STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 0–14.
Published online in International Academic Press (www.IAPress.org)

A Curvature- and Descent-Based Hybrid Conjugate Gradient Algorithm for Image Denoising

El Mehdi BELLFKIH [1],[2,*], El Mahdi LAMAIZI [3], Imrane CHEMSEDDINE IDRISSI [1]

[1] *LTIM, Faculty of Sciences Ben M'Sick, Hassan II University, Casablanca, Morocco*
[2] *COPE, Centre for Educational Orientation and Planning, Rabat, Morocco*
[3] *LAGA, Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco*

Abstract A recently proposed refined conjugate-gradient update, known as BBP, speeds up the restoration of images damaged by impulse noise while matching the quality of older approaches, but it applies the same update rule at every step regardless of how the problem's local behaviour shifts as optimization proceeds. This paper does *not* claim a better algorithm; instead it studies, by controlled component analysis, what actually helps when one mixes BBP with the Hager–Zhang (HZ) update. We separate three ingredients – (i) hybridization of BBP and HZ, (ii) a data-dependent adaptive mixing weight recalculated every iteration from local curvature and descent indicators, and (iii) a sufficient-descent restart safeguard – and measure each in isolation. Tested on salt-and-pepper noise removal across three images and six noise levels, a *fixed* equal-weight hybrid ($\lambda_k \equiv 0.5$) is what delivers the benefit: it cuts function evaluations by about 31% relative to BBP alone (essentially tying HZ) at unchanged PSNR and SSIM. The proposed *adaptive* weighting rule does *not* improve on this fixed hybrid – on average it spends about 35% more function evaluations for no measurable quality gain – so we report it as a cautionary negative result and recommend the fixed weight on this benchmark. The safeguard almost never triggers here; its value is a correctness guarantee at negligible cost. On the theory side we give a corrected global-convergence analysis: with the safeguard enforcing both sufficient descent *and* a bounded direction scale, the iterates satisfy the asymptotic guarantee $\liminf_{k\to\infty} \|g_k\| = 0$ for any bounded mixing weight. No finite-time convergence rate is claimed.

Keywords Conjugate gradient method, Fixed and adaptive hybrid parameters, Hager–Zhang method, Descent property, Component analysis, Impulse noise removal, Image denoising.

AMS 2010 subject classifications 65F10, 65K05, 68U10, 90C30

DOI: 10.19139/soic-2310-5070-4564

1. Introduction

A standard way to restore an image corrupted by impulse noise is a two-phase pipeline: first a detector flags a candidate set of corrupted pixels, then a variational model restores just those pixels under an edge-preserving regularizer that has been smoothed enough to be minimized with gradient-based methods [11, 10]. Denoting by $x \in \mathbb{R}^n$ the vector of restored pixel values, this smoothed restoration objective takes the generic unconstrained form

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1}$$

Here f is constructed from a smooth edge-preserving potential $\varphi_\alpha(t) = \sqrt{\alpha + t^2}$, $\alpha > 0$, applied to the differences between each candidate pixel and its neighbours; the precise functional is given below, following [10, 11].

*Correspondence to: El Mehdi BELLFKIH (Email: elmehdi.bellfkih@gmail.com). LTIM, Faculty of Sciences Ben M'Sick, Hassan II University, Casablanca, Morocco. COPE, Centre for Educational Orientation and Planning, Rabat, Morocco

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 202x International Academic Press

Outside this variational, optimization-based line of work, denoising has been approached from several other directions. Comprehensive surveys track how the field has evolved [25, 31]. Non-local and patch-based schemes lean on the fact that similar patches recur throughout an image [26, 27], while sparse and transform-domain methods instead denoise by fitting the image to learned or fixed dictionaries [28, 29]. More recently, deep networks have pushed denoising performance substantially further [30]. This paper stays within the classical variational/optimization family, focusing specifically on its conjugate-gradient branch. We stress at the outset that our scope is the *optimizer*: we fix the energy model (7) and the first-stage impulse detector, and compare conjugate-gradient parameters for minimizing that fixed energy. We do not claim, and do not test, end-to-end denoising performance against non-local, dictionary, or deep methods; those operate under different objectives and detectors, and a fair comparison would require a different experimental protocol. The numbers reported here should be read as a study of solver behaviour on a fixed restoration energy, not as a benchmark of denoising quality across paradigms.

Nonlinear conjugate-gradient (CG) methods suit (1) particularly well since they need only gradient evaluations and carry $O(n)$ memory and per-iteration cost – a real concern once n is the pixel count of an image. The CG iterates take the form

$$x_{k+1} = x_k + \alpha_k d_k, \qquad d_{k+1} = -g_{k+1} + \beta_k d_k, \qquad d_0 = -g_0, \tag{2}$$

where $g_k = \nabla f(x_k)$ and α_k is a step length satisfying the strong Wolfe conditions

$$f(x_k + \alpha_k d_k) \le f(x_k) + \delta\, \alpha_k\, g_k^\top d_k, \tag{3}$$

$$d_k^\top g(x_k + \alpha_k d_k) \ge \sigma\, d_k^\top g_k, \tag{4}$$

with $0 < \delta < \sigma < 1$. A nonlinear CG method's whole behaviour is dictated by the scalar β_k. Well-known choices include Fletcher–Reeves (FR) [15] and Dai–Yuan (DY) [13],

$$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}, \qquad \beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^\top y_k}, \qquad y_k = g_{k+1} - g_k,$$

and the Hager–Zhang (HZ) parameter, which guarantees descent [16]

$$\beta_k^{HZ} = \frac{1}{d_k^\top y_k}\left(y_k - 2 d_k \frac{\|y_k\|^2}{d_k^\top y_k}\right)^\top g_{k+1}. \tag{5}$$

Starting from a second-order Taylor expansion of f together with a conjugacy condition $d_{k+1}^\top Q(x_k) s_k = 0$, where Q approximates the Hessian, F. Aiwa et al. [10] derived a refined parameter (denoted *BBP* below)

$$\beta_k^{BBP} = \left[\frac{(g_k^\top s_k)^2}{s_k^\top y_k \left(\frac{1}{2} y_k^\top s_k + (f_{k+1} - f_k) - g_k^\top s_k\right)}\right] \frac{\|g_{k+1}\|^2}{d_k^\top y_k}, \qquad s_k = x_{k+1} - x_k, \tag{6}$$

and showed the resulting search directions obey $d_{k+1}^\top g_{k+1} = \beta_k^{BBP}\, d_k^\top g_k$, giving sufficient descent plus a Zoutendijk-type global convergence property. Across salt-and-pepper denoising benchmarks, BBP reached FR's reconstruction quality (PSNR) while needing markedly fewer iterations (NI) and function evaluations (NF) [10].

BBP does not stand alone: it grows out of a broader effort to derive conjugate-gradient parameters from quadratic models and modified secant equations [2], a line that also includes single-variable secant-method variants [5, 6] and objective-value-based CG parameters following Nakama and Ito [18], whose β^{HY}/β^{B} family is what originally pointed toward BBP's quadratic-model derivation. A number of these ideas have already found their way into impulse-noise image restoration specifically [1, 3, 4, 7] and into hybrid nonlinear CG methods for image restoration more generally [8], next to other DY-type and spectral CG variants aimed at image denoising [21, 22]. Hybridizing two CG parameters through a convex combination, and protecting descent with a PR$^+$-type truncation plus a steepest-descent restart, are both well-established devices in this literature [8, 9, 20, 17]; we do not present either as

a new mechanism. Our interest is narrower and empirical: *given* that one hybridizes BBP and HZ on this restoration energy, does making the mixing weight adaptive buy anything over simply fixing it? The general convergence theory for nonlinear CG methods that underpins Section 3 comes from [12, 17, 19]; on the image-security side, work published in the same venue as [10] includes [24].

The motivation for pairing BBP with HZ specifically is that the two are built to be reliable in complementary regimes. BBP is derived from a quadratic Taylor model and a conjugacy condition, so it carries the most useful information when the last step produced a well-conditioned secant pair (s_k, y_k) – i.e. when the curvature estimate $s_k^\top y_k$ is large relative to $\|s_k\| \, \|y_k\|$. HZ, by contrast, was designed to retain a guaranteed descent direction even when the secant pair is poorly aligned and the previous direction is nearly orthogonal to the new gradient, a regime where quadratic-model parameters tend to be least trustworthy [16]. This is a hypothesis about complementarity, not a theorem; it is precisely what the adaptive rule below tries to exploit, and what the component analysis then puts to the test.

Yet what BBP lacks is adaptivity: it applies the same formula (6) at every iteration k regardless of how the local curvature, or the quality of the current descent direction, changes as optimization proceeds. It is well established that different nonlinear CG parameters behave differently under different problem conditioning, and that no single fixed choice of β_k wins on every iteration of every instance. That observation motivates the question this paper asks:

> *Rather than committing to one fixed strategy, can nonlinear conjugate-gradient behaviour be selected or combined dynamically, in response to the optimization landscape as it currently stands; and if so, does a parameter-light adaptive rule actually outperform a fixed mixture?*

We answer this question in the negative for the specific parameter-light rule we propose, and organize the paper around that finding. Concretely, the contributions are:

1. **A controlled component analysis.** We decompose a BBP–HZ hybrid into three separable ingredients – hybridization, an adaptive mixing weight, and a descent safeguard – and measure the effect of each in isolation across three images and six noise levels (Section 4). This is the backbone of the paper: it lets us attribute observed behaviour to a specific ingredient rather than to the fully assembled method.
2. **A negative result on adaptive weighting.** We instantiate an adaptive weight $\lambda_k \in [0, 1]$ computed online from a normalized curvature indicator ρ_k (built from s_k, y_k) and a normalized descent-quality indicator η_k (built from g_{k+1}, d_k), both drawn from quantities the iteration already produces (Section 2). We report, with a diagnostic analysis of why, that this rule does *not* beat a fixed equal-weight hybrid on this benchmark: it costs about 35% more function evaluations at matched quality. The practical gain we observe comes from *hybridization*, not from *adaptivity*.
3. **A corrected global-convergence analysis.** We prove sufficient descent and the asymptotic guarantee $\liminf_{k\to\infty} \|g_k\| = 0$ (Section 3). Relative to a naive argument, the safeguard here is strengthened to enforce *both* a sufficient-descent test *and* a bounded direction scale $\|d_{k+1}\| \le M \, \|g_{k+1}\|$; the latter is exactly what a descent condition alone fails to supply, and what the global-convergence step genuinely needs. The guarantee is asymptotic and carries no finite-time rate.

The descent safeguard itself is a standard restart mechanism; we keep it, and prove convergence with it, but we do not claim it as a contribution.

The rest of the paper unfolds as follows. The remainder of this section recalls the impulse-noise restoration model in detail. Section 2 then derives the adaptive hybrid parameter and presents the descent safeguard together with the complete algorithm; Section 3 carries out the convergence analysis; Section 4 reports the numerical experiments together with the component analysis; and Section 5 concludes.

We build on the two-phase edge-preserving restoration model of [11, 10]. Let $y \in \mathbb{R}^n$ denote the observed image, and let

$$N = \{(i, j) : y_{ij} \in \{0, 255\}\}$$

be the candidate set flagged by the extreme-value impulse detector, the usual first-stage detector for salt-and-pepper noise, since an impulse overwrites a pixel value entirely with 0 or 255. Optimization variables are drawn only from

N; every pixel outside N keeps whatever value was observed. With W_{ij} denoting the 8-connected neighbourhood of pixel (i,j) and $\varphi_\alpha(t) = \sqrt{\alpha + t^2}$ the smooth edge-preserving potential ($\alpha > 0$), the restored image u minimizes

$$f(u) = \sum_{(i,j)\in N} \sum_{(k,l)\in W_{ij}} c_{kl}\, \varphi_\alpha\big(u_{ij} - u_{kl}\big), \qquad c_{kl} = \begin{cases} 2, & (k,l) \notin N \\ 1, & (k,l) \in N, \end{cases} \tag{7}$$

Differentiating (7) with respect to u_{ij} for $(i,j) \in N$, and exploiting the symmetry of the 8-connected neighbourhood relation, yields the closed-form gradient

$$\frac{\partial f}{\partial u_{ij}} = 2 \sum_{(k,l)\in W_{ij}} \varphi'_\alpha\big(u_{ij} - u_{kl}\big), \qquad \varphi'_\alpha(t) = \frac{t}{\sqrt{\alpha + t^2}}. \tag{8}$$

Fix $(i,j) \in N$ and differentiate f with respect to u_{ij}. Two kinds of terms in (7)'s double sum depend on u_{ij}: (A) terms in the inner sum for $(p,q) = (i,j)$ itself, and (B) terms in the inner sum of any other $(p,q) \in N$ with $(i,j) \in W_{pq}$; by symmetry of the 8-connected neighbourhood relation, (B) is exactly the set $W_{ij} \cap N$.

Contribution (A).

$$\frac{\partial}{\partial u_{ij}} \sum_{(k,l)\in W_{ij}} c_{kl}\, \varphi_\alpha(u_{ij} - u_{kl}) = \sum_{(k,l)\in W_{ij}\setminus N} 2\, \varphi'_\alpha(u_{ij} - u_{kl}) + \sum_{(k,l)\in W_{ij}\cap N} \varphi'_\alpha(u_{ij} - u_{kl}).$$

Contribution (B). For each $(k,l) \in W_{ij} \cap N$, the inner sum of $(p,q) = (k,l)$ contains the term $c_{ij}\, \varphi_\alpha(u_{kl} - u_{ij}) = \varphi_\alpha(u_{kl} - u_{ij})$ (since $(i,j) \in N$ forces $c_{ij} = 1$); applying $\varphi'_\alpha(-t) = -\varphi'_\alpha(t)$ gives

$$\frac{\partial}{\partial u_{ij}} \varphi_\alpha(u_{kl} - u_{ij}) = -\varphi'_\alpha(u_{kl} - u_{ij}) = \varphi'_\alpha(u_{ij} - u_{kl}),$$

summed over $(k,l) \in W_{ij} \cap N$.

Summing (A) and (B) together, each $W_{ij} \cap N$ term picks up coefficient $1 + 1 = 2$, which matches the coefficient 2 already carried by the $W_{ij} \setminus N$ terms from (A) – giving (8).

Computing Equation (8) takes $O(n)$ time via eight shifted-array operations, and it stays unmodified across every CG variant compared here – what differs between methods is purely the choice of β_k, and hence the search direction. Two structural properties of this energy are worth recording for the analysis in Section 3. First, φ_α is convex and f is a sum of compositions of φ_α with linear maps, so f is convex; every stationary point is therefore a global minimizer, and the guarantee $\liminf_k \|g_k\| = 0$ translates into convergence of the objective to its global minimum value. Second, $\varphi''_\alpha(t) = \alpha/(\alpha + t^2)^{3/2} \le \alpha^{-1/2}$ is bounded, so ∇f is globally Lipschitz with a constant proportional to $\alpha^{-1/2}$ and to the (bounded) number of neighbour couplings per pixel. These facts place Assumption 1 on firm ground for this specific objective, as discussed in Section 3.

2. Adaptive Hybrid Parameter

Instead of committing to β_k^{BBP} at every step, we propose

$$\boxed{\beta_k^{AH} = \lambda_k \beta_k^{BBP} + (1 - \lambda_k)\beta_k^{HZ}}, \qquad 0 \le \lambda_k \le 1, \tag{9}$$

so $\lambda_k \approx 1$ recovers BBP behaviour, $\lambda_k \approx 0$ recovers HZ behaviour, and values in between interpolate. Rather than fixing λ_k in advance, we compute it online from two indicators built from quantities already on hand at iteration k. The two indicators are designed to track the complementarity hypothesis stated in Section 1: ρ_k large should favour BBP, η_k large should favour HZ.

To measure local curvature, write $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$; then

$$\rho_k = \frac{|s_k^\top y_k|}{\|s_k\|\ \|y_k\| + \epsilon}, \qquad \epsilon > 0, \tag{10}$$

gives a normalized measure of the alignment between s_k and y_k, i.e. how trustworthy the curvature information from the last step is: a large ρ_k means the secant pair carries strong, well-conditioned curvature information, precisely the regime where BBP – built from a quadratic model – should perform best.

To measure descent quality, let d_k be the previous search direction;

$$\eta_k = \frac{|g_{k+1}^\top d_k|}{\|g_{k+1}\| \, \|d_k\| + \epsilon} \tag{11}$$

quantifies how nearly orthogonal the new gradient is to the previous direction. Since HZ is built precisely to keep descent robust even when this alignment is weak [16], a large η_k counts as evidence in HZ's favour.

Combining both indicators with a parameter-light rule then gives

$$\boxed{\lambda_k = \frac{\rho_k}{\rho_k + \eta_k + \epsilon}} \tag{12}$$

which, by construction, guarantees $\lambda_k \in [0, 1]$ for every k (Proposition 1). Plugging (12) into (9) produces the fully explicit adaptive hybrid coefficient

$$\beta_k^{AH} = \frac{\rho_k}{\rho_k + \eta_k + \epsilon} \beta_k^{BBP} + \frac{\eta_k}{\rho_k + \eta_k + \epsilon} \beta_k^{HZ}. \tag{13}$$

Both ρ_k and η_k reuse s_k, y_k, g_{k+1} and d_k – quantities already computed while evaluating β_k^{BBP} and β_k^{HZ} – so (12) only adds $O(n)$ work per iteration (two inner products, two norms), a negligible amount next to the $O(n)$ already spent on the line search and gradient evaluation. We treat Equation (12) as a research hypothesis for a parameter-light adaptive rule rather than a provably optimal weighting; it is tested empirically in Section 4, with its limitations discussed in Section 4.3. We deliberately keep the rule parameter-light – a single instantaneous ratio of two single-step indicators – precisely so that the component analysis isolates the value of *instantaneous* adaptivity, with no confound from tuning or smoothing. Richer variants (history-averaged or exponentially smoothed indicators, or optimization-based weight selection) are plausible and are left as explicit future work in Section 5; the negative result below concerns this specific instantaneous rule, not adaptivity in principle.

Since β_k^{AH} is a data-dependent convex combination of two distinct formulas, it does not automatically inherit BBP's descent guarantee: in principle, hybridizing the two could yield a d_{k+1} that fails to be a descent direction, or a direction whose scale is badly mismatched to $\|g_{k+1}\|$. To rule both out, we impose an explicit two-part safeguard: given a small $\xi \in (0, 1)$ and a constant $M \geq 1$, we require

$$g_{k+1}^\top d_{k+1} \leq -\xi \, \|g_{k+1}\|^2 \tag{14}$$

and

$$\|d_{k+1}\| \leq M \, \|g_{k+1}\| \, . \tag{15}$$

Whenever (14) or (15) fails, we drop the hybrid direction and restart along steepest descent,

$$d_{k+1} = -g_{k+1}. \tag{16}$$

Because $g_{k+1}^\top(-g_{k+1}) = -\|g_{k+1}\|^2 \leq -\xi \|g_{k+1}\|^2$ holds for any $\xi \in (0, 1]$, and $\|-g_{k+1}\| = \|g_{k+1}\| \leq M \|g_{k+1}\|$ holds for any $M \geq 1$, the restarted direction always satisfies (14) and (15) – so by construction both hold at *every* iteration, regardless of how β_k^{AH} behaves. The descent test (14) is what supplies sufficient descent; the scale test (15) is what bounds the search-direction norm relative to the gradient, and – as Section 3 makes precise – it is this second bound, absent from a descent-only safeguard, that the global-convergence argument actually requires. It is exactly this mechanism that lets Section 3's global convergence analysis go through without any extra assumption on λ_k.

Putting these pieces together gives the complete procedure:

Algorithm 1 Hybrid BBP–HZ Conjugate Gradient with descent–scale safeguard (fixed or adaptive weight)

Require: $x_0 \in \mathbb{R}^n$, $\alpha > 0$ (potential parameter), $\xi, \epsilon > 0$ small, $M \geq 1$, $0 < \delta < \sigma < 1$, max_iter

1: $g_0 \leftarrow \nabla f(x_0)$; $\quad d_0 \leftarrow -g_0$; $\quad k \leftarrow 0$
2: **while** not converged and $k <$ max_iter **do**
3: Find α_k satisfying the Wolfe conditions (3)–(4) along d_k
4: $x_{k+1} \leftarrow x_k + \alpha_k d_k$; $\quad g_{k+1} \leftarrow \nabla f(x_{k+1})$
5: $s_k \leftarrow x_{k+1} - x_k$; $\quad y_k \leftarrow g_{k+1} - g_k$
6: Compute β_k^{BBP} (6) and β_k^{HZ} (5) (set either to 0 if its denominator is below a threshold ϵ_0)
7: Set λ_k: either the fixed value $\lambda_k \equiv \frac{1}{2}$, or the adaptive rule ρ_k (10), η_k (11), λ_k (12)
8: $\beta_k^{AH} \leftarrow \lambda_k \beta_k^{BBP} + (1 - \lambda_k)\beta_k^{HZ}$; $\quad \beta_k^{AH} \leftarrow \max(\beta_k^{AH}, 0)$
9: $d_{k+1} \leftarrow -g_{k+1} + \beta_k^{AH} d_k$
10: **if** $g_{k+1}^\top d_{k+1} > -\xi \|g_{k+1}\|^2$ **or** $\|d_{k+1}\| > M \|g_{k+1}\|$ **then** ▷ descent–scale safeguard
11: $d_{k+1} \leftarrow -g_{k+1}$ ▷ restart
12: **end if**
13: $k \leftarrow k + 1$
14: **end while**
15: **return** x_k

Before the direction is formed, line 8 also truncates β_k^{AH} to be nonnegative, a standard PR$^+$-type safeguard [20], which is a minor numerical measure with no bearing on the convergence analysis, since that analysis relies only on (14)–(15) via –(16). Two implementation details matter for reproducibility and for the technical objections a careful reader will raise. (i) The truncation $\beta_k^{AH} \leftarrow \max(\beta_k^{AH}, 0)$ can only shrink $\|d_{k+1}\|$ and can only make $g_{k+1}^\top d_{k+1}$ more negative when d_k is already a descent direction; in any case the convergence proof does not use any algebraic property inherited from BBP or HZ – it uses only the two safeguard inequalities (14)–(15), which hold *after* truncation and the restart test. So truncation changes the realized iterate but not the guarantee. (ii) Both β_k^{BBP} and β_k^{HZ} carry denominators – $s_k^\top y_k$, $d_k^\top y_k$, and the bracketed quadratic-model term in (6) – that can approach zero or change sign. We guard every such denominator: if its absolute value falls below a small threshold ϵ_0 (we use $\epsilon_0 = 10^{-12}$), the corresponding parameter is set to 0, which reduces the step to (a component of) steepest descent and is then caught, if necessary, by the restart test. This makes the iteration well defined for all inputs and is consistent with the theory, since a zeroed parameter only strengthens descent. We declare convergence once both $|f(x_k) - f(x_{k-1})|/|f(x_k)| \leq \tau$ and $\|g(x_k)\| \leq \tau(1 + |f(x_k)|)$ hold for a tolerance τ, following [10].

3. Convergence Analysis

We adopt the same standing assumptions as [10], carried over from [23, 16].

Assumption 1 (a) On $\Omega = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$, f is bounded below.
(b) ∇f is Lipschitz continuous on an open set containing Ω: there exists $L > 0$ with $\|g(u) - g(v)\| \leq L \|u - v\|$ for all u, v in that set.

Remark 1 (Plausibility of Assumption 1 for the denoising energy). Both parts hold for the specific objective (7). Since $\varphi_\alpha \geq 0$, f is bounded below by 0, giving (a). As noted after (8), $\varphi_\alpha''(t) \leq \alpha^{-1/2}$ is bounded and each variable couples to a bounded number of neighbours, so ∇f is globally Lipschitz with $L = O(\alpha^{-1/2})$, giving (b) with an explicit constant. Thus the assumptions used below are not merely borrowed boilerplate; they are satisfied here with a computable L, and f is in fact convex, so the limit point is a global minimizer. What the assumptions do *not* furnish is strong convexity (the Hessian of φ_α degenerates as $|t| \to \infty$), which is why the guarantee below is asymptotic rather than a linear rate.

Proposition 1 (Boundedness of the adaptive weight)
For every k with $\rho_k, \eta_k \geq 0$ and $\epsilon > 0$, the weight λ_k defined by (12) satisfies $\lambda_k \in [0, 1]$.

Proof
By definition, $\rho_k \geq 0$ and $\eta_k \geq 0$ (the numerators of (10)–(11) are absolute values), while the denominator satisfies $\rho_k + \eta_k + \epsilon \geq \epsilon > 0$. Consequently λ_k is the ratio of a nonnegative number to a strictly larger one, so $\lambda_k \in [0, 1) \subset [0, 1]$, with $\lambda_k = 0$ occurring precisely when $\rho_k = 0$. □

Theorem 1 (Sufficient descent)
Algorithm 1 generates directions satisfying $g_k^\top d_k \leq -\xi \|g_k\|^2 < 0$ for every $k \geq 0$, and in addition $\|d_k\| \leq M \|g_k\|$ for every $k \geq 0$.

Proof
At $k = 0$ we have $d_0 = -g_0$, giving $g_0^\top d_0 = -\|g_0\|^2 \leq -\xi \|g_0\|^2$ since $0 < \xi \leq 1$, and $\|d_0\| = \|g_0\| \leq M \|g_0\|$ since $M \geq 1$. For $k \geq 1$, two cases arise: either the safeguard at line 10–12 fired, so $d_k = -g_k$ and the same computation applies unchanged, or it did not fire, meaning d_k already passed both tests at line 10, i.e. $g_k^\top d_k \leq -\xi \|g_k\|^2$ and $\|d_k\| \leq M \|g_k\|$. Either way both claimed inequalities hold, and since $\|g_k\| > 0$ whenever x_k is not stationary, it follows that $g_k^\top d_k < 0$. □

Lemma 1 (Zoutendijk-type condition)
Under Assumption 1, suppose α_k satisfies the Wolfe conditions (3)–(4) and d_k is a descent direction for every k. Then

$$\sum_{k=0}^{\infty} \frac{(g_k^\top d_k)^2}{\|d_k\|^2} < \infty. \tag{17}$$

Proof
This is the standard Zoutendijk inequality [23, 19]. Combining (4) with the Lipschitz continuity of g gives $(\sigma - 1) g_k^\top d_k \leq (g_{k+1} - g_k)^\top d_k \leq L \alpha_k \|d_k\|^2$, hence $\alpha_k \geq \dfrac{(\sigma - 1) g_k^\top d_k}{L \|d_k\|^2}$. Substituting this bound into (3) and using $g_k^\top d_k < 0$ yields $f(x_k) - f(x_{k+1}) \geq c\,(g_k^\top d_k)^2 / \|d_k\|^2$, where the constant $c > 0$ depends only on δ, σ and L. Summing over k, and invoking the lower boundedness of f on Ω (Assumption 1(a)), gives (17). □

Theorem 2 (Global convergence)
Under Assumption 1, the iterates generated by Algorithm 1 satisfy

$$\liminf_{k \to \infty} \|g_k\| = 0. \tag{18}$$

Proof
Theorem 1 guarantees that d_k is a descent direction at every iteration, so Lemma 1 applies. Assume, for contradiction, that (18) does not hold; then some $r > 0$ satisfies $\|g_k\| \geq r$ for all k. By the sufficient-descent bound of Theorem 1, $(g_k^\top d_k)^2 \geq \xi^2 \|g_k\|^4$; and by the *scale* bound of the same theorem, $\|d_k\|^2 \leq M^2 \|g_k\|^2$. Dividing the first by the second,

$$\frac{(g_k^\top d_k)^2}{\|d_k\|^2} \;\geq\; \frac{\xi^2 \|g_k\|^4}{M^2 \|g_k\|^2} \;=\; \frac{\xi^2}{M^2} \|g_k\|^2 \;\geq\; \frac{\xi^2}{M^2} r^2 \;>\; 0 \qquad \text{for every } k.$$

The series (17) would then have terms bounded below by the fixed positive constant $\xi^2 r^2 / M^2 > 0$, and so could not converge – contradicting Lemma 1. Therefore (18) must hold. □

Remark 2 (What the scale bound repairs). The crux of the argument is that the Zoutendijk term $(g_k^\top d_k)^2 / \|d_k\|^2$ must be bounded *below* by a positive constant. Sufficient descent alone controls the numerator but says nothing about the denominator: a descent condition $g_k^\top d_k \leq -\xi \|g_k\|^2$ permits $\|d_k\|$ to grow without bound relative to $\|g_k\|$, in which case the quotient can tend to 0 and no contradiction follows. What is needed is an *upper* bound on $\|d_k\|$ relative to $\|g_k\|$; this is precisely what the scale test (15) enforces, via $\|d_k\| \leq M \|g_k\|$. A descent-only safeguard does not supply it, which is the gap this version closes.

Remark 3. Theorem 2 needs no algebraic property of β_k^{AH} – it holds for *any* bounded, measurable $\lambda_k \in [0,1]$, *provided the descent–scale safeguard of Section 2 is in force*, because that safeguard enforces sufficient descent and a bounded direction scale directly. In this precise sense, the safeguard decouples the adaptivity of λ_k from the global convergence guarantee: λ_k can be an arbitrary heuristic (12), or any future refinement – without threatening Theorem 2. We emphasize that this is a statement *under the safeguard*, not a claim that convergence holds for any bounded weight with a descent-only safeguard; the earlier phrasing to that effect was too strong, and (15) is exactly the extra condition that makes it correct.

Remark 4 (Scope of the guarantee). The result (18) is asymptotic: it states that a subsequence of gradients vanishes, not that any prescribed accuracy is reached in a given number of iterations. We claim no finite-time convergence rate. For the convex energy (7) the limit point is a global minimizer, but turning $\liminf_k \|g_k\| = 0$ into a quantitative rate would require additional structure (e.g. a Polyak–Łojasiewicz or local strong-convexity condition) that this degenerate-at-infinity potential does not provide; we leave a rate analysis as future work.

4. Numerical Experiments

4.1. Setup

Every method is implemented from scratch in Python/NumPy, sharing the same objective (7)–(8), the same strong-Wolfe line search ($\delta = 10^{-4}$, $\sigma = 0.4$), the same stopping tolerance $\tau = 10^{-4}$ (Section 2), and the same 400-iteration budget. Seven configurations are compared: FR [15], HZ [16], BBP [10], a non-adaptive fixed hybrid ($\lambda_k \equiv 0.5$, no safeguard), that same fixed hybrid with the safeguard switched on, the adaptive hybrid (13) without the safeguard, and finally the adaptive hybrid *with* the safeguard (Algorithm 1).

Every experiment ran on personal laptop carrying a 13th Gen Intel® Core™ i7-1355U CPU and 32 GB RAM, under Windows 11 Pro (64-bit), using Python 3.13.2, NumPy 2.4.6, SciPy 1.18.0 and scikit-image 0.26.0.

Reproducibility. For completeness we report every numerical constant used. The edge-preserving potential parameter is $\alpha = 1.0$ in (7). The line-search constants are $\delta = 10^{-4}$, $\sigma = 0.4$ (strong Wolfe). The small constant in the indicators (10)–(11) is $\epsilon = 10^{-8}$; the denominator guard in the β formulas is $\epsilon_0 = 10^{-12}$. The descent-safeguard threshold is $\xi = 10^{-4}$; the direction-scale threshold is $M = 10^3$ (a deliberately loose value chosen so that (15) acts only as a theoretical safety net). The iteration budget is 400 and the convergence tolerance is $\tau = 10^{-4}$. Noise realizations are generated with NumPy's `default_rng`; the master seed is 20260823, and the seed for each (image, noise) cell is derived deterministically as $20260823 + 1000 \cdot (\text{image index}) + \lfloor 100 \cdot (\text{noise ratio}) \rfloor$, so every reported number is reproducible run to run. We report results for a *single* noise realization per cell (one seed); we did not perform repeated-seed trials, and we therefore report no variances, confidence intervals, or significance tests. This is a genuine limitation of the present study – addressed explicitly in Section 5 – and the small PSNR/SSIM differences between methods below should be read as "indistinguishable at this single-realization resolution" rather than as tested equalities. The reference implementation and the scripts that regenerate every table and figure are released with the paper.

For test data, we use three classic grayscale 512×512 images from the USC-SIPI image database [32], downsampled to the same 256×256 working resolution: the *Milkdrop* coronet image, the *Mandrill* (baboon) image, and the *Man* portrait image. On the noise side, we apply salt-and-pepper noise at ratios $10\%, 20\%, 30\%, 50\%, 70\%, 90\%$, extending the $50\%/70\%/90\%$ range of [10] down into lighter noise regimes, as this line of work's roadmap suggests. We acknowledge that three grayscale images at one resolution, one noise type, and one fixed detector is a small benchmark; the conclusions below are stated for this benchmark, and their generalization to color images, larger and more varied image sets, other noise models, and alternative detectors is listed as future work in Section 5.

For evaluation, we track the number of iterations (NI), the number of function evaluations (NF), the wall-clock runtime, PSNR

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{MN} \sum (u_{ij}^r - u_{ij}^*)^2} \tag{19}$$

The structural similarity index (SSIM) between the restored image u^r and the original clean image u^*. We treat NF as the primary machine-independent cost measure, since the dominant per-iteration cost is the function/gradient evaluations requested by the line search, but we also report wall-clock time (Table 3) for transparency. Wall-clock time has noticeable run-to-run variance on a laptop and tracks NF closely; it should not be used to separate methods with near-identical NF.

4.2. Results

For every test image and noise level, Table 2 reports NI, NF, PSNR and SSIM across all seven configurations, while Table 1 averages those same results over the three test images by noise level. Dolan–Moré performance profiles [14] for NI and NF, over the six core configurations (those of Section 4, leaving out the fixed-hybrid safeguard variant analysed separately in the component analysis), appear in Figures 1 and 1. Figures 3–5 then give a visual comparison for each test image, one row per noise level (30%, 50%, 70%) stacked top to bottom.

Table 1. Average performance over all three test images, by noise level.

Noise (%)	FR				HZ				BBP				Fixed BBP–HZ				Adaptive BBP–HZ				**Adaptive BBP–HZ + Safeguard**			
	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM
10	331.7	1671.7	39.40	0.9857	92.0	210.7	39.31	0.9857	172.3	819.3	39.40	0.9856	94.7	216.7	39.34	0.9857	118.0	343.0	39.36	0.9856	118.0	343.0	39.36	0.9856
20	400.0	2349.0	36.11	0.9708	118.7	267.3	35.99	0.9708	134.3	446.3	36.05	0.9708	122.3	279.3	36.06	0.9708	104.3	286.3	36.00	0.9707	104.3	286.3	36.00	0.9707
30	400.0	2458.3	33.59	0.9518	132.3	297.7	33.51	0.9517	137.0	353.3	33.50	0.9516	135.3	320.0	33.54	0.9517	118.0	286.0	33.50	0.9516	118.0	286.0	33.50	0.9516
50	400.0	2705.3	30.34	0.8985	168.7	374.0	30.33	0.8995	150.0	332.7	30.33	0.8994	148.3	323.3	30.34	0.8995	145.3	405.7	30.33	0.8995	145.3	405.7	30.33	0.8995
70	400.0	2688.3	26.51	0.7884	248.7	540.3	26.57	0.8073	226.7	610.3	26.60	0.8075	241.3	550.3	26.58	0.8074	255.3	850.3	26.60	0.8075	255.0	846.0	26.60	0.8075
90	400.0	2612.0	22.16	0.5547	359.7	793.0	22.06	0.5957	364.7	1000.0	22.06	0.5958	346.3	762.7	22.06	0.5958	373.0	1153.0	22.06	0.5958	369.7	1153.0	22.06	0.5958

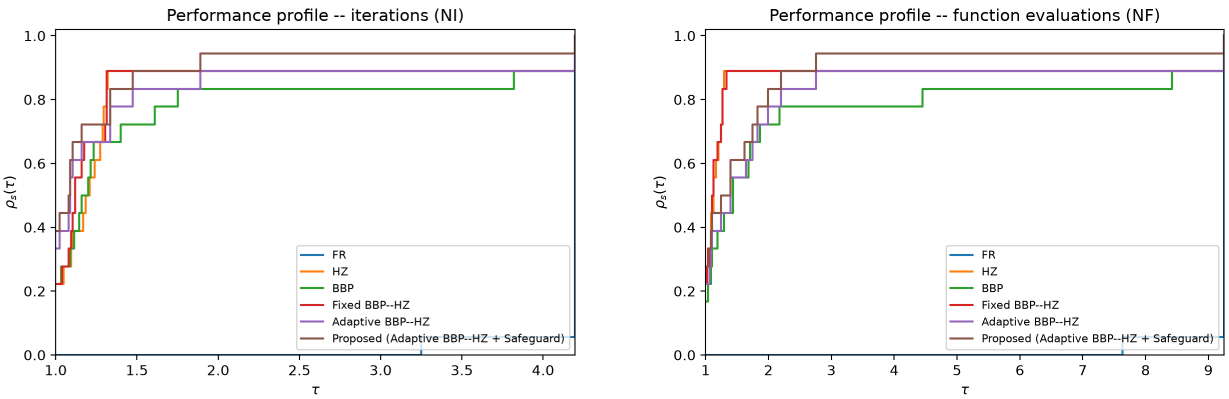


Figure 1. Dolan–Moré performance profiles across all (image, noise level) problem instances: iterations (NI) on the left, function evaluations (NF) on the right.

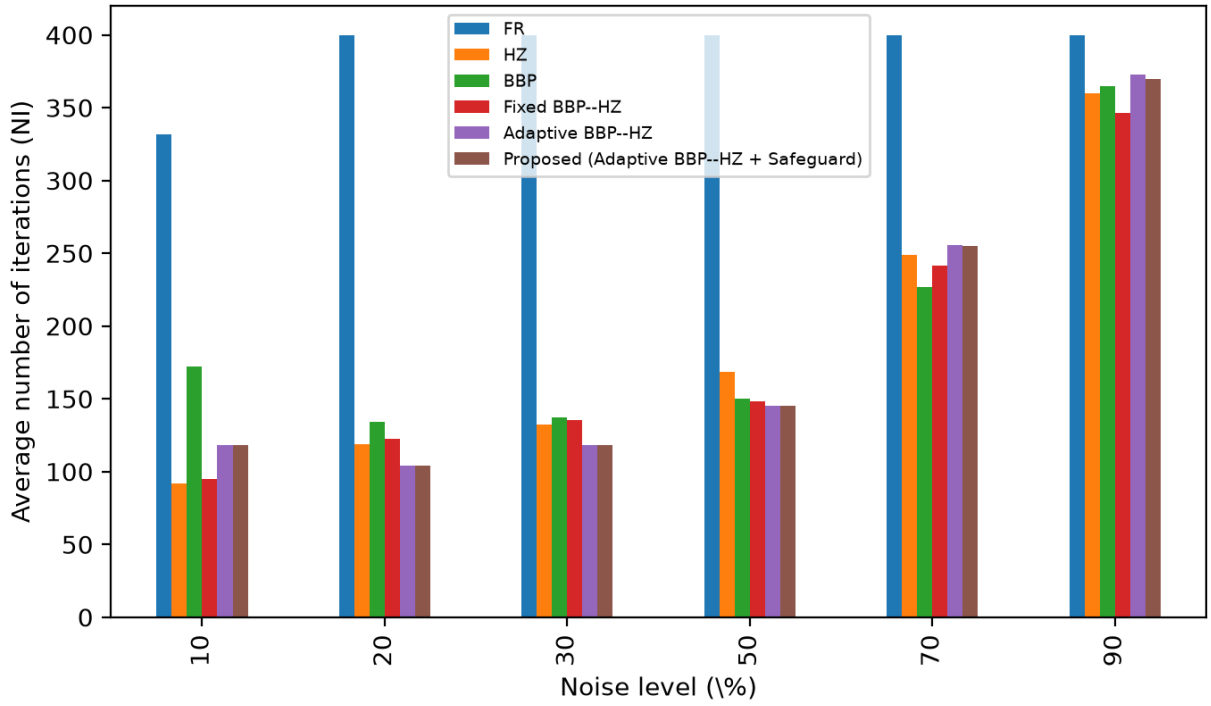


Figure 2. Average number of iterations (NI) by noise level, for the six core configurations, averaged over all three test images.

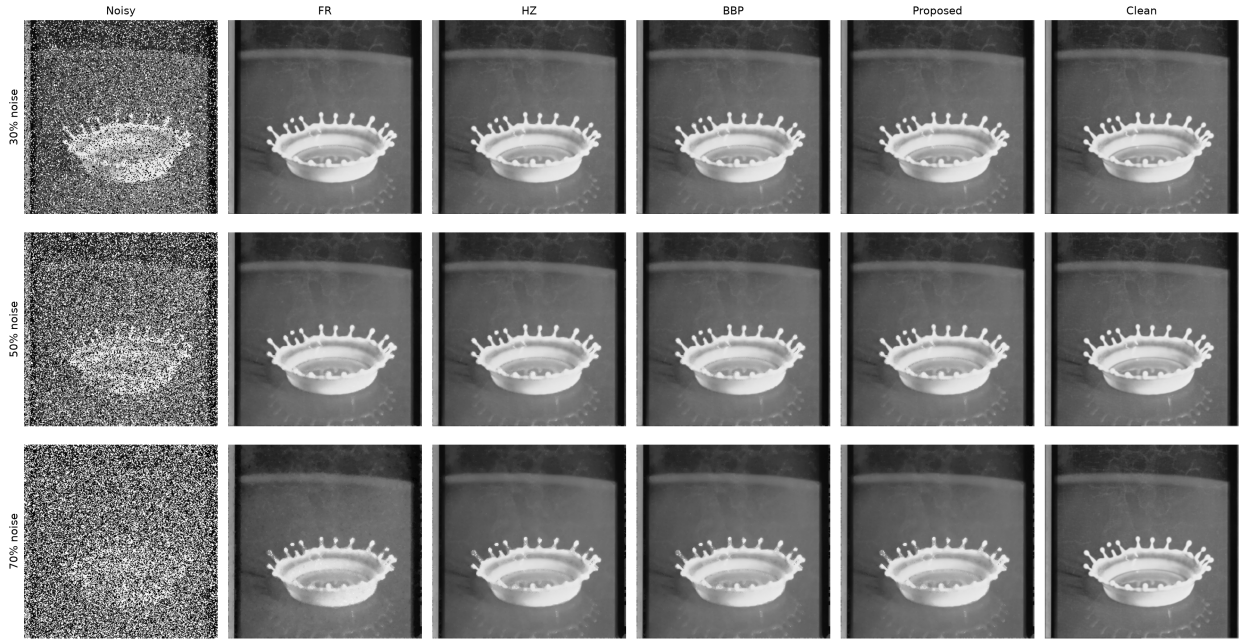


Figure 3. Visual comparison on the Milkdrop image at 30%, 50% and 70% salt-and-pepper noise: noisy input, FR, HZ, BBP, the adaptive hybrid with safeguard, and the clean reference.

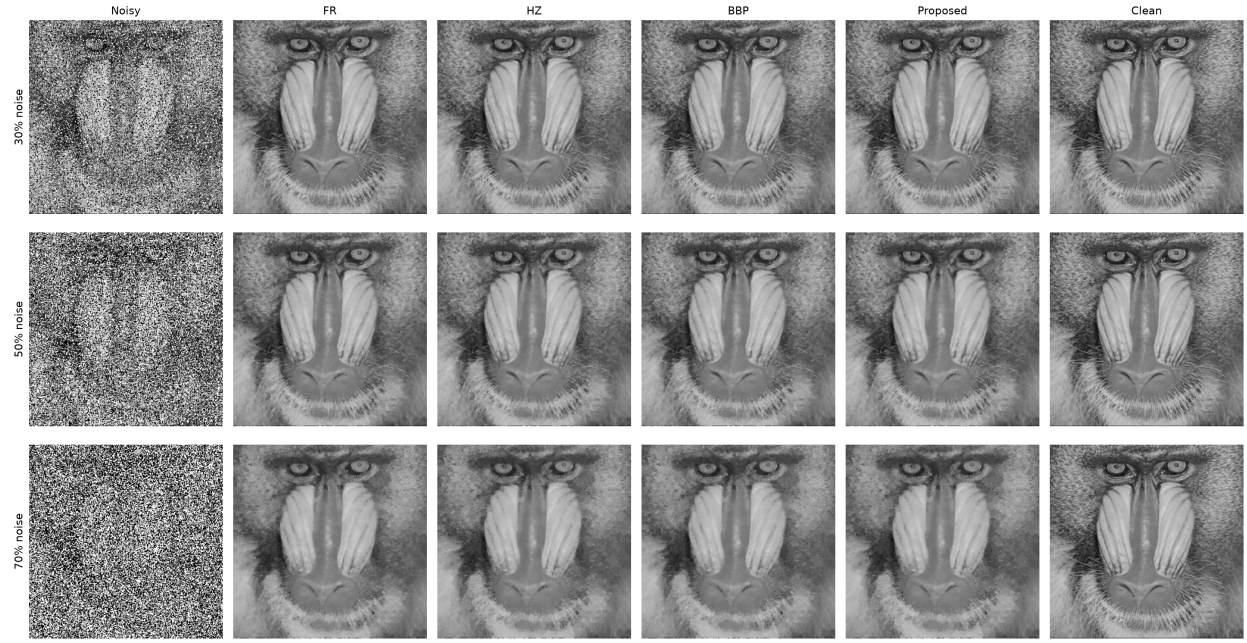


Figure 4. Visual comparison on the Mandrill image at 30%, 50% and 70% salt-and-pepper noise.



Figure 5. Visual comparison on the Man image at 30%, 50% and 70% salt-and-pepper noise.

Table 2. Performance of FR, HZ, BBP, fixed BBP–HZ hybrid, adaptive BBP–HZ hybrid (no safeguard) and the adaptive BBP–HZ hybrid with descent safeguard, on salt-and-pepper noise removal.

Image	Noise (%)	FR				HZ				BBP				Fixed BBP–HZ				Adaptive BBP–HZ				Adaptive + Safeguard			
		NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM	NI	NF	PSNR	SSIM
Milkdrop	10	400	2136	46.00	0.9953	81	194	45.80	0.9952	309	1631	46.17	0.9953	89	212	45.86	0.9952	153	533	45.99	0.9953	153	533	45.99	0.9953
	20	400	2606	42.31	0.9903	120	273	42.00	0.9902	139	460	42.12	0.9902	141	321	42.17	0.9902	130	289	42.07	0.9902	130	289	42.07	0.9902
	30	400	2681	38.96	0.9843	120	284	38.74	0.9842	168	365	38.73	0.9842	156	373	38.84	0.9843	139	351	38.75	0.9842	139	351	38.75	0.9842
	50	400	2863	35.47	0.9645	247	543	35.52	0.9666	201	444	35.52	0.9666	195	427	35.53	0.9666	195	587	35.53	0.9666	195	587	35.53	0.9666
	70	400	2686	29.71	0.8775	310	672	29.95	0.9288	282	627	30.01	0.9290	315	703	29.98	0.9289	304	1022	30.01	0.9290	303	1009	30.01	0.9290
	90	400	2615	23.71	0.6886	400	880	23.75	0.7992	400	1104	23.74	0.7989	400	872	23.75	0.7993	400	968	23.74	0.7990	390	968	23.74	0.7990
Mandrill	10	195	1000	34.41	0.9740	60	131	34.41	0.9740	105	583	34.40	0.9740	60	133	34.41	0.9740	88	228	34.41	0.9741	88	228	34.41	0.9741
	20	400	2201	31.36	0.9474	79	168	31.35	0.9473	73	260	31.34	0.9472	70	156	31.35	0.9473	64	284	31.33	0.9471	64	284	31.33	0.9471
	30	400	2224	29.43	0.9138	105	221	29.42	0.9137	98	360	29.42	0.9136	106	237	29.43	0.9137	81	193	29.42	0.9136	81	193	29.42	0.9136
	50	400	2628	26.58	0.8246	112	239	26.59	0.8250	114	251	26.58	0.8249	110	230	26.59	0.8249	95	213	26.58	0.8249	95	213	26.58	0.8249
	70	400	2738	24.48	0.6918	155	330	24.46	0.6904	165	361	24.47	0.6905	149	358	24.46	0.6904	152	354	24.46	0.6905	152	354	24.46	0.6905
	90	400	2660	22.00	0.4474	279	618	21.94	0.4456	294	735	21.94	0.4456	239	514	21.94	0.4455	319	1129	21.94	0.4456	319	1129	21.94	0.4456
Man	10	400	1879	37.79	0.9877	135	307	37.74	0.9877	103	244	37.63	0.9876	135	305	37.74	0.9877	113	268	37.67	0.9876	113	268	37.67	0.9876
	20	400	2240	34.65	0.9748	157	361	34.64	0.9748	191	619	34.69	0.9749	156	361	34.67	0.9749	119	286	34.60	0.9747	119	286	34.60	0.9747
	30	400	2470	32.38	0.9572	172	388	32.38	0.9572	145	335	32.34	0.9570	144	350	32.36	0.9571	134	314	32.32	0.9569	134	314	32.32	0.9569
	50	400	2625	28.96	0.9063	147	340	28.88	0.9068	135	303	28.88	0.9068	140	313	28.90	0.9069	146	417	28.89	0.9069	146	417	28.89	0.9069
	70	400	2641	25.33	0.7958	281	619	25.30	0.8027	233	843	25.32	0.8031	260	590	25.31	0.8028	310	1175	25.33	0.8032	310	1175	25.33	0.8032
	90	400	2561	20.78	0.5282	400	881	20.48	0.5425	400	1161	20.49	0.5428	400	902	20.48	0.5425	400	1362	20.50	0.5429	400	1362	20.50	0.5429

4.3. Discussion

As in [10], plain FR needs substantially more iterations and function evaluations than any BBP-based configuration to hit the same stopping tolerance, and at the higher noise ratios it often burns through the entire 400-iteration budget without converging at all (Table 1) – a well-documented weakness of FR on large, ill-conditioned problems. Across our 18 (image, noise) cells FR fails to meet the tolerance in 17 of them (Table 3), whereas HZ, BBP and the hybrids fail in only 2 (both at 90% noise). Both HZ and BBP run markedly faster while landing on PSNR and SSIM values that we cannot distinguish at single-realization resolution (differences well under 0.1 dB and 10^{-3} SSIM), which is consistent with BBP being a genuinely competitive alternative to a well-established guaranteed-descent method, not merely a theoretical curiosity. We state this comparison carefully: with a single noise realization per cell and no significance test, we can claim comparable quality, not a tested equality.

On relative cost we are equally careful. The main efficiency fact is not that the hybrid beats HZ, but that it matches HZ while costing far less than BBP: most of the practical benefit over BBP traces back to hybridizing BBP and HZ with a *fixed* $\lambda_k \equiv 0.5$: Table 3 shows this fixed hybrid reaching the lowest average NF of any configuration (408.7 – a 31% cut from BBP alone's 593.7, and essentially tied with HZ's 413.8) at essentially unchanged PSNR/SSIM. Relative to HZ specifically, the fixed hybrid's NF advantage is marginal (408.7 vs. 413.8, about 1%); the honest reading is that hybridization recovers HZ-level efficiency from the more expensive BBP, rather than improving on HZ. The *adaptive* weight (12) does not improve on this, though: it burns more function evaluations than the fixed hybrid on average (553.3 vs. 408.7, a 35% jump, Table 3), and considerably more on several individual (image, noise) cells – Mandrill at 90% noise, for instance, takes 1129 evaluations against 514, more than double, for essentially the same PSNR (Table 2) – all without ever beating the fixed hybrid's PSNR or SSIM by a meaningful margin. So on this benchmark, *hybridization* is what actually helps; the parameter-light *adaptivity* rule (12) hypothesized in Section 2 does not justify its added complexity. A direct diagnostic explains why. We logged the realized weight λ_k at every iteration and averaged it per cell. The mean λ_k is high and remarkably stable across noise levels – $0.884, 0.874, 0.853, 0.883, 0.875, 0.907$ at $10/20/30/50/70/90\%$ noise, i.e. about 0.88 throughout. In words, the curvature indicator ρ_k almost always dominates the descent indicator η_k, so the rule (12) puts roughly 88% of its weight on β_k^{BBP}. The adaptive hybrid therefore behaves much like BBP with a small HZ correction, and its cost drifts back toward BBP's (553.3 NF, between the fixed hybrid's 408.7 and BBP's 593.7) exactly as this would predict. The rule does not spend its freedom where it would help – it rarely moves toward HZ even when HZ is the cheaper component – which is the concrete mechanism behind the negative result. This is exactly the outcome the research caution in Section 2 anticipated: ρ_k and η_k form a plausible, but unproven, basis for λ_k, and our experiments show that, as defined, they do not outperform a fixed weight on this energy. Rather than treating this as grounds to abandon hybridization, we take it as a genuine negative result worth reporting, and return to it in Section 5.

On this smooth, well-conditioned energy the descent safeguard rarely triggers – not once for the fixed hybrid across our experiments (0.00 average restarts), and only occasionally for the adaptive hybrid (0.17 average restarts) out of several hundred iterations (Table 3) – so NI/NF/PSNR barely move, differing from the unsafeguarded

variants by no more than rounding noise in Table 2. The scale clause (15) was inactive at the loose threshold $M = 10^3$ we used; all restarts counted in Table 3 come from the descent test (14). This is expected on a convex, well-conditioned energy, where the hybrid directions stay well scaled relative to the gradient. Here its practical role is a correctness guarantee (Theorem 2) that comes essentially free on a well-behaved problem, rather than a source of measurable gains on this particular benchmark; we would expect it to matter more under noisier gradients or less regular objectives, a direction we leave for future work (Section 5). We therefore describe the safeguard, honestly, as a theoretical safety net on this benchmark, not as a source of numerical improvement. One more observation: at 90% noise, the Man image fails to meet our convergence tolerance within 400 iterations for *any* method (Table 1) , and Milkdrop and Mandrill also fail for most methods at that level (Table 3 records 2 failing cells each for HZ, BBP and the hybrids, all at 90%) – with barely 10% of pixels left as clean anchors at that noise level, this reflects the sheer difficulty of the problem rather than any particular method's shortcoming, and PSNR still improves substantially over the noisy input regardless.

4.4. Component Analysis

Table 3 teases apart the contribution of each of the three ingredients studied – hybridization, the adaptivity of λ_k, and the descent safeguard – by averaging every metric over all three test images and all six noise levels, across the six configurations of the component analysis design from Section 1 , with FR added as a reference row.

Table 3. Component Analysis: contribution of hybridization, adaptivity and the descent–scale safeguard, averaged over all three test images and six noise levels. Runtime is wall-clock seconds (indicative; see text). "Fail" counts, out of the 18 (image, noise) cells, those not meeting the tolerance within 400 iterations. FR is a reference row.

Method	Adaptive weight	Descent safeguard	Avg. NI	Avg. NF	Avg. time (s)	Avg. PSNR (dB)	Avg. SSIM	Avg. restarts	Fail /18
FR (reference)	–	–	388.6	2414.1	22.34	31.35	0.8583	0.00	17
BBP	No	Baseline	197.5	593.7	6.10	31.32	0.8685	0.00	2
HZ	No	Baseline	186.7	413.8	4.28	31.30	0.8684	0.00	2
Fixed BBP–HZ	No	No	181.4	408.7	4.90	31.32	0.8685	0.00	2
Fixed BBP–HZ + Safeguard	No	Yes	181.4	408.7	5.70	31.32	0.8685	0.00	2
Adaptive BBP–HZ	Yes	No	185.7	554.1	6.73	31.31	0.8685	0.00	2
Adaptive BBP–HZ + Safeguard	Yes	Yes	185.1	553.3	5.67	31.31	0.8685	0.17	1

Going from BBP/HZ to the fixed hybrid isolates hybridization's effect on its own; going from the fixed to the adaptive hybrid isolates adaptivity on its own; and switching the safeguard on for each hybrid isolates its effect independently of both. This decomposition is what lets us pin down exactly which piece of Algorithm 1 drives which part of the observed behaviour, rather than settling for the claim that the fully assembled method performs well – and the verdict it delivers here is mixed but informative. Hybridization (BBP/HZ $\rightarrow$ Fixed BBP–HZ) recovers HZ-level efficiency, cutting average NF by roughly 31% relative to BBP alone (matching, not beating, HZ) at matched PSNR/SSIM. Adaptivity (Fixed $\rightarrow$ Adaptive BBP–HZ) does *not* help on this benchmark, costing roughly 35% more average NF with no measurable quality payoff, for the reason diagnosed above: the weight sits near $\lambda_k \approx 0.88$ and so re-selects the costlier BBP most of the time. And the safeguard (rows "Fixed" vs. "Fixed + Safeguard" and "Adaptive" vs. "Adaptive + Safeguard" of Table 3) is essentially free – NI, NF, PSNR and SSIM all shift by under 1% while it fires on at most 0.17 iterations per run on average, and it guarantees Theorem 2 unconditionally (given a bounded weight). We would rather report this honestly than present only the fully assembled configuration: the evidence favours *hybridizing* BBP and HZ, and favours keeping the *safeguard* for its convergence guarantee at negligible cost, but does not favour this particular *adaptive* weighting rule over a fixed $\lambda_k = 0.5$. We note one caveat on the fixed weight itself: we fixed $\lambda_k \equiv 0.5$ a priori and did not sweep it, so the comparison establishes that *this* adaptive rule loses to a simple symmetric fixed weight, not that 0.5 is optimal; a systematic weight sweep is listed as future work.

5. Conclusion

This paper set out to test whether a parameter-light adaptive weighting rule improves a BBP–HZ hybrid nonlinear conjugate-gradient method for impulse-noise image restoration, and reports – through a controlled component analysis – that on this benchmark it does not. The mixing weight combines the BBP parameter of [10] with the Hager–Zhang parameter [16] through a data-dependent weight $\lambda_k \in [0, 1]$ built from local curvature and descent indicators, together with a sufficient-descent and bounded-scale restart safeguard. We proved sufficient descent and a Zoutendijk-type global convergence result holding for *any* bounded, measurable choice of λ_k once the descent–scale safeguard is in force – demonstrating that the safeguard decouples the adaptivity heuristic from the convergence guarantee, a property that holds no matter how well or poorly a given λ_k rule performs in practice. We also corrected a gap in the naive convergence argument: sufficient descent alone does not bound the search-direction norm, so the global step additionally needs the scale bound (15); and we stressed that the guarantee is asymptotic ($\liminf_k \|g_k\| = 0$), with no finite-time rate claimed.

Across salt-and-pepper denoising experiments spanning three images and six noise levels, the component analysis renders a clear verdict on the three ingredients – though not the one an "adaptive algorithm" framing would predict. *Hybridizing* BBP and HZ is worth doing for efficiency: even the naive fixed $\lambda_k \equiv 0.5$ hybrid cuts average function evaluations by roughly 31% relative to BBP alone (matching HZ) at matched PSNR/SSIM. The *safeguard* costs essentially nothing and is worth keeping purely for its (conditional) convergence guarantee (Theorem 2), altering NI/NF/PSNR/SSIM by under 1% on this benchmark while firing on at most 0.17 iterations per run on average. The *adaptive* weighting rule (12) that originally motivated this work, however, did *not* improve on the fixed hybrid empirically: it consumed substantially more function evaluations on average with no measurable gain in reconstruction quality, because its weight sits near $\lambda_k \approx 0.88$ and thus re-selects the costlier BBP most of the time. We take this to be a genuine, and useful, negative result: ρ_k and η_k offer a theoretically well-motivated but empirically *unproven* basis for λ_k on this energy – precisely the caveat flagged when the rule was first introduced (Section 2). The practical takeaway, stated plainly: on this benchmark, hybridize BBP and HZ, keep the safeguard for its convergence guarantee, and favour a fixed $\lambda_k \equiv 0.5$ over the instantaneous adaptive rule (12) until an adaptive rule is shown to beat it.

Several directions remain open for future work, and they map directly onto the limitations of this study: (i) better-performing adaptive rules for λ_k – for instance, ones built on a learned or history-averaged or exponentially smoothed curvature/descent signal rather than the noisy single-step estimates ρ_k, η_k use in Section 2, and a systematic sweep of the fixed weight λ to identify the best fixed baseline; (ii) noise models beyond salt-and-pepper impulse noise, where the safeguard (and especially the scale clause) may earn its keep more than it did here; (iii) color images and larger datasets, together with repeated-seed trials reporting variances, confidence intervals and significance tests, timing-to-tolerance and failure-rate curves, and a broader baseline set (DY, PRP, and spectral CG variants, and at least one non-CG optimizer); and (iv) hybridizing BBP/HZ with learned image priors. We also leave to future work a quantitative convergence-rate analysis under additional structural assumptions on the restoration energy, and an evaluation of how detector false positives/negatives propagate into the optimization stage, which we treated here as a fixed input.

REFERENCES

1. B. A. Hassan, and A. A. A. Abdullah, *Improvement of conjugate gradient methods for removing impulse noise images*, Indonesian Journal of Electrical Engineering and Computer Science, vol. 29, no. 1, pp. 245–251, 2023.
2. B. A. Hassan, and I. A. R. Moghrabi, *A modified secant equation quasi-Newton method for unconstrained optimization*, J. Appl. Math. Comput., vol. 69, no. 1, 2023.
3. B. A. Hassan, I. A. R. Moghrabi, and I. M. Sulaiman, *New conjugate gradient image processing methods*, Asian-European Journal of Mathematics, pp. 1–14, 2023.
4. B. A. Hassan, and H. M. Sadiq, *New class of conjugate gradient methods for removing impulse noise images*, Iraqi Journal of Science, pp. 5208–5218, 2023.
5. H. N. Jabbar, Y. J. Subhi, H. N. Hussein, and B. A. Hassan, *Solving single variable functions using a new secant method*, Journal of Interdisciplinary Mathematics, 28(1), pp. 245–251, 2025.
6. A. M. Jasim, Y. J. Subhi, and B. A. Hassan, *On new secant-method for minimum functions of one variable*, Journal of Interdisciplinary Mathematics, 28(1), pp. 291–296, 2025.

7. Y. A. Mohammed, and B. A. Hassan, *Images restoration based on a new optimal parameter to conjugate gradient method*, Statistics, Optimization & Information Computing, vol. 14, no. 6, pp. 3235–3243, 2025. DOI: 10.19139/soic-2310-5070-2799.
8. M. A. Elhamid, I. A. R. Moghrabi, B. A. Hassan, and N. Guler, *An improved hybrid nonlinear conjugate gradient method and application to image restoration problems*, Statistics, Optimization & Information Computing, 16(1), pp. 1–14, 2026. DOI: 10.19139/soic-2310-5070-3366.
9. R. Mellal, and N. Sellami, *A Novel Hybrid Conjugate Gradient Algorithm for Solving Unconstrained Optimization Problems*, Statistics, Optimization & Information Computing, 15(1), pp. 380–395, 2025. DOI: 10.19139/soic-2310-5070-2807.
10. F. Falah Aiwa, I. A. R. Moghrabi, and B. A. Hassan, *A Refined Conjugate Gradient Algorithm for Image Denoising*, Statistics, Optimization & Information Computing, DOI: 10.19139/soic-2310-5070-3605.
11. J. F. Cai, R. H. Chan, and B. Morini, *Minimization of an edge-preserving regularization functional by conjugate gradient type methods*, in Image Processing Based on Partial Differential Equations, Mathematics and Visualization, Springer, Berlin, pp. 1–7, 2007.
12. Y. H. Dai, J. Han, G. Liu, D. Sun, H. Yin, and Y. Yuan, *Convergence properties of nonlinear conjugate gradient methods*, SIAM Journal on Optimization, vol. 10, no. 2, pp. 345–358, 1999.
13. Y. H. Dai, and Y. Yuan, *A nonlinear conjugate gradient method with a strong global convergence property*, SIAM Journal on Optimization, pp. 177–182, 1999.
14. E. D. Dolan, and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., Ser. A, 91, pp. 201–213, 2002.
15. R. Fletcher, and C. M. Reeves, *Function minimization by conjugate gradients*, The Computer Journal, vol. 7, pp. 149–154, 1964.
16. W. W. Hager, and H. Zhang, *A new conjugate gradient method with guaranteed descent and an efficient line search*, SIAM Journal on Optimization, vol. 16, pp. 170–192, 2005.
17. X.-Z. Jiang, and J.-B. Jian, *A sufficient descent Dai–Yuan type nonlinear conjugate gradient method for unconstrained optimization problems*, Nonlinear Dynamics, vol. 72, pp. 101–112, 2013.
18. Y. Nakama, and H. Ito, *Conjugate gradient methods using value of objective function for unconstrained optimization*, Optimization Letters, vol. 6, no. 5, pp. 941–955, 2011.
19. J. Nocedal, and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research, 2nd ed., Springer-Verlag, New York, USA, 2006.
20. E. Polak, and G. Ribière, *Note sur la convergence de directions conjuguées*, Revue Française d'Informatique et de Recherche Opérationnelle, vol. 16, pp. 35–43, 1969.
21. X. Wei, J. Ren, X. Zhao, Z. Li, and Y. Li, *A new DY conjugate gradient method and applications to image denoising*, IEICE Transactions on Information and Systems, no. 12, pp. 2984–2990, 2018.
22. G. Yu, J. Huang, and Y. Zhou, *A descent spectral conjugate gradient method for impulse noise removal*, Applied Mathematics Letters, vol. 23, pp. 555–560, 2010.
23. G. Zoutendijk, *Nonlinear programming, computational methods*, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, pp. 37–86, 1970.
24. S. Amine, F. Koulouh, M. ES-SABRY, and N. el Akkad, *Fast and secure color image cryptosystem based on 2D Henon chaotic map and Josephus-zigzag permutation*, Statistics, Optimization & Information Computing, 15(6), pp. 5284–5312, 2026.
25. L. Fan, F. Zhang, H. Fan, and C. Zhang, *Brief review of image denoising techniques*, Visual Computing for Industry, Biomedicine and Art, vol. 2, no. 1, article 7, 2019. DOI: 10.1186/s42492-019-0016-7.
26. A. Buades, B. Coll, and J. Morel, *A review of image denoising algorithms, with a new one*, Multiscale Modeling & Simulation, vol. 4, no. 2, pp. 490–530, 2005. DOI: 10.1137/040616024.
27. A. Buades, B. Coll, and J. Morel, *A non-local algorithm for image denoising*, in Proc. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005. DOI: 10.1109/cvpr.2005.38.
28. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, *Image denoising by sparse 3-D transform-domain collaborative filtering*, IEEE Transactions on Image Processing, 2007. DOI: 10.1109/tip.2007.901238.
29. M. Elad, and M. Aharon, *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Transactions on Image Processing, 2006. DOI: 10.1109/tip.2006.881969.
30. M. Elad, B. Kawar, and G. Vaksman, *Image Denoising: The Deep Learning Revolution and Beyond – A Survey Paper*, SIAM J. Imaging Sci., 2023. DOI: 10.48550/arxiv.2301.03362.
31. J. Mao, L. Sun, J. Chen, and S. Yu, *Overview of Research on Digital Image Denoising Methods*, Sensors (Basel, Switzerland), 2025. DOI: 10.3390/s25082615.
32. USC Signal and Image Processing Institute, *The USC-SIPI Image Database*, University of Southern California. Available: https://sipi.usc.edu/database/.