mathbf{x}_2) = \hat{\mathcal{H}}_\gamma(\mathbf{y}, \mathbf{z}, \mathbf{x}) + \frac{\alpha^2}{2\gamma}\|\mathbf{x}_1 - \mathbf{x}_2\|^2.$$

Algorithm 4.2 A nonsmooth extrapolated PnP-DYS method

Choose the parameters $\alpha \geq 0$ and $\gamma > 0$. Given \mathbf{x}^0 and $\mathbf{x}^{-1} = \mathbf{x}^0$, set $k = 0$.

while the stopping criteria is not satisfied, **do**

$$\begin{cases} \mathbf{w}^k = \mathbf{x}^k + \alpha(\mathbf{x}^k - \mathbf{x}^{k-1}), \\ \mathbf{y}^{k+1} = \mathcal{D}_\sigma(\mathbf{w}^k), \\ \mathbf{z}^{k+1} = \text{Prox}_{\gamma f}(2\mathbf{y}^{k+1} - \gamma \nabla h(\mathbf{y}^{k+1}) - \mathbf{w}^k), \\ \mathbf{x}^{k+1} = \mathbf{w}^k + (\mathbf{z}^{k+1} - \mathbf{y}^{k+1}). \end{cases}$$

end while

Now we give the convergence results of [Algorithm 4.2](#) based on the conclusions in [Section 3](#) and the discussions in [\[28\]](#).

Theorem 4.3. *Let $g_\sigma : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ of class \mathcal{C}^2 with L -Lipschitz continuous gradient with $L < 1$, and $\mathcal{D}_\sigma = I - \nabla g_\sigma$ with $\text{Im}(\mathcal{D}_\sigma)$ being convex. Let $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ is a proper closed function and h is differentiable L_h -Lipschitz continuous gradient. Suppose that f , g_σ and h are bounded from below, Then, for α and γ satisfying [Assumption 3.2](#) with $L_{f_1} := \frac{L}{\gamma(1-L)}$ and $l := \frac{L}{\gamma(L+1)}$, the sequence $\{(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k)\}_{k \geq 1}$ generated by [Algorithm 4.2](#) which is assumed to be bounded verify that*

- (i) $\left\{ \hat{\Theta}_{\alpha, \gamma}(\mathbf{y}^k, \mathbf{z}^k, \mathbf{x}^k, \mathbf{x}^{k-1}, \mathbf{x}^{k-2}) \right\}_{k \geq 1}$ is nonincreasing and converges.
- (ii) the sequences $\{\Delta_{\mathbf{x}}^k\}$, $\{\Delta_{\mathbf{y}}^k\}$ and $\{\mathbf{y}^k - \mathbf{z}^k\}$ vanish with rate $\min_{k \leq K} \|\Delta_{\mathbf{x}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, $\min_{k \leq K} \|\Delta_{\mathbf{y}}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, and $\min_{k \leq K} \|\mathbf{y}^k - \mathbf{z}^k\| = \mathcal{O}(\frac{1}{\sqrt{K}})$, respectively.
- (iii) any cluster point $\mathbf{u}^* := (\mathbf{y}^*, \mathbf{z}^*, \mathbf{x}^*, \mathbf{x}^*, \mathbf{x}^*)$ of sequence $\{\mathbf{u}^k\}_{k \geq 1}$ is a critical point of the problem [\(4.3\)](#), i.e., it holds that $0 \in \partial F_{\gamma, \sigma}(\mathbf{y}^*)$.
- (iv) if h is a twice continuously differentiable function with a bounded Hessian, i.e., there exists a constant $M > 0$ such that $\|\nabla^2 h(\mathbf{y})\| \leq M, \forall \mathbf{y} \in \mathbb{R}^n$, and $F_{\gamma, \sigma}$ in [\(4.3\)](#) is a KL function. Then, the whole sequence $\{\mathbf{u}^k\}_{k \geq 1}$ is convergent.

Proof. It follows from [Proposition 4.1](#) that ϕ_σ is $\frac{L}{L+1}$ -weakly convex and $\nabla \phi_\sigma$ is $\frac{L}{1-L}$ -Lipschitz on $\text{Im}(\mathcal{D}_\sigma)$. Thus, the problem [\(4.3\)](#) can be seen as a special form of [\(1.1\)](#) with $f_1 = \frac{1}{\gamma} \phi_\sigma$ and $f_2 = f$. Since $\text{Im}(\mathcal{D}_\sigma)$ is convex, it follows from [\[28, Appendix C.2\]](#) and [Lemma 3.5](#) that (i) and (ii) hold. According to the assumptions on g_σ , we know that \mathcal{D}_σ is continuous on $\text{Im}(\mathcal{D}_\sigma)$, and then the assertion (iii) can be obtained according to [Theorem 3.7](#). Moreover, the conclusion (iv) can be derived from [Theorem 3.10](#). This completes the proof. ■

Remark 4.4. As discussed in [\[28, 47\]](#), one can ensure that the Lipschitz constant $L < 1$ for ∇g_σ is to softly constrain it by penalizing the spectral norm of the Hessian of g_σ in the denoiser training loss. This approach will be further explained in the experiments. In addition, if $L > 1$, one can relax the deep prior with a parameter $\eta \in [0, 1]$, given by $\mathcal{D}_\sigma^\eta = \eta \mathcal{D}_\sigma + (1-\eta)I$. It is important to note that the relaxed deep prior \mathcal{D}_σ^η exhibits the same property as stated in [Proposition 4.1](#). More specifically, \mathcal{D}_σ^η continues to be the proximal operator of a certain

weakly convex functional. As a result, the condition becomes $\eta L < 1$, which can be easily guaranteed since $\eta \in [0, 1]$. We refer to [25, Subsection 3.4] for more discussions.

5. Numerical experiments. In this section, we implement the extrapolated DYS algorithm with or without PnP denoiser on image deblurring and image super-resolution tasks, and compare numerical results with other advanced models and methods. All experiments are implemented with PyTorch on an NVIDIA RTX A6000 GPU.

We consider the image restoration problem with both sparse-induced regularization and Tikhonov regularization, whose mathematical model can be read as

$$(5.1) \quad \min_{\mathbf{x} \in \mathbb{R}^n} \frac{1}{2\nu^2} \|A\mathbf{x} - \mathbf{b}\|^2 + r(\mathbf{x}) + \frac{\beta}{2} \|\mathbf{x}\|^2,$$

where $r(\cdot)$ is the sparse-induced regularizer which maybe nonconvex, \mathbf{b} is the observation, ν is the Gaussian noise level and A is the linear operator. When A denotes the blur kernel, the model (5.1) corresponds to image deblurring problem, which aims to restore a clean image \mathbf{x}^* from the observed image \mathbf{b} . Additionally, if $A = SB$, where B denotes the blur operator and S is the standard s -fold downampler (i.e., selecting the upper-left pixel for each distinct $s \times s$ patch), the model (5.1) reduces to the image super-resolution problem. This problem involves enhancing the resolution and quality of a low-resolution image to generate a high-resolution version of the same image. We can see that the model (5.1) falls into the form of (1.1) with $f_1(\mathbf{x}) = \frac{1}{2\nu^2} \|A\mathbf{x} - \mathbf{b}\|^2$, $f_2(\mathbf{x}) = r(\mathbf{x})$ and $h(\mathbf{x}) = \frac{\beta}{2} \|\mathbf{x}\|^2$. Additionally, the following model with sparse-induced regularization and box constraint is also widely used in solving image deblurring and image super-resolution problems:

$$(5.2) \quad \min_{\mathbf{x} \in \mathcal{B}} \frac{1}{2\nu^2} \|A\mathbf{x} - \mathbf{b}\|^2 + r(\mathbf{x}),$$

where \mathcal{B} is a convex box. Model (5.2) is a special form of (1.1) if $r(\cdot)$ is smooth with $f_1(\mathbf{x}) = r(\mathbf{x})$, $f_2(\mathbf{x}) = \delta_{\mathcal{B}}(\mathbf{x})$ and $h(\mathbf{x}) = \frac{1}{2\nu^2} \|A\mathbf{x} - \mathbf{b}\|^2$, where $\delta_{\mathcal{B}}(\cdot)$ denotes the indicator function.

In the experiments, we will consider two cases of $r(\cdot)$ for (5.1) and (5.2) as follows:

1. $r(\mathbf{x}) = \|\mathbf{x}\|_{\text{TV}}$, the isotropic total-variational (TV) regularizer [20, 54];
2. $r(\mathbf{x}) = \frac{1}{\gamma} \phi_{\sigma}(\mathbf{x})$, the nonconvex regularizer in (4.2) induced by Gradient Step (GS) denoiser \mathcal{D}_{σ} .

We refer to the model (5.1) with the above two regularizers as TVTik and DeTik. Similarly, the model (5.2) with both regularizers is denoted as TVBox and DeBox, respectively. As discussed in Section 4, DeTik can be solved by Algorithm 4.1, while DeBox should be solved by Algorithm 4.2 due to the nonsmoothness of $\delta_{\mathcal{B}}$. For the classical TV-based models, i.e., TVTik and TVBox, the split Bregman algorithm is applicable. More specifically, we import the image processing package ‘scikit-image’ in Python with ‘skimage.restoration.denoise.tv_bregman’ for solving the isotropic TV-subproblem with a maximum iteration of 100. Certainly, Algorithm 3.1 also can be used to solve TVTik, as there are two smooth terms with Lipschitz continuous gradient involved in (5.1). We initialize all the tested algorithms with $\mathbf{x}^{-1} = \mathbf{x}^0 = \mathbf{b}$. The algorithms are terminated when the relative difference between consecutive values of the objective function is less than $\varepsilon = 10^{-8}$ or the number of iterations exceeds $k_{\max} = 1000$.

As aforementioned, we utilize the deep GS denoiser to replace the traditional regularizer. Specifically, in the experiments, we employ the classical DRUNet [78] as our denoiser \mathcal{D}_{σ} .

DRUNet incorporates both U-Net and ResNet architectures and takes an additional noise level map as input, achieving state-of-the-art performance in Gaussian noise removal. To ensure $L < 1$ of the Lipschitz constant of ∇g_σ in (4.1), following the approach in [47], we regularize the training loss of \mathcal{D}_σ using the spectral norm of the Hessian of g_σ as follows:

$$(5.3) \quad \mathcal{L}_S(\sigma) = \mathbb{E}_{\mathbf{x} \sim p, \xi_\sigma \sim \mathcal{N}(0, \sigma^2)} [\|\mathcal{D}_\sigma(\mathbf{x} + \xi_\sigma) - \mathbf{x}\|^2 + \mu \max(\|\nabla^2 g_\sigma(\mathbf{x} + \xi_\sigma)\|_S, 1 - \epsilon)],$$

where p is the distribution of a dataset of clean images and $\|\cdot\|_S$ is the spectral norm. We set $\epsilon = 0.1$ and $\mu = 0.01$ according to [28]. Following the setting of [27], we have retrained the DRUNet [78] with loss function (5.3) on the Berkeley segmentation dataset, Waterloo Exploration Database, DIV2K dataset, and Flickr2K dataset. For the image deblurring problem, ten different blur kernels¹ (from Ker1 to Ker10) and three noise levels: $\nu = \{2.55, 7.65, 12.75\}$ will be used to simulate the degraded image.

5.1. Effect of extrapolation. We first test the effectiveness of extrapolation parameter α by applying Algorithm 4.1 to solve the DeTik model. For the DeTik model, we know that $L_{f_1} = \frac{1}{\nu^2} \lambda_{\max}(A^\top A)$, $l = -L_{f_1}$ and $L_h = \beta$, where λ_{\max} denotes the maximal eigenvalue of a given matrix. In the experiment, we set the model parameter $\beta \in [0.0005, 0.001]$ for different noise levels in (5.1). It follows from Assumption 3.2 that $0 \leq \alpha < \Lambda(\gamma)$. Therefore, for a given and fixed γ that satisfies (3.4), we test the values of $\alpha = \{0, 0.25, 0.50, 0.75, 0.99\} * \Lambda(\gamma)$ by performing a numerical comparison of the computational cost and the quality of recovery for the image deblurring problem.

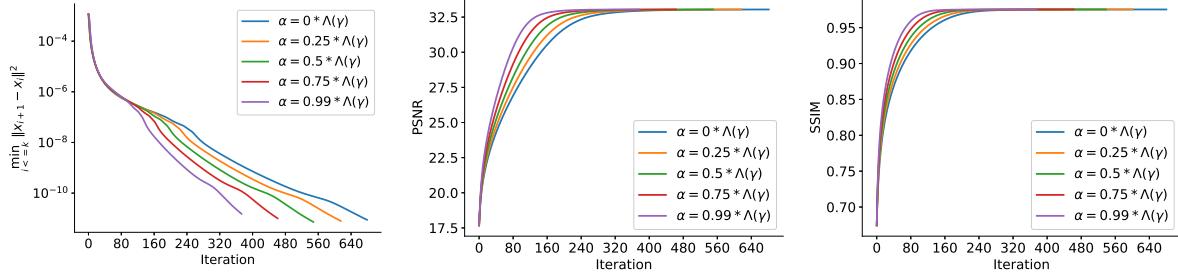


Figure 1. Effect of α in Algorithm 4.1 for solving DeTik model on ‘butterfly’ with Ker1 and noise level 2.55. Increasing the extrapolation parameter α speeds-up the convergence of the algorithm. This increased convergence speed does not alter the quality of the proposed restoration.

In Figure 1, we report the effect of α on ‘butterfly’ with Ker1 and noise level 2.55. More specifically, the evolution curves of the convergence of residual $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|$ at rate $\min_{j \leq k} \|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2$, PSNR and SSIM values with respect to the number of iterations are presented, which showcases the advantage of the proposed extrapolation step. Furthermore, the detailed results include iteration number (Iter.), computational time in seconds (Time(s)), recovered PSNR (dB), and SSIM for three tested images (butterfly, leaves, and starfish) in Sect3C with different levels of noise are reported in Appendix A. From the presented results, we can see that Algorithm 4.1 exhibits improved performance as the extrapolation stepsize α

¹https://github.com/Huang-chao-yan/convergent_pnp/tree/main/kernels

increases, particularly in terms of computational cost. In our subsequent experiments, we set $\alpha = 0.99 * \Lambda(\gamma)$ for a given γ to obtain results more efficiently.

5.2. Parameter analysis. In this subsection, we study the influence of the parameters and initialization of [Algorithm 4.1](#) for solving the DeTik model. Recall that DeTik can be read as

$$\min_{\mathbf{x} \in \mathbb{R}^n} \frac{1}{2\nu^2} \|A\mathbf{x} - \mathbf{b}\|^2 + \frac{1}{\gamma} \phi_\sigma(\mathbf{x}) + \frac{\beta}{2} \|\mathbf{x}\|^2,$$

where ν and σ are the noise levels of the synth input image and the denoiser ϕ_σ , respectively. We fix model parameter β for different noise levels as that in the last subsection, and roughly estimate σ proportionally to the input noise level ν as $\sigma = \sigma_\nu * \nu$, where σ_ν is a positive constant. Consequently, the parameters we will be testing are $\gamma_\nu = \frac{\nu^2}{\gamma}$ and $\sigma_\nu = \sigma/\nu$.

In [Figure 2](#), we display the average PSNR value of Set3C using 10 tested blur kernels under a noise level of 2.55, where γ_ν ranges from 0.1 to 1.4 with a step size of 0.1. From the results, we can see that the instances with γ_ν values around 1 exhibit superior performance compared to other cases. This observation is further supported by the restored images on the right-hand side, which demonstrate that the quality of that corresponding to $\gamma_\nu = 1$ is better than those for $\gamma_\nu = 0.01$ and $\gamma_\nu = 1.3$. When $\gamma_\nu = 0.01$, the noise is removed, but the blur remains. for a larger value of $\gamma_\nu = 1.3$, both the noise and blur remain. Hence, in our experiments, we chose $\gamma_\nu = 1$ to address the noise level of 2.55. Next, we test the effect of the parameter σ_ν and present the average PSNR value of Set3C with 10 tested blur kernels under noise level 2.55 for $\sigma_\nu \in [0.6, 2.4]$ with a step size 0.2 in [Figure 3](#). The results indicate that almost no deblurring occurs when the value of σ_ν is small. Conversely, as σ_ν increases, excessive smoothing takes place, resulting in the loss of image details. Based on both the curve analysis and the visual outcomes, we select $\sigma_\nu \in [1, 2]$.

We further investigate the impact of the initialization of [Algorithm 4.1](#). In [Figure 4](#), we plot the average PSNR value of Set3C obtained from 10 tested blur kernels under a noise level of 2.55. Due to the nonconvex regularizer, the proposed scheme is sensitive to initial value. Following the setting of [28], the initial \mathbf{x}^0 is varied with different noise levels: $\{0.01, 2.55, 5, 7.5, 10\}$. Based on the PSNR curve and visual quality in [Figure 4](#), we can see

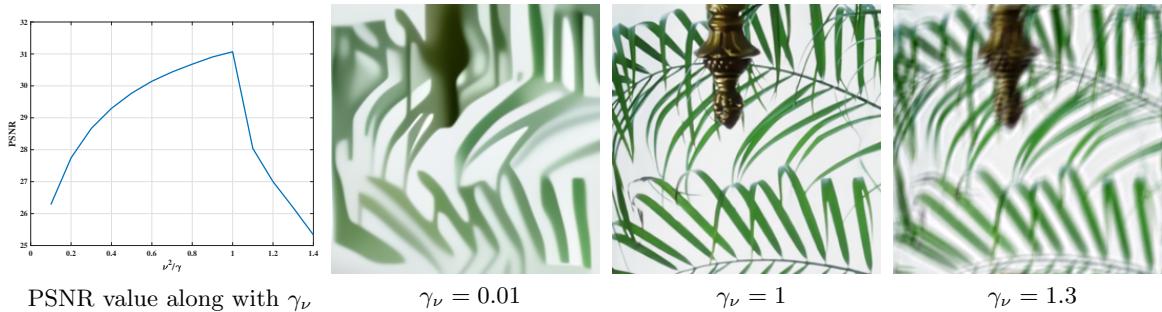


Figure 2. Influence of the parameter $\gamma_\nu = \frac{\nu^2}{\gamma}$ for deblurring with DeTik model. First column: average PSNR along with the γ_ν . The other parameters are fixed. Remaining columns: visual results for deblurring ‘leaves’ with various γ_ν .

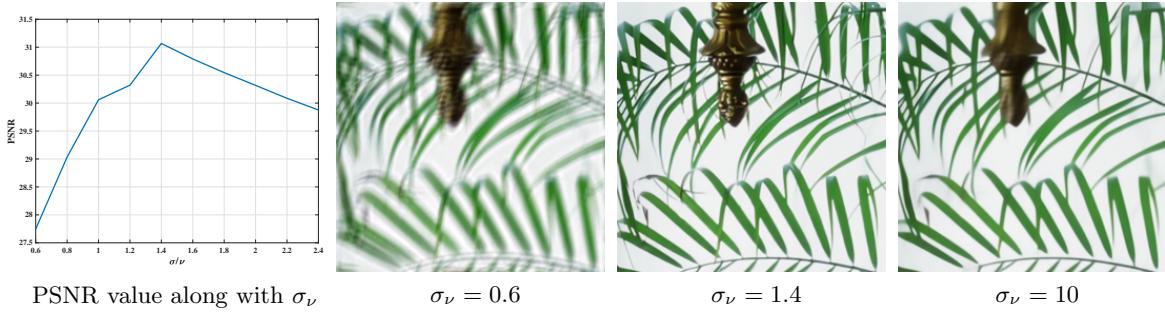


Figure 3. Influence of the parameter σ_ν for deblurring with DeTik model. First column: average PSNR along with the σ_ν . The other parameters are fixed. Remaining columns: visual results for deblurring ‘leaves’ with various σ_ν .

that a suitable initial input is crucial for the image deblurring task. When an initial input closely resembles the ground truth image, certain images may not undergo further iterations and terminate prematurely, particularly when the stopping criteria remain unchanged. On the other hand, if a heavily noisy image serves as the initial input, the iteration process progresses smoothly. However, the resulting image retains the heavy noise due to the low-level denoiser’s inability to effectively handle such noise levels. In our experiments, we adopt the observation as the initial input to ensure the validity of the obtained results.

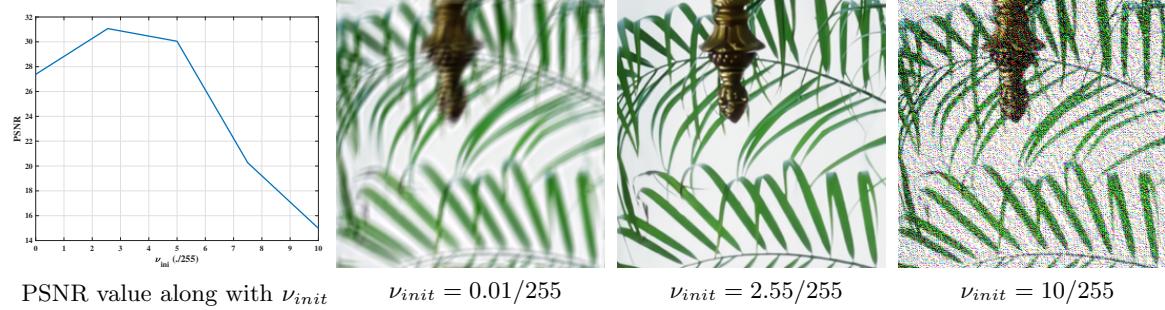


Figure 4. Influence of the initialitation \mathbf{x}^0 for deblurring with DeTik model. First column: average PSNR along with different \mathbf{x}^0 . The other parameters are fixed. Remaining columns: visual results for deblurring ‘leaves’ with various \mathbf{x}^0 .

5.3. Image deblurring and super-resolution. In this subsection, we are devoted to demonstrating the effectiveness and robustness of the proposed [Algorithm 4.1](#) and [Algorithm 4.2](#) by solving image deblurring and super-resolution problems.

As discussed in [subsection 4.1](#), [Algorithm 4.1](#) can be utilized to solve DeTik model due to the smoothness of $f_1(\mathbf{x}) = \frac{1}{2\nu^2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$, where ν is the noise level; [Algorithm 4.2](#) can be used to solve DeBox model mentioned in [\(5.2\)](#). We first determine an appropriate γ satisfying [Assumption 3.2](#) and set $\alpha = 0.99 * \Lambda(\gamma)$. We consider Gaussian noise with 3 noise levels $\nu \in \{2.55, 7.65, 12.75\}/255$, i.e., $\nu \in \{0.01, 0.03, 0.05\}$, and 2 scale factors $\times 2, \times 3$. For the tested noise levels, we set $\sigma = \{1.4\nu, 0.7\nu, 0.6\nu\}$, $\nu^2/\gamma = \{1, 0.9, 0.6\}$ in [Algorithm 4.1](#) for both image deblurring and super-resolution. For all noise levels, we set $\sigma = \{2\nu, 1\nu, 0.75\nu\}$

Table 1

Numerical results (PSNR(dB)) of our DeTik and DeBox for image deblurring with Ker1 and 3 noise levels on Dataset Set3C.

Noise Level	2.55			7.65			12.75		
	Images	Butterfly	Leaves	Starfish	Butterfly	Leaves	Starfish	Butterfly	Leaves
Degraded	17.68	16.50	21.56	17.48	16.34	21.09	17.10	16.06	20.28
DeTik	33.18	34.02	33.14	29.91	30.34	29.78	27.94	28.06	27.58
DeBox	33.62	33.80	33.53	29.75	30.20	29.59	27.90	27.96	27.57

and $\nu^2/\gamma = \{5, 1.5, 1\}$ in [Algorithm 4.2](#) for both tasks. We test the proposed algorithms for different tasks and compare the numerical results recovered by DeTik and DeBox.

For the image deblurring task, we test four classical datasets, i.e., Set3C, Set14, Kodak24, and Set17, with different blur kernels and noise levels. For the sake of brevity, we present the image deblur results of Ker1 with various noise levels on Set3C in [Table 1](#), and more results can be found in [Appendix B](#). Our proposed methods demonstrate competitive performance in the task of image deblurring across different noise levels. On the other hand, the visual results of image ‘powerpoint2002’ in Set14 degraded by the blur Ker6 and noise level 12.75 can be found in [Figure 5](#). To assess the convergence of the proposed algorithms in the experimental aspect, the evolution and energy curves are plotted and presented alongside the corresponding recovered images.

For the image super-resolution task, we set the scale factor as $\times 2$ and $\times 3$. Meanwhile, the blur and noise (mentioned in the deblurring task) are also considered in the experiments. The image super-resolution results on datasets Set5, CBSD68, and Urban100 are reported in [Appendix B](#). More specifically, we report the numerical results on Set5 in [Table 2](#). Furthermore, the visual results for noise level 7.65 with blur Ker8 and scale factor $\times 2$ are shown in [Figure 6](#). The evolution and energy curves demonstrate the convergence of the proposed approaches in the experiment, which aligns with our theoretical results.

Table 2

Numerical results (PSNR(dB)) of our DeTik and DeBox for image super-resolution with Ker1 and 3 noise levels and scales $\times 2$ and $\times 3$ on Dataset Set5.

Scales	Noise Level	2.55					7.65					12.75				
		Images	Baby	Bird	Butterfly	Head	Woman	Baby	Bird	Butterfly	Head	Woman	Baby	Bird	Butterfly	Head
$\times 2$	Degraded	28.82	24.73	17.75	25.52	22.73	27.61	24.23	17.64	24.93	22.41	25.90	23.36	17.43	23.94	21.82
	DeTik	33.93	31.90	27.88	29.17	30.63	32.49	29.75	26.42	28.53	29.02	31.51	27.87	24.67	27.74	26.81
	DeBox	34.26	31.85	27.19	29.19	30.51	32.45	29.53	26.16	28.32	28.89	31.42	27.85	24.77	27.63	26.95
$\times 3$	Degraded	28.75	24.72	17.75	25.43	22.66	27.20	24.05	17.61	24.65	22.23	25.15	22.94	17.34	23.40	21.47
	DeTik	32.40	29.17	22.51	28.29	27.44	31.51	27.59	23.68	27.77	26.47	30.46	26.05	22.35	27.20	25.14
	DeBox	32.53	29.09	22.91	27.67	27.06	31.54	27.44	23.37	27.69	26.45	30.63	26.15	22.44	27.15	25.20

5.4. Comparison with state-of-the-art methods. In the preceding subsections, we have substantiated the validity of the proposed algorithm in handling both smooth and non-smooth objective functions. However, these evaluations alone do not entirely showcase the advantage of our method. Hence, in this subsection, we conduct a comparative analysis with state-of-the-art methods to provide further evidence of the exceptional effectiveness of our approach.

5.4.1. Comparisons with advanced deblurring models. Following the implementation of the plug-and-play strategy, our proposed method integrates a denoiser into the objective function. Consequently, several methods that employ the same strategy are compared. While

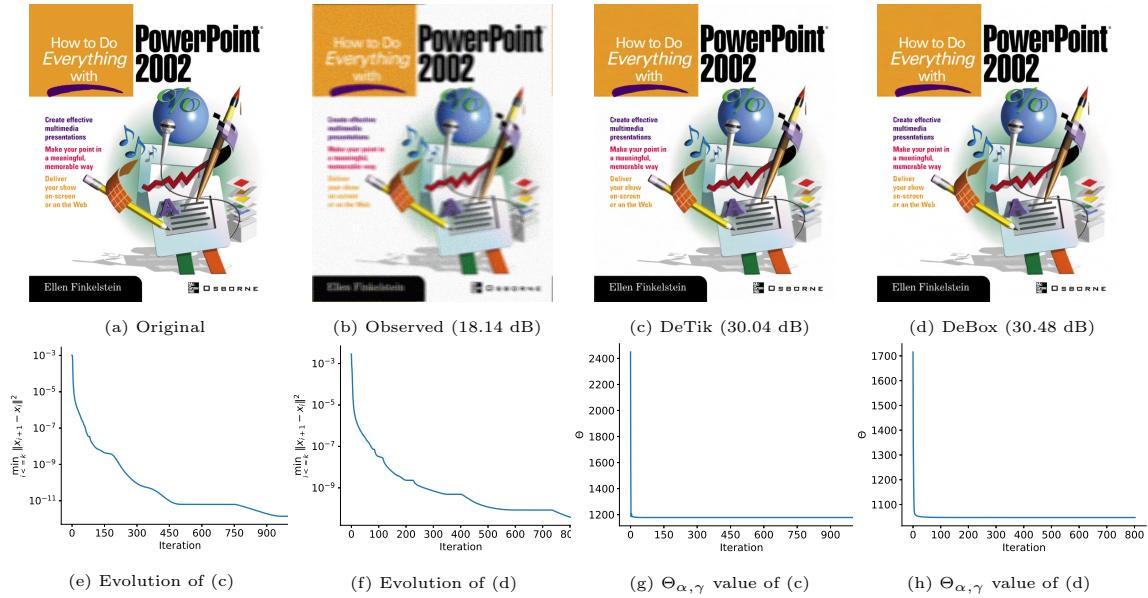


Figure 5. The deblurring results of DeTik and DeBox on image degradation with Ker6 and noise level 12.75. The evolution and $\Theta_{\alpha,\gamma}$ value along with the number of iterations.

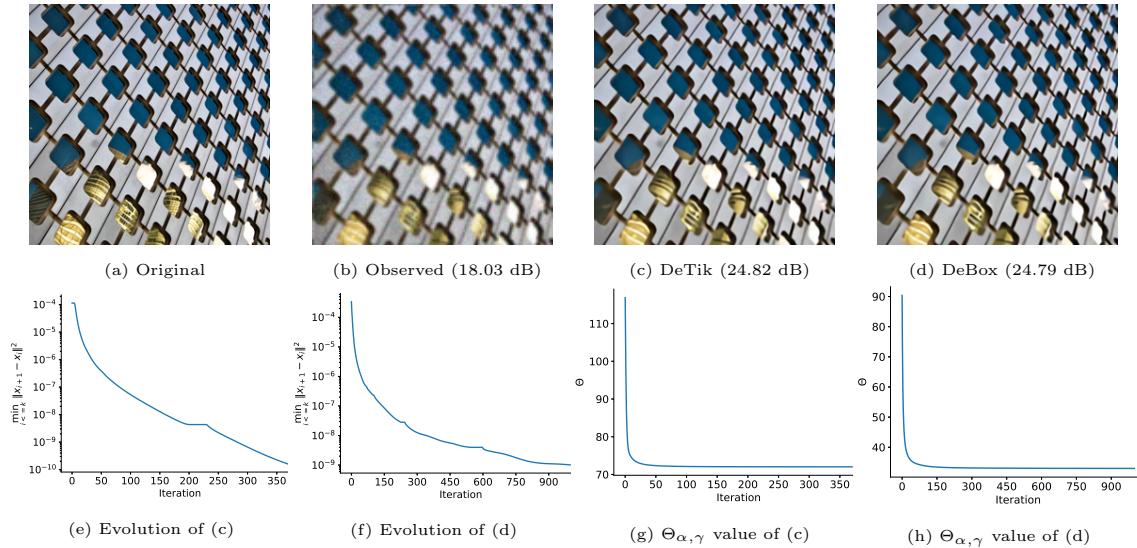


Figure 6. The super-resolution results of DeTik and DeBox on image degradation with scale (x2) Ker8 and noise level 7.65. The evolution and $\Theta_{\alpha,\gamma}$ value along with the number of iterations.

these methods yield competitive results, it is important to note that our proposed method holds a distinct advantage in terms of theoretical analysis. Specifically, our method guarantees convergence, whereas not all of the compared methods provide such a guarantee. In this paper, some plug-and-play methods and unrolling models DWDN [18], DPIR [78] with IRCNN [80] (DP-IRCNN), DPIR [78] with DRUNet (DPIR), and DREDDUN [30] are compared. All the

Table 3

Comparison on average image deblurring results ($PSNR(dB)$) of the state-of-the-art methods with our methods on Set3C, Set14, and Set17 datasets.

Datasets	Noise Level	Degraded	DWDN	DP-IRCNN	DPIR	DREDDUN	Alg. 3.1		Alg. 4.1		ADMM		Alg. 4.2	
							TVTik	DeTik	TVBox	DeBox	TVBox	DeBox	TVBox	DeBox
Set3C	2.55	19.93	30.92	30.92	32.55	30.71	29.46	30.98	28.84	31.24				
	7.65	19.52	28.62	27.60	28.60	28.62	25.10	28.78	25.18	28.62				
	12.75	18.84	26.92	25.93	26.80	26.97	23.34	27.08	23.39	27.08				
Set14	2.55	22.82	31.08	30.64	31.76	31.16	28.47	30.17	27.68	30.08				
	7.65	22.10	28.41	28.13	28.79	28.57	26.68	28.47	26.06	28.33				
	12.75	21.03	27.20	27.03	27.32	27.38	25.30	27.30	25.10	27.32				
Set17	2.55	25.28	33.14	32.35	33.98	33.41	30.56	32.60	30.67	32.43				
	7.65	24.07	30.39	29.83	30.64	30.62	27.73	30.64	27.85	30.55				
	12.75	22.55	28.93	28.74	29.40	29.24	26.33	29.25	26.54	29.29				

compared codes were obtained either from the official published versions or were graciously provided by the authors themselves.

To provide more comprehensive results of the image deblurring, we compiled the average

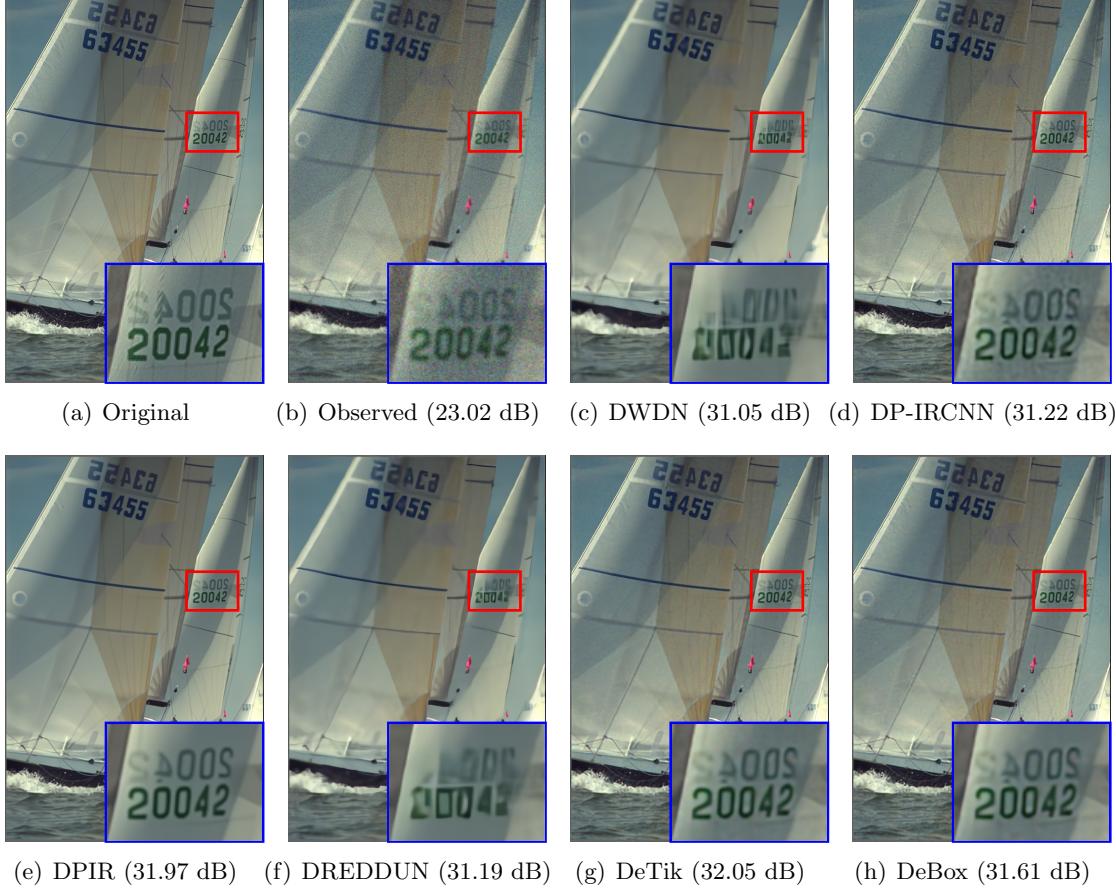


Figure 7. Image deblurring results with Ker9 and noise level 12.75. (g) is the result of proposed Algorithm 4.1 for DeTik; (h) is the result of the proposed Algorithm 4.2 for DeBox.

Table 4

Comparison on average image super-resolution results ($PSNR(dB)$) of the state-of-the-art methods with our methods on Set5 and Urban100 datasets.

Scales	Datasets	Noise Level	Bicubic	USRNet	DP-IRCNN	DPIR	DREDDUN	Alg. 3.1	Alg. 4.1	ADMM	Alg. 4.2
								TVTik	DeTik	TVBox	DeBox
$\times 2$	Set5	2.55	24.21	30.75	29.33	31.07	30.49	28.16	30.29	27.70	30.51
		7.65	23.48	29.38	27.76	28.81	28.46	26.59	29.16	26.04	29.12
		12.75	22.45	27.98	26.96	27.60	27.34	23.26	27.91	24.40	27.99
	Urban100	2.55	19.15	25.67	25.34	25.40	25.43	21.23	24.10	21.51	23.86
		7.65	18.93	24.49	23.69	24.52	23.81	19.86	24.34	20.80	23.18
		12.75	18.53	22.92	22.68	23.18	22.89	19.24	23.29	19.81	22.34
$\times 3$	Set5	2.55	23.29	30.11	27.99	28.95	28.55	25.79	28.20	26.09	28.42
		7.65	22.71	28.19	26.52	27.22	27.11	25.19	27.65	25.72	27.64
		12.75	21.84	27.04	25.68	26.18	26.14	24.57	26.61	25.03	26.67
	Urban100	2.55	18.54	24.03	22.80	23.62	23.12	21.52	23.14	20.45	21.72
		7.65	18.35	22.12	21.90	22.36	21.67	20.05	21.65	19.92	21.45
		12.75	18.00	20.93	20.37	20.91	20.91	19.16	20.70	19.40	20.93

results for 10 blur kernels and 3 noise levels in [Table 3](#). We list the results of the proposed two algorithms with two cases, respectively. From the numerical results, it becomes evident that our DeTik and DeBox yield competitive performance compared to deep learning-based plug-and-play and unrolling methods. Nevertheless, it is important to note that the traditional TVTik and TVBox cases may exhibit less satisfactory results, which is understandable considering that deep learning-based models have the advantage of leveraging more prior information compared to traditional priors. Furthermore, the visual results are depicted in [Figure 7](#) for a more comprehensive illustration. Note that we only present our PnP-based results (DeTik and DeBox) for visual comparison. We can see that although the PnP-based methods usually cause over-smoothing, the proposed algorithms (DeTik and DeBox) exhibit superior performance in detail restoration compared to the other methods.

5.4.2. Comparisons with advanced super-resolution models. For image super-resolution task, USRNet [79], IRCNN [80] (DP-IRCNN), DPIR [78] with DRUNet (DPIR), and DRED-DUN [30] are compared. All the compared codes used in our study were obtained either from the official published versions or were graciously provided by the authors themselves. Note that when addressing the image super-resolution task with sample scales $\times 2$ and $\times 3$, we simulated the degraded images by incorporating blur and noise during the sampling process. Specifically, we added 10 blur kernels and introduced the 3 Gaussian noises mentioned earlier.

The average image super-resolution results of the proposed algorithms with other advanced super-resolution models are listed in [Table 4](#). We can see that our methods achieve competitive results under different scaling factors. While it is true that some compared methods outperform the proposed algorithm in some degradation cases, it is important to note that most of these methods lack convergence guarantees. Furthermore, we conducted a visual comparison of the renderings in [Figure 8](#), in which the proposed methods exhibit distinct advantages. Our proposed method excels in detail recovery when compared to other methods. Hence, based on both theoretical guarantees and experimental evidence, the algorithms we proposed exhibit distinct advantages when applied to image super-resolution tasks.

6. Conclusions. This paper studied an extrapolated three-operator splitting method for solving a class of structural nonconvex optimization problems that minimize the sum of three functions. Our method extends the Davis-Yin splitting approach, which encompasses the

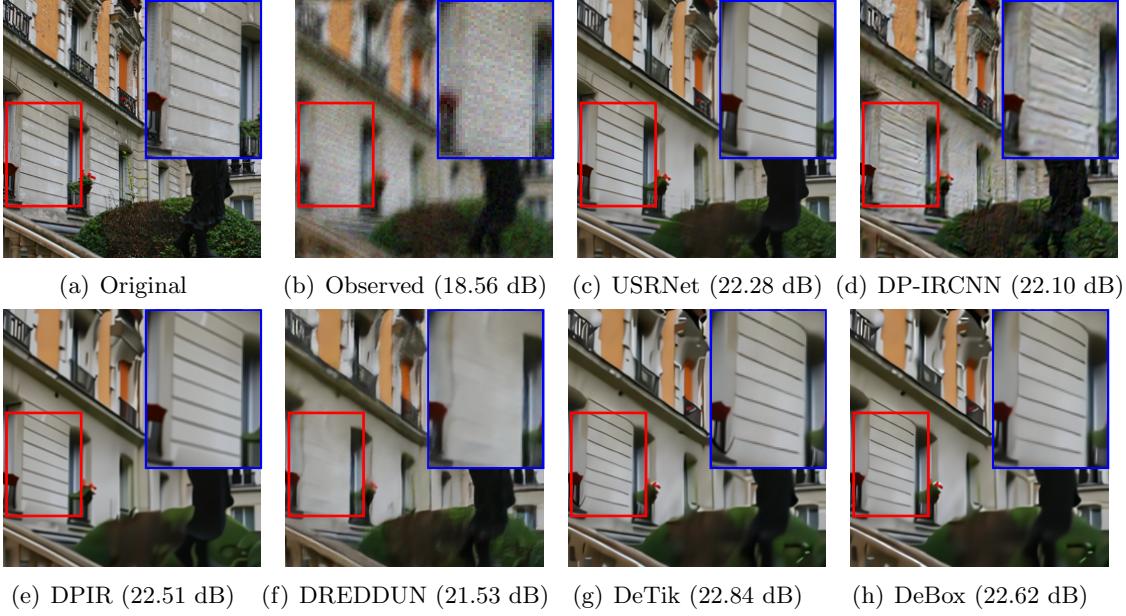


Figure 8. Image super-resolution results with scale $\times 2$, Ker1 and noise level 7.65. (g) is the result of proposed [Algorithm 4.1](#) for DeTik; (h) is the results of the proposed [Algorithm 4.2](#) for DeBox.

widely-used forward-backward and Douglas-Rachford splitting methods, and introduces extrapolation techniques to handle nonconvex optimization problems. The convergence to a stationary point has been established by leveraging the Kurdyka-Łojasiewicz property. To further enhance the applicability, we applied the proposed splitting method within the Plug-and-Play (PnP) approach, incorporating a learned denoiser. The extrapolated PnP-based splitting methods replace the regularization step with a denoiser based on gradient step-based techniques, and we have provided theoretical guarantees for their convergence. This integration allows us to leverage the power of learning-based models. Furthermore, we have conducted extensive numerical experiments to evaluate the performance of our proposed methods on image deblurring and super-resolution problems. The results of these experiments have demonstrated the advantages and efficiency of the extrapolation strategy employed in our algorithmic framework. Importantly, our experiments have highlighted the superiority of the learning-based model with the PnP denoiser in terms of image quality.

In future research, we will consider the variants of the proposed method, such as incorporating line search, inexact solving techniques, and dynamically adapting parameter choices, to extend the applicability of our framework to a broader range of practical problems. Further theoretical investigations are warranted to establish convergence guarantees for splitting methods combined with other efficient PnP denoisers, such as the Bregman-based denoiser proposed in [26] for various Poisson inverse problems. In addition, investigating the potential applications of the proposed methods in the field of medical image processing is a crucial aspect of our future work.

Appendix A. Experimental results on effect of extrapolation. We report the average image deblurring results under 10 different blur kernels and 3 noise levels in [Table 5](#), which

include iteration number (Iter.), computational time in seconds (Time(s)), recovered PSNR (dB), and SSIM for three tested images (butterfly, leaves, and starfish) in Sect3C with different levels of noise. From the presented results, we can see that [Algorithm 4.1](#) exhibits improved performance as the extrapolation stepsize α increases, particularly in terms of computational cost. Increasing the extrapolation parameter α speeds-up the convergence of the algorithm. This increased convergence speed does not alter the quality of the proposed restoration.

Table 5

Parameter analysis of α in [Algorithm 4.1](#) for image deblurring by DeTik model on the dataset Set3C with different noise levels.

α	Image	butterfly			leaves			starfish		
		Noise Level	2.55	7.65	12.75	2.55	7.65	12.75	2.55	7.65
0	Iter.	681	1001	512	436	596	428	388	396	513
	Time(s)	25.97	36.79	18.87	15.54	20.92	15.61	13.56	13.75	18.30
	PSNR	33.18	29.91	27.94	33.97	30.33	28.05	33.11	29.78	27.57
	SSIM	0.9760	0.9569	0.9367	0.9890	0.9760	0.9617	0.9551	0.9233	0.8866
$0.25 * \Lambda(\gamma)$	Iter.	617	972	457	383	532	381	340	351	417
	Time(s)	22.54	37.87	17.03	13.13	19.00	13.72	11.42	12.40	14.74
	PSNR	33.18	29.91	27.94	33.97	30.33	28.05	33.11	29.78	27.57
	SSIM	0.9760	0.9569	0.9367	0.9890	0.9760	0.9617	0.9551	0.9233	0.8866
$0.50 * \Lambda(\gamma)$	Iter.	550	901	403	332	467	226	297	308	396
	Time(s)	20.07	36.18	15.07	11.79	16.32	12.07	10.11	10.51	13.88
	PSNR	33.18	29.91	27.94	33.97	30.33	28.05	33.11	29.78	27.57
	SSIM	0.9760	0.9569	0.9367	0.9890	0.9760	0.9617	0.9551	0.9233	0.8866
$0.75 * \Lambda(\gamma)$	Iter.	463	873	348	279	405	317	251	265	507
	Time(s)	16.77	37.86	12.46	10.06	14.28	10.99	8.71	9.35	17.20
	PSNR	33.18	29.91	27.94	33.9	30.33	28.05	33.11	29.78	27.58
	SSIM	0.9760	0.9569	0.9367	0.9890	0.9760	0.9617	0.9551	0.9233	0.8866
$0.99 * \Lambda(\gamma)$	Iter.	375	861	491	225	344	26	204	223	276
	Time(s)	13.63	32.35	18.41	7.66	12.10	9.21	6.69	7.69	9.31
	PSNR	33.18	29.91	27.95	33.97	30.33	28.05	33.14	29.78	27.57
	SSIM	0.9760	0.9569	0.9367	0.9890	0.9760	0.9617	0.9551	0.9233	0.8866

Appendix B. Experimental results on robustness of [Algorithm 4.1](#) and [Algorithm 4.2](#).

To further demonstrate the effectiveness of the proposed methods, we compare the results recovered by the model TVTik and DeTik in [Figure 9](#) and [Figure 10](#), and the model DeBox and TVBox in [Figure 11](#) and [Figure 12](#), for image deblurring and super-resolution, respectively.

We use the Matlab built-in function ‘boxplot’ to create a box plot. As shown in [Figure 9](#), each picture contains 9 boxes. The yellow, pink, and blue boxes represent the average PSNR values of the degraded images, the images restored by TVTik and DeTik, and the first, second, and third sets of yellow, pink, and blue boxes correspond to the noise levels of 2.55, 7.65, and 12.75, respectively. On each box, the central mark indicates the median, and the bottom and top edges of the box indicate the 25th and 75th percentiles. The whiskers extend to the most extreme data points not considered outliers, and the outliers are plotted individually using the dot symbol. From the box plot, we can see that the median of DeTik is higher than that of TVTik. Note that the TVTik model also enhances the quality of the degraded image when compared to the yellow boxes. These results demonstrate that [Algorithm 4.1](#) is efficient in image restoration, as it successfully restores images affected by 10 different kernels and 3 different noise levels. Similarly to the deblurring results, the box plot is presented to show the super-resolution outcomes. The first and second rows of the box plot display the

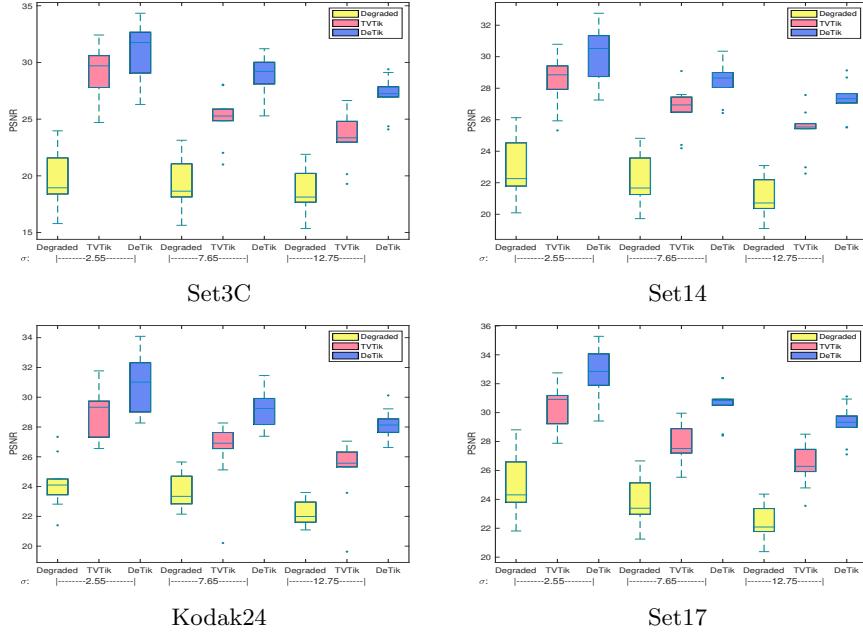


Figure 9. Average results ($PSNR(dB)$) of $TVTik$ and $DeTik$ for image deblurring with 10 different blur kernels and 3 noise levels on $Set3C$, $Set14$, $Kodak24$, and $Set17$ datasets.

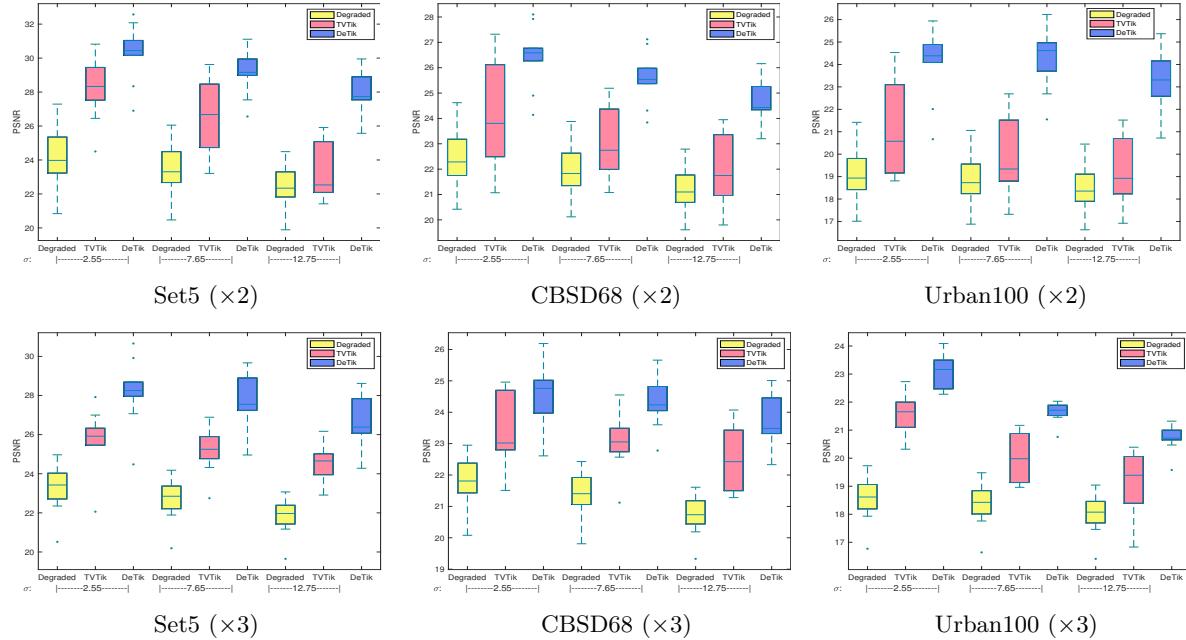


Figure 10. Average results ($PSNR(dB)$) of $TVTik$ and $DeTik$ for image super-resolution with 2 scales ($\times 2$ and $\times 3$), 10 different blur kernels and 3 noise levels on $Set5$, $CBSD68$, and $Urban100$ datasets.

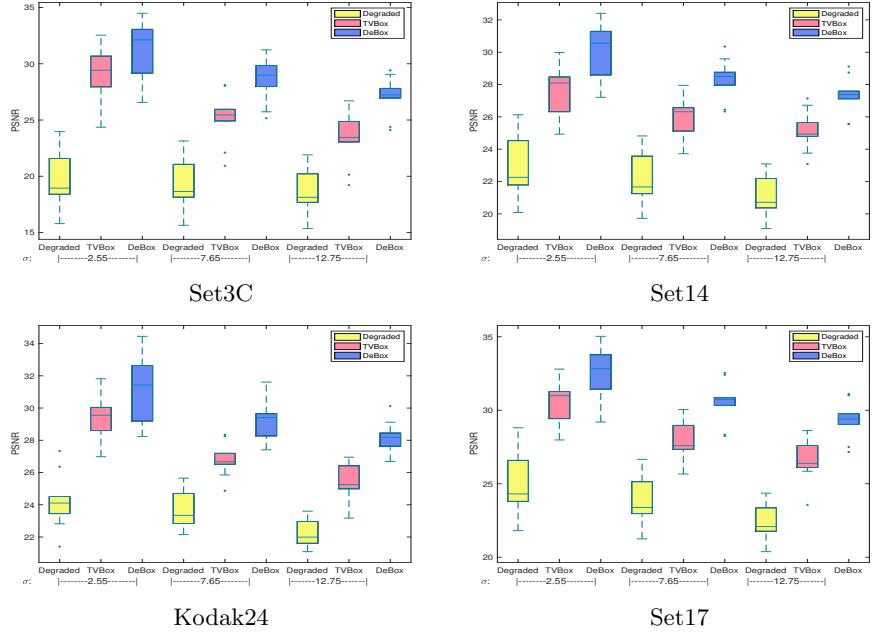


Figure 11. Average results ($PSNR(dB)$) of $TVBox$ and $DeBox$ for image deblurring with 10 different blur kernels and 3 noise levels on $Set3C$, $Set14$, $Kodak24$, and $Set17$ datasets.

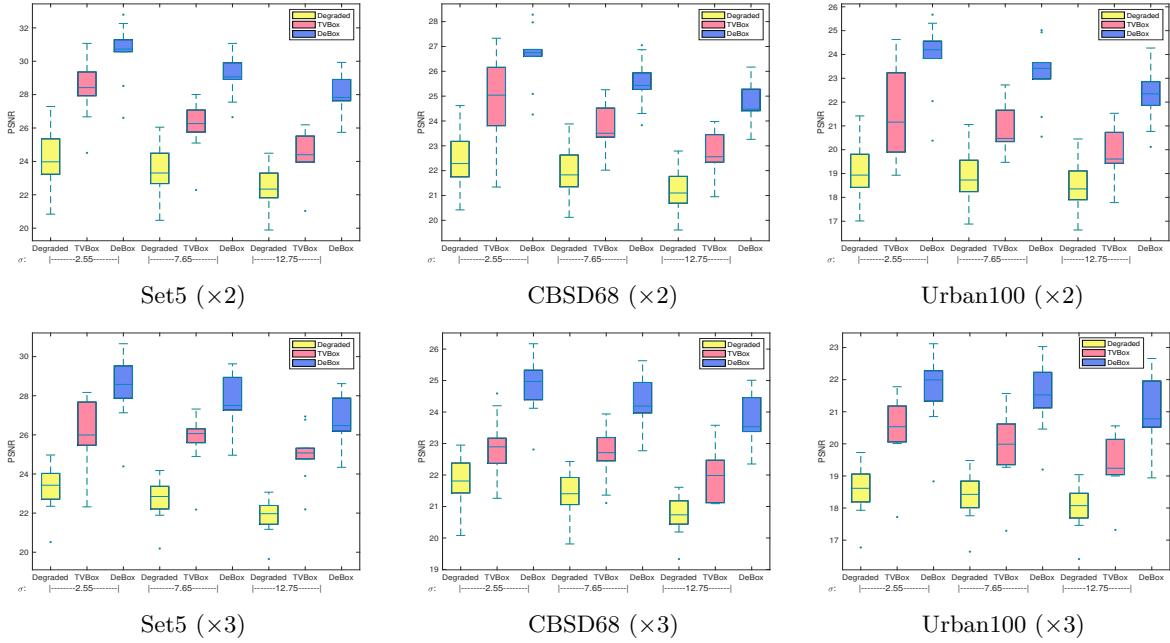


Figure 12. Average numerical results ($PSNR(dB)$) of $TVBox$ and $DeBox$ for image super-resolution with 2 scales ($\times 2$ and $\times 3$), 10 different blur kernels and 3 noise levels on $Set5$, $CBSD68$, and $Urban100$ datasets.

results of super-resolution under degradation with scale factors $\times 2$ and $\times 3$, respectively. The results presented in [Figure 10](#) also demonstrate that the proposed algorithm effectively solves the tested models, and DeTik outperforms TVTik in terms of recovery quality for image super-resolution.

For different noise levels and blur kernels, the average image restoration results of Set3C, Set14, Kodak24, and Set17 with box plot are demonstrated in [Figure 11](#). The yellow, pink, and blue boxes denote the average PSNR of the degraded images, the image restored by TVBox and DeBox. The first, second, and third sets of yellow, pink, and blue boxes correspond to the noise levels of 2.55, 7.65, and 12.75, respectively. Similarly, the super-resolution results for two scale factors, $\times 2$ and $\times 3$, are presented in [Figure 12](#). The result demonstrates that the proposed method exhibits consistent and stable image restoration performance. From [Figure 11](#) and [Figure 12](#), we can see that [Algorithm 4.2](#) effectively solves the DeBox model, and DeBox outperforms TVBox in terms of recovery quality for both image deblurring and super-resolution tasks. The experiment results also demonstrate that [Algorithm 4.2](#) can handle the minimization with the non-differentiable term.

Acknowledgement. The authors are grateful to the anonymous referees for their valuable comments, which largely improve the quality of this paper.

REFERENCES

- [1] M. AHOOKHOSH, A. THEMELIS, AND P. PATRINOS, *A Bregman forward-backward linesearch algorithm for nonconvex composite optimization: superlinear convergence to nonisolated local minima*, SIAM Journal on Optimization, 31 (2021), pp. 653–685.
- [2] H. ATTOUCH, J. BOLTE, P. REDONT, AND A. SOUBEYRAN, *Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality*, Mathematics of Operations Research, 35 (2010), pp. 438–457.
- [3] H. ATTOUCH, J. BOLTE, AND B. F. SVAITER, *Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods*, Mathematical Programming, 137 (2013), pp. 91–129.
- [4] H. ATTOUCH, J. PEYPOUQUET, AND P. REDONT, *A dynamical approach to an inertial forward-backward algorithm for convex minimization*, SIAM Journal on Optimization, 24 (2014), pp. 232–256.
- [5] A. BARAKAT AND P. BIANCHI, *Convergence rates of a momentum algorithm with bounded adaptive step size for nonconvex optimization*, in Asian Conference on Machine Learning, PMLR, 2020, pp. 225–240.
- [6] A. BECK AND M. TEBOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.
- [7] F. BIAN AND X. ZHANG, *A three-operator splitting algorithm for nonconvex sparsity regularization*, SIAM Journal on Scientific Computing, 43 (2021), pp. 2809–2839.
- [8] J. BOLTE, S. SABACH, AND M. TEBOULLE, *Proximal alternating linearized minimization for nonconvex and nonsmooth problems*, Mathematical Programming, 146 (2014), pp. 459–494.
- [9] R. I. BOT, E. R. CSETNEK, AND S. C. LÁSZLÓ, *An inertial forward–backward algorithm for the minimization of the sum of two nonconvex functions*, EURO Journal on Computational Optimization, 4 (2016), pp. 3–25.
- [10] G. T. BUZZARD, S. H. CHAN, S. SREEHARI, AND C. A. BOUMAN, *Plug-and-play unplugged: Optimization-free reconstruction using consensus equilibrium*, SIAM Journal on Imaging Sciences, 11 (2018), pp. 2001–2020.
- [11] C. CASTERA, J. BOLTE, C. FÉVOTTE, AND E. PAUWELS, *An inertial newton algorithm for deep learning*, The Journal of Machine Learning Research, 22 (2021), pp. 5977–6007.
- [12] C. CHEN, S. MA, AND J. YANG, *A general inertial proximal point algorithm for mixed variational inequality problem*, SIAM Journal on Optimization, 25 (2015), pp. 2120–2142.

- [13] R. COHEN, Y. BLAU, D. FREEDMAN, AND E. RIVLIN, *It has potential: Gradient-driven denoisers for convergent solutions to inverse problems*, Advances in Neural Information Processing Systems, 34 (2021), pp. 18152–18164.
- [14] P. L. COMBETTES AND J.-C. PESQUET, *Fixed point strategies in data science*, IEEE Transactions on Signal Processing, 69 (2021), pp. 3878–3905.
- [15] L. CONDAT, D. KITAHARA, A. CONTRERAS, AND A. HIRABAYASHI, *Proximal splitting algorithms for convex optimization: A tour of recent advances, with new twists*, SIAM Review, 65 (2023), pp. 375–435.
- [16] D. DAVIS AND W. YIN, *A three-operator splitting scheme and its optimization applications*, Set-Valued and Variational Analysis, 25 (2017), pp. 829–858.
- [17] L.-J. DENG, R. GLOWINSKI, AND X.-C. TAI, *A new operator splitting method for the Euler elastica model for image smoothing*, SIAM Journal on Imaging Sciences, 12 (2019), pp. 1190–1230.
- [18] J. DONG, S. ROTH, AND B. SCHIELE, *Deep wiener deconvolution: Wiener meets deep learning for image deblurring*, Advances in Neural Information Processing Systems, 33 (2020), pp. 1048–1059.
- [19] R. G. GAVASKAR, C. D. ATHALYE, AND K. N. CHAUDHURY, *On plug-and-play regularization using linear denoisers*, IEEE Transactions on Image Processing, 30 (2021), pp. 4802–4813.
- [20] T. GOLDSTEIN AND S. OSHER, *The split bregman method for l_1 -regularized problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 323–343.
- [21] K. GUO AND D. HAN, *A note on the Douglas–Rachford splitting method for optimization problems involving hypoconvex functions*, Journal of Global Optimization, 72 (2018), pp. 431–441.
- [22] K. GUO, D. HAN, AND X. YUAN, *Convergence analysis of Douglas–Rachford splitting method for “strongly + weakly” convex programming*, SIAM Journal on Numerical Analysis, 55 (2017), pp. 1549–1577.
- [23] D. HAN, *A survey on some recent developments of alternating direction method of multipliers*, Journal of the Operations Research Society of China, (2022), pp. 1–52.
- [24] J. HERTRICH, S. NEUMAYER, AND G. STEIDL, *Convolutional proximal neural networks and plug-and-play algorithms*, Linear Algebra and its Applications, 631 (2021), pp. 203–234.
- [25] S. HURAULT, A. CHAMBOLLE, A. LECLAIRE, AND N. PAPADAKIS, *Convergent Plug-and-Play with proximal denoiser and unconstrained regularization parameter*, arXiv preprint arXiv:2311.01216, (2023).
- [26] S. HURAULT, U. KAMILOV, A. LECLAIRE, AND N. PAPADAKIS, *Convergent Bregman plug-and-play image restoration for Poisson inverse problems*, arXiv preprint arXiv:2306.03466, (2023).
- [27] S. HURAULT, A. LECLAIRE, AND N. PAPADAKIS, *Gradient step denoiser for convergent plug-and-play*, in International Conference on Learning Representations (ICLR’22), 2022.
- [28] S. HURAULT, A. LECLAIRE, AND N. PAPADAKIS, *Proximal denoiser for convergent plug-and-play optimization with nonconvex regularization*, in International Conference on Machine Learning, PMLR, 2022, pp. 9483–9505.
- [29] P. JAIN, P. KAR, ET AL., *Non-convex optimization for machine learning*, Foundations and Trends® in Machine Learning, 10 (2017), pp. 142–363.
- [30] S. KONG, W. WANG, X. FENG, AND X. JIA, *Deep red unfolding network for image restoration*, IEEE Transactions on Image Processing, 31 (2022), pp. 852–867.
- [31] S. G. KRANTZ AND H. R. PARKS, *A primer of real analytic functions*, Springer Science & Business Media, 2002.
- [32] P. LATAFAT AND P. PATRINOS, *Asymmetric forward–backward–adjoint splitting for solving monotone inclusions involving three operators*, Computational Optimization and Applications, 68 (2017), pp. 57–93.
- [33] H. LE, N. GILLIS, AND P. PATRINOS, *Inertial block proximal methods for non-convex non-smooth optimization*, in International Conference on Machine Learning, PMLR, 2020, pp. 5671–5681.
- [34] G. LI AND T. K. PONG, *Douglas–Rachford splitting for nonconvex optimization with application to non-convex feasibility problems*, Mathematical Programming, 159 (2016), pp. 371–401.
- [35] J. LI, C. HUANG, R. CHAN, H. FENG, M. K. NG, AND T. ZENG, *Spherical image inpainting with frame transformation and data-driven prior deep networks*, SIAM Journal on Imaging Sciences, 16 (2023), pp. 1179–1196.
- [36] M. LI AND Z. WU, *Convergence analysis of the generalized splitting methods for a class of nonconvex optimization problems*, Journal of Optimization Theory and Applications, 183 (2019), pp. 535–565.

- [37] J. LIANG, J. FADILI, AND G. PEYRÉ, *A multi-step inertial forward-backward splitting method for non-convex optimization*, Advances in Neural Information Processing Systems, 29 (2016).
- [38] J. LIANG, J. FADILI, AND G. PEYRÉ, *Activity identification and local linear convergence of forward-backward-type methods*, SIAM Journal on Optimization, 27 (2017), pp. 408–437.
- [39] S. B. LINDSTROM AND B. SIMS, *Survey: sixty years of Douglas–Rachford*, Journal of the Australian Mathematical Society, 110 (2021), pp. 333–370.
- [40] H. LIU, X.-C. TAI, AND R. GLOWINSKI, *An operator-splitting method for the gaussian curvature regularization model with applications to surface smoothing and imaging*, SIAM Journal on Scientific Computing, 44 (2022), pp. A935–A963.
- [41] J. LIU, S. ASIF, B. WOHLBERG, AND U. KAMILOV, *Recovery analysis for plug-and-play priors using the restricted eigenvalue condition*, Advances in Neural Information Processing Systems, 34 (2021), pp. 5921–5933.
- [42] Y. LIU AND W. YIN, *An envelope for Davis–Yin splitting and strict saddle-point avoidance*, Journal of Optimization Theory and Applications, 181 (2019), pp. 567–587.
- [43] D. A. LORENZ AND T. POCK, *An inertial forward-backward algorithm for monotone inclusions*, Journal of Mathematical Imaging and Vision, 51 (2015), pp. 311–325.
- [44] Y. NESTEROV, *Introductory lectures on convex optimization: A basic course*, vol. 87, Springer Science & Business Media, 2003.
- [45] P. OCHS, Y. CHEN, T. BROX, AND T. POCK, *ipiano: Inertial proximal algorithm for nonconvex optimization*, SIAM Journal on Imaging Sciences, 7 (2014), pp. 1388–1419.
- [46] S. ONO, *Primal-dual plug-and-play image restoration*, IEEE Signal Processing Letters, 24 (2017), pp. 1108–1112.
- [47] J.-C. PESQUET, A. REPETTI, M. TERRIS, AND Y. WIAUX, *Learning maximally monotone operators for image recovery*, SIAM Journal on Imaging Sciences, 14 (2021), pp. 1206–1237.
- [48] D. N. PHAN AND N. GILLIS, *An inertial block majorization minimization framework for nonsmooth nonconvex optimization*, Journal of Machine Learning Research, 24 (2023), pp. 1–41.
- [49] T. POCK AND S. SABACH, *Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems*, SIAM Journal on Imaging Sciences, 9 (2016), pp. 1756–1787.
- [50] B. T. POLYAK, *Some methods of speeding up the convergence of iteration methods*, Ussr Computational Mathematics and Mathematical Physics, 4 (1964), pp. 1–17.
- [51] H. RAGUET, J. FADILI, AND G. PEYRÉ, *A generalized forward-backward splitting*, SIAM Journal on Imaging Sciences, 6 (2013), pp. 1199–1226.
- [52] E. T. REEHORST AND P. SCHNITER, *Regularization by denoising: Clarifications and new interpretations*, IEEE Transactions on Computational Imaging, 5 (2018), pp. 52–67.
- [53] R. T. ROCKAFELLAR AND R. J.-B. WETS, *Variational analysis*, vol. 317, Springer Science & Business Media, 2009.
- [54] L. I. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Physica D: Nonlinear Phenomena, 60 (1992), pp. 259–268.
- [55] E. RYU, J. LIU, S. WANG, X. CHEN, Z. WANG, AND W. YIN, *Plug-and-play methods provably converge with properly trained denoisers*, International Conference on Machine Learning, (2019), pp. 5546–5557.
- [56] E. K. RYU, A. B. TAYLOR, C. BERGELING, AND P. GISELSSON, *Operator splitting performance estimation: Tight contraction factors and optimal parameter selection*, SIAM Journal on Optimization, 30 (2020), pp. 2251–2271.
- [57] A. SALIM, L. CONDAT, K. MISHCHENKO, AND P. RICHTÁRIK, *Dualize, split, randomize: Toward fast nonsmooth optimization algorithms*, Journal of Optimization Theory and Applications, 195 (2022), pp. 102–130.
- [58] S. SETZER, *Operator splittings, bregman methods and frame shrinkage in image processing*, International Journal of Computer Vision, 92 (2011), pp. 265–280.
- [59] S. SREEHARI, S. V. VENKATKRISHNAN, B. WOHLBERG, G. T. BUZZARD, L. F. DRUMMY, J. P. SIMMONS, AND C. A. BOUMAN, *Plug-and-play priors for bright field electron tomography and sparse interpolation*, IEEE Transactions on Computational Imaging, 2 (2016), pp. 408–423.
- [60] Y. SUN, B. WOHLBERG, AND U. S. KAMILOV, *An online plug-and-play algorithm for regularized image reconstruction*, IEEE Transactions on Computational Imaging, 5 (2019), pp. 395–408.
- [61] Y. SUN, Z. WU, X. XU, B. WOHLBERG, AND U. S. KAMILOV, *Scalable plug-and-play ADMM with*

- convergence guarantees, *IEEE Transactions on Computational Imaging*, 7 (2021), pp. 849–863.
- [62] Y. TANG, M. WEN, AND T. ZENG, *Preconditioned three-operator splitting algorithm with applications to image restoration*, *Journal of Scientific Computing*, 92 (2022), pp. 1–26.
- [63] A. THEMELIS AND P. PATRINOS, *Douglas–Rachford splitting and ADMM for nonconvex optimization: Tight convergence results*, *SIAM Journal on Optimization*, 30 (2020), pp. 149–181.
- [64] A. THEMELIS, L. STELLA, AND P. PATRINOS, *Forward-backward envelope for the sum of two nonconvex functions: Further properties and nonmonotone linesearch algorithms*, *SIAM Journal on Optimization*, 28 (2018), pp. 2274–2303.
- [65] A. THEMELIS, L. STELLA, AND P. PATRINOS, *Douglas–Rachford splitting and ADMM for nonconvex optimization: accelerated and newton-type linesearch algorithms*, *Computational Optimization and Applications*, 82 (2022), pp. 395–440.
- [66] T. TIRER AND R. GIREYES, *Image restoration by iterative denoising and backward projections*, *IEEE Transactions on Image Processing*, 28 (2018), pp. 1220–1234.
- [67] S. V. VENKATAKRISHNAN, C. A. BOUMAN, AND B. WOHLBERG, *Plug-and-play priors for model based reconstruction*, in 2013 IEEE Global Conference on Signal and Information Processing, IEEE, 2013, pp. 945–948.
- [68] S. VILLA, S. SALZO, L. BALDASSARRE, AND A. VERRI, *Accelerated and inexact forward-backward algorithms*, *SIAM Journal on Optimization*, 23 (2013), pp. 1607–1633.
- [69] Q. WANG AND D. HAN, *A generalized inertial proximal alternating linearized minimization method for nonconvex nonsmooth problems*, *Applied Numerical Mathematics*, 189 (2023), pp. 66–87.
- [70] K. WEI, A. AVILES-RIVERO, J. LIANG, Y. FU, H. HUANG, AND C.-B. SCHÖNLIEB, *Tfpnp: Tuning-free plug-and-play proximal algorithms with applications to inverse imaging problems*, *The Journal of Machine Learning Research*, 23 (2022), pp. 699–746.
- [71] T. WU, W. WU, Y. YANG, F.-L. FAN, AND T. ZENG, *Retinex image enhancement based on sequential decomposition with a plug-and-play framework*, *IEEE Transactions on Neural Networks and Learning Systems*, (2023), pp. 1–14.
- [72] Z. WU, C. LI, M. LI, AND A. LIM, *Inertial proximal gradient methods with Bregman regularization for a class of nonconvex optimization problems*, *Journal of Global Optimization*, 79 (2021), pp. 617–644.
- [73] Z. WU AND M. LI, *General inertial proximal gradient method for a class of nonconvex nonsmooth optimization problems*, *Computational Optimization and Applications*, 73 (2019), pp. 129–158.
- [74] J. YANG AND Y. ZHANG, *Alternating direction algorithms for L_1 -problems in compressive sensing*, *SIAM Journal on Scientific Computing*, 33 (2011), pp. 250–278.
- [75] P. YIN, Y. LOU, Q. HE, AND J. XIN, *Minimization of L_1 - L_2 for compressed sensing*, *SIAM Journal on Scientific Computing*, 37 (2015), pp. 536–583.
- [76] A. YURTSEVER, V. MANGALICK, AND S. SRA, *Three operator splitting with a nonconvex loss function*, in International Conference on Machine Learning, PMLR, 2021, pp. 12267–12277.
- [77] J. ZENG, T. T.-K. LAU, S. LIN, AND Y. YAO, *Global convergence of block coordinate descent in deep learning*, in International Conference on Machine Learning, PMLR, 2019, pp. 7313–7323.
- [78] K. ZHANG, Y. LI, W. ZUO, L. ZHANG, L. VAN GOOL, AND R. TIMOFTE, *Plug-and-play image restoration with deep denoiser prior*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (2021), pp. 6360–6376.
- [79] K. ZHANG, L. VAN GOOL, AND R. TIMOFTE, *Deep unfolding network for image super-resolution*, in IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 3217–3226.
- [80] K. ZHANG, W. ZUO, S. GU, AND L. ZHANG, *Learning deep cnn denoiser prior for image restoration*, in IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3929–3938.