Variable metric techniques for forward-backward methods in imaging

S. Bonettini[a,*], F. Porta[a], V. Ruggiero[b], L. Zanni[a]

[a] *Dipartimento di Fisica, Informatica e Matematica, Università di Modena e Reggio Emilia via Campi 213/b, 41125 Modena, Italy*
[b] *Dipartimento di Matematica e Informatica, Università di Ferrara via Machiavelli 30, 44121 Ferrara, Italy*

Abstract

Variable metric techniques are a crucial ingredient in many first order optimization algorithms. In practice, they consist in a rule for computing, at each iteration, a suitable symmetric, positive definite scaling matrix to be multiplied to the gradient vector. Besides quasi-Newton BFGS techniques, which represented the state-of-the-art since the 70's, new approaches have been proposed in the last decade in the framework of imaging problems expressed in variational form. Such recent approaches are appealing since they can be applied to large scale problems without adding significant computational costs and they produce an impressive improvement in the practical performances of first order methods. These scaling strategies are strictly connected to the shape of the specific objective function and constraints of the optimization problem they are applied to; therefore, they are able to effectively capture the problem features. On the other side, this strict problem dependence makes difficult extending the existing techniques to more general problems. Moreover, in spite of the experimental evidence of their practical effectiveness, their theoretical properties are not well understood. The aim of this paper is to investigate these issues; in particular, we develop a unified framework for scaling techniques, multiplicative algorithms and the Majorization-Minimization approach. With this inspiration, we propose a scaling matrix rule for variable metric first order methods applied to nonnegatively constrained problems exploiting a suitable structure of the objective function. Finally, we evaluate the effectiveness of the proposed approach on some image restoration problems.

*Corresponding author
Email addresses: silvia.bonettini@unimore.it (S. Bonettini), federica.porta@unimore.it (F. Porta), valeria.ruggiero@unife.it (V. Ruggiero), luca.zanni@unimore.it (L. Zanni)

Keywords: numerical optimization, forward–backward methods, variable metric tecniques, image restoration
2010 MSC: 65K05, 90C06, 90C90

1. Introduction

In image restoration problems, the data are represented by a nonnegative vector $g \in \mathbb{R}^m$ corresponding to some noisy measurements of the true object, $\tilde{x} \in \mathbb{R}^n$, we would like to observe. In many applications, the measurement (or acquisition) process can be modeled by a linear operator $H \in \mathbb{R}^{m\times n}$ and the image restoration problem consists in computing an approximation of $\tilde{x}$, knowing g and H, possibly taking into account also a nonnegative background constant $b_g \in \mathbb{R}^m$. Due to the ill-posedness of the problem and to the presence of noise, a direct solution of the linear system $Hx + b_g = g$ does not produce a meaningful solution. As a valid alternative, the Bayesian paradigm [1, 2] leads to a variational reformulation of the inverse problem as

$$\min_{x\in\mathbb{R}^n} \mathcal{D}(Hx + b_g; g) + \mathcal{R}(x) \tag{1}$$

where $\mathcal{D}(Hx + b_g; g)$ expresses the data discrepancy, while $\mathcal{R}(x)$ represents a regularization term, introducing some *a priori* information in the model and forcing desired properties on the solution.

The fit-to-data term is usually defined through the Maximum Likelihood principle according to the noise statistics. For example, the data discrepancy corresponding to Poisson noise is the Kullback-Leibler divergence

$$\mathcal{D}(z; g) = \sum_{i=1}^{m} g_i \log \frac{g_i}{z_i} + z_i - g_i, \tag{2}$$

while the least squares functional

$$\mathcal{D}(z; g) = \frac{1}{2}\|z - g\|^2 \tag{3}$$

is related to Gaussian noise. The functions in the above examples are both convex and smooth, but other kinds of noise lead to nonconvex (e.g. Cauchy noise, signal dependent Gaussian noise) or nonsmooth (impulse noise) data discrepancy.

As regards the regularization term, it may consists in simple bounds, for example nonnegativity constraints, or in more general functions. Typical examples are: Total Variation,

Hypersurface Potential and Markov Random Fields [3, 1, 2] for edge preserving, Tikhonov for smoothness preservation, ℓ_1 for sparsity promoting, or a combination of them.
By separating smooth from nonsmooth terms, (1) can be rewritten also as

$$\min_{x\in\mathbb{R}^n} \; F(x) \equiv \Phi(x) + \Psi(x). \tag{4}$$

where $\Phi : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is possibly nonconvex, continuously differentiable on an open subset of $\mathbb{R}^n$ containing $dom(\Psi) = \{x \in \mathbb{R}^n : \Psi(x) < +\infty\}$ and $\Psi : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is convex, possibly nonsmooth. A special instance of (4) is the constrained minimization problem

$$\min_{x\in\Omega} \; \Phi(x), \tag{5}$$

which is recovered when Ψ reduces to the indicator function ι_Ω of a non-empty, closed, convex set Ω

$$\iota_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega \\ +\infty & \text{otherwise} \end{cases} .$$

In the framework of signal and image processing, one of the most popular approaches for solving (4) is the family of *forward–backward* (FB) methods [4, 5], which includes as special instance the projected gradient methods [6] for the case (5).
Such kind of methods are well suited for large scale problems, such as imaging applications, where a medium accuracy solution is usually satisfactory, since they use only first order information - the gradient of $\Phi(x)$ - and their implementation does not require a large amount of memory. On the other side, they often exhibit a slow convergence behaviour, resulting in a large number of iterations to obtain an acceptable approximation of the solution.
Variable metric techniques have been proposed in the recent literature as a tool to be included in FB methods, especially with the aim of accelerating the progress towards a solution [7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]. More precisely, the variable metric approach is based on the following definition of the proximity operator of a convex function Ψ at x with respect to the metric induced by a given symmetric, positive definite matrix D:

$$\text{prox}_{\Psi,D}(x) := \underset{y\in\mathbb{R}^n}{\text{argmin}} \; \Psi(y) + \frac{1}{2}\|y - x\|_D^2, \tag{6}$$

where the D-metric of a vector y is defined as $\|y\|_D = \sqrt{y^T D y}$. Then, the general iteration of a variable metric FB method can be stated as follows [19, 20, 12]:

$$\begin{aligned} d^{(k)} &= \text{prox}_{\alpha_k \Psi, D^{(k)}}(x^{(k)} - \alpha_k {D^{(k)}}^{-1} \nabla \Phi(x^{(k)})) - x^{(k)} \\ x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)}, \end{aligned} \tag{7}$$

where α_k and λ_k are positive parameters controlling the steplength and $D^{(k)}$ is a symmetric, positive definite matrix, to be chosen at each iteration k.

It is worth stressing that $D^{(k)}$ not only affects the metric involved in the proximal point computation, but its inverse also multiplies the gradient direction: for this reason, variable metric strategies are also referred in the literature as *scaling techniques*.

When $\Psi = \iota_\Omega$, the proximity operator reduces to the projection operator with respect to the $D^{(k)}$-norm, $\text{prox}_{\alpha_k \Psi, D^{(k)}} \equiv \text{P}_{\Omega, D^{(k)}}$

$$\text{P}_{\Omega, D^{(k)}}(x) = \underset{y \in \Omega}{\text{argmin}} \ \frac{1}{2} \|y - x\|^2_{D^{(k)}}$$

and the variable metric FB scheme (7) consists in the Scaled Gradient Projection (SGP) method [10, 13].

We point out that different choices of the parameters λ_k, α_k and $D^{(k)}$ in (7) lead to different algorithms in terms of convergence properties and practical performance.

While the literature on FB methods provides several approaches on how to select the steplength parameters, also in an adaptive way, the scaling matrix selection is a challenging and less investigated problem. The latter issue is the main focus of this paper.

From a practical point of view, a good scaling matrix selection rule should consist in a formula which, given the current iterate $x^{(k)}$ and, possibly, the gradient of $\Phi(x^{(k)})$ produces a matrix $D^{(k)}$ such that: 1) the theoretical convergence of the iterates to a solution is preserved; 2) it is easy to invert; 3) it improves the effectiveness of the whole algorithm.

As far as the theoretical convergence issue, iteration (7) has been analyzed by several authors under different settings [10, 13, 15, 21, 19, 20, 22, 23]. Typically, the assumptions on the scaling matrix are not very restrictive: indeed, regardless of other properties, the convergence of the iterates to a solution can be proved when $\{D^{(k)}\}_{k \in \mathbb{N}}$ is any sequence of symmetric positive definite matrices whose eigenvalues are bounded or, in addition, also converge to a fixed value, namely:

- $\dfrac{1}{L} \leq \delta_i(D^{(k)}) \leq L, \quad i = 1, ..., n, \quad \text{for all} \ \ k \geq 0, \quad L \geq 1, \ \ \text{where} \ \delta_i(D^{(k)})$ represents

the i-th eigenvalue of $D^{(k)}$,

or, in addition,

- $D^{(k+1)} \preceq (1+\xi_k)D^{(k)}, \qquad \xi_k \geq 0, \qquad \sum_{k=0}^{\infty} \xi_k < \infty$, where, given $A, B \in \mathbb{R}^{n\times n}$ symmetric and positive definite matrices, the notation $A \preceq B$ indicates that $B - A$ is a symmetric and positive definite matrix. This last condition states that the sequence $\{D^{(k)}\}_{k\in\mathbb{N}}$ asymptotically approaches a constant matrix [19, Lemma 2.3].

The ideas developed in this work are thought for variable metric FB methods whose sequence of scaling matrices only satisfies one of the previous recalled hypotheses.

Under these assumptions on $\{D^{(k)}\}_{k\in\mathbb{N}}$, the convergence rate on the objective function values is at most $\mathcal{O}\left(\frac{1}{k}\right)$ [24, 13, 22, 23] and the same lower complexity bounds can be given independently on the choice of the parameters [16]. Nevertheless the convergence analysis does not give a clear indication on how to define $D^{(k)}$ in order to improve the performances. Actually, in [12], the authors build a sequence of scaling matrices starting from the convergence analysis but the hypotheses made on $D^{(k)}$ are different from the ones we are considering.

On the other side, it has been numerically shown [10, 13, 25] that method (7) equipped with suitable strategies for selecting $D^{(k)}$ can reach performances comparable with algorithms with known superlinear convergence rate.

In spite of the experimental evidence of the huge impact that scaling techniques may have on algorithm performance, it is still unclear how to give an explanation of these good numerical results, particularly at the early stage of the iterative process, and how to design general, adaptive and effective rules.

Our aim is to investigate these two issues, also giving an overview of existing approaches. The main contribution of the paper is a rule for computing a scaling matrix when $\Psi(x)$ in (4) includes nonnegativity constraints and $\Phi(x)$ has a suitable, quite general structure.

Our discussion moves from the Split Gradient technique [9, 26], described in Section 2, a simple strategy for computing a diagonal scaling matrix $D^{(k)}$ for nonnegatively constrained problems, which is at the basis of two very popular image deconvolution methods: the Lucy-Richardson (LR) method, known also as Expectation Maximization (EM), and the Iterative Space Reconstruction Algorithm (ISRA). Borrowing the ideas in [27], in Section 3 we show that applying the Split Gradient scaling technique corresponds to the minimization of suit-

able auxiliary (or surrogate) functions. Hence, from the properties of surrogate functions, these scaling techniques promote the objective function decrease. In this framework, a surrogate function, and the corresponding scaling matrix, can be computed for any term $\Phi(x)$ which can be written as a combination of logarithms and/or powers, as described in Section 3.1. These ideas are extended in Section 3.2, to consider a larger class of function: in particular, Tikhonov, Hypersurface potential, which is a generalization of the Total Variation functional, and Markov Random Fields can be included in our extended analysis, as well as nonnegatively constrained least squares problems where the Hessian may have negative entries. At the best of our knowledge, these contributions are new.

The scaling matrix obtained from the auxiliary function approach must be adapted to fit the convergence framework for FB methods, as explained in Section 3.3. Here, we also show that the same scaling techniques can be included also in the variant of FB methods based on an inertial/extrapolation step.

The numerical assessment of the proposed techniques is described in Section 4, where the impact of scaling strategies is evaluated on some image restoration applications.

Notations . Subscripts denote the entries of matrices and vectors, possibly in square brackets: for example, x_i and $[x]_i$ denote the i-th component of the vector x (or the i-th occurrence of a sequence of scalars), while A_{ij} and $[A]_{ij}$ indicate the element of A on the i-th row, j-th column. Superscripts in round brackets indicate an element of a finite or infinite sequence of vector or matrices (for example $x^{(k)}$, $D^{(k)}$). We denote by $\mathrm{sign}(\cdot)$ the sign function, i.e. $\mathrm{sign}(x) = 1$ if $x \geq 0$, $\mathrm{sign}(x) = -1$ if $x < 0$. Any function from $\mathbb{R}$ to $\mathbb{R}$ applied to a vector, as well as equalities and inequalities between vectors, are to be intended component-wise.

2. Scaling matrix selection strategy: state of the art

In this section we describe some popular methods based on scaling/variable metric strategies which can be cast in the form (7).

The most classical examples of scaled gradient methods are Newton, quasi-Newton and Gauss-Newton methods. However, the scaling matrix associated to these classical examples requires the computation of the Hessian $\nabla^2\Phi(x^{(k)})$ or an approximation of it, and this may lead to expensive additional computations, especially on large scale problems.

Therefore, in the following we will consider scaling strategies based only on the gradient

$\nabla\Phi(x^{(k)})$, which is already available since it is the fundamental ingredient of the FB iteration itself.

2.1. The Split Gradient strategy

This technique was introduced in the framework of image deblurring [9, 26] and nonnegative matrix factorization [28, 27], as a way to design scaled gradient methods.
The Split Gradient strategy was initially proposed for solving problem (4) when $\Psi(x)$ reduces to nonnegativity constraints. For sake of simplicity, here and in the following we restrict our attention to the case

$$\min_{x\geq 0} \Phi(x), \tag{8}$$

even if the core of our discussion can be extended to lower bounds constraints and also when $\Psi(x)$ is the sum of the indicator function of the nonnegative orthant plus other convex, nonsmooth terms.
The first order optimality conditions of (8) write as

$$x\cdot\nabla\Phi(x) = 0, \quad x \geq 0, \quad \nabla\Phi(x) \geq 0, \tag{9}$$

where $\cdot$ denotes the component-wise product. The basic idea is to devise two functions $V, U : \mathbb{R}^n \to \mathbb{R}^n$ such that the gradient of Φ can be decomposed as

$$\nabla\Phi(x) = V(x) - U(x), \text{ with } V(x) > 0 \ \ \forall x > 0, \ \text{ and } \ U(x) \geq 0 \ \ \forall x > 0. \tag{10}$$

Thus, the equalities in (9) can be rewritten as a fixed point equation

$$x = x.\frac{U(x)}{V(x)}, \tag{11}$$

where the fraction symbol indicates component-wise division. Given a strictly positive starting vector $x^{(0)}$, the above equation can be solved by the fixed point method

$$x^{(k+1)} = x^{(k)}.\frac{U(x^{(k)})}{V(x^{(k)})}. \tag{12}$$

Since the starting vector $x^{(0)}$ is positive, all the subsequent iterates remain strictly positive.

Examples. In the field of the variational approach to image restoration, two very popular methods are the Iterative Space Reconstruction Algorithm (ISRA) [29] and the Expectation Maximization or Richardson Lucy (EM-RL) method [7, 30]. Both methods apply to the

nonnegatively constrained problem (8) with $\Phi(x) = \mathcal{D}(Hx; g)$, where $\mathcal{D}(\cdot; g)$ is defined as in (3) in the first case, while the latter is for the minimization of the Kullback-Leibler divergence (2). In many applications, the linear model satisfies the following assumptions:

$$H_{ij} \geq 0, \quad \sum_{i=1}^{m} H_{ij} > 0, \quad \sum_{j=1}^{n} H_{ij} > 0 \quad \forall i = 1, ..., m, \quad j = 1, ..., n, \tag{13}$$

therefore, when g is a positive vector, the following decompositions and corresponding multiplicative iterations perfectly fit in the framework of (10) and (12):

$$\nabla\Phi(x) = \underbrace{H^T g}_{U(x)} - \underbrace{H^T H x}_{V(x)} \;\Rightarrow\; x^{(k+1)} = x^{(k)} \cdot \frac{H^T g}{H^T H x^{(k)}} \qquad \text{(ISRA)} \tag{14}$$

$$\nabla\Phi(x) = \underbrace{H^T \frac{g}{Hx}}_{U(x)} - \underbrace{H^T \mathbb{1}}_{V(x)} \;\Rightarrow\; x^{(k+1)} = \frac{x^{(k)}}{H^T \mathbb{1}} \cdot H^T \frac{g}{H x^{(k)}} \qquad \text{(EM-RL)}, \tag{15}$$

(here $\mathbb{1}$ is the vector with all entries equal to one). □

Iteration (12) is expressed in a *multiplicative* form, but it can be written also as a scaled gradient method

$$x^{(k+1)} = x^{(k)} - \frac{x^{(k)}}{V(x^{(k)})}(V(x^{(k)}) - U(x^{(k)})) = x^{(k)} - \frac{x^{(k)}}{V(x^{(k)})} \cdot \nabla\Phi(x^{(k)}) \tag{16}$$

which fits in (7) with $\alpha_k = \lambda_k = 1$ and

$$D^{(k)} = \operatorname{diag}\left(\frac{V(x^{(k)})}{x^{(k)}}\right). \tag{17}$$

Therefore, the gradient splitting (10) leads to a multiplicative algorithm (12) which, in turn, can be interpreted as a scaled gradient method (16). The convergence of the iteration (16) is not assured in general (see [9, 26]). Nevertheless, this approach suggests a strategy to devise a scaling matrix in the framework of variable metric FB methods. In other words, the Split Gradient strategy for scaling matrix selection consists in: 1) finding a gradient decomposition as in (10); 2) computing the matrix $D^{(k)}$ in (17) (or an adaptation of it) to be employed in (7).

Such strategy has been successfully applied in a quite large class of problems, resulting in a significant improvement of the convergence behaviour of the underlying variable metric method [10, 15, 21, 18].

It is worth stressing that, in general, the splitting (10) is not uniquely defined, but several significant problems have a kind of natural decomposition, corresponding to well behaving algorithms. However, a clear explanation of these good numerical results as well as a general rule to devise a gradient splitting leading to an effective scaling matrix is still missing.

2.2. The Majorization Minimization approach

In this section we describe a different approach to scaling matrix selection, which is related to the framework of Majorization-Minimization (MM) methods. This class of algorithms is based on the following definition.

Definition 2.1. *Let $\Phi : \mathbb{R}^n \to \mathbb{R}$ be a real valued function and let $\bar{x}$ be a point belonging to its domain. Then, an auxiliary or surrogate function of Φ at $\bar{x}$ is any function $G(\cdot, \bar{x})$ satisfying the following majorization conditions:*

(i) $G(\bar{x}, \bar{x}) = \Phi(\bar{x})$;

(ii) $G(x, \bar{x}) \geq \Phi(x)$ for any x in dom(Φ).

If Φ and $G(\cdot, \bar{x})$ are differentiable functions, then we have [31, Prop. 1] $\nabla G(\bar{x}, \bar{x}) = \nabla \Phi(\bar{x})$, where $\nabla G(\bar{x}, \bar{x})$ denotes the gradient of the function $G(\cdot, \bar{x})$ evaluated at $\bar{x}$.
A MM scheme for solving the problem (8) is given by the following basic iteration

$$x^{(k+1)} = \underset{x \geq 0}{\operatorname{argmin}} \; G(x, x^{(k)}); \tag{18}$$

when $G(\cdot, \bar{x})$ is chosen as a strictly convex function with respect to the first argument, the iteration (18) is well-definite.
A direct consequence of the properties of surrogate functions is that (18) is a descent method, i.e. $\Phi(x^{(k+1)}) \leq \Phi(x^{(k)})$:

$$\Phi(x^{(k+1)}) \leq G(x^{(k+1)}, x^{(k)}) \leq G(x^{(k)}, x^{(k)}) = \Phi(x^{(k)})$$

In the literature on MM methods, a relevant issue is the construction of a surrogate in specific cases, often exploiting convexity or concavity properties of the objective function (see [32, 12, 33, 27, 34] and references therein). Indeed, an effective implementation of the method (18) requires that the minimizer of the surrogate is easily computed with an explicit formula. Two interesting examples where the surrogate is separable are described below.

Examples. In [34, 28], given a strictly positive vector $\bar{x}$, the following surrogates are computed for the least squares functional and for the KL divergence ($b_g = 0$):

$$\begin{aligned} G^{LS}(x,\bar{x}) &= (H\bar{x}-g)^T(H\bar{x}-g) + (H\bar{x}-g)^T H(x-\bar{x}) + \frac{1}{2}(x-\bar{x})\mathrm{diag}\left(\frac{H^T H\bar{x}}{\bar{x}}\right)(x-\bar{x}) \\ G^{KL}(x,\bar{x}) &= \sum_{i=1}^{m}\left\{ g_i \log g_i - g_i \sum_{j=1}^{n} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i} \log\left(\frac{x_j}{\bar{x}_j}[H\bar{x}]_i\right) + [Hx]_i - g_i \right\}. \end{aligned}$$

An easy calculation shows that

$$x^{(k+1)} = x^{(k)} \cdot \frac{H^T g}{H^T H x^{(k)}} = x^{(k)} - \frac{x^{(k)}}{H^T H x^{(k)}} \nabla\Phi(x^{(k)}) = \arg\min G^{LS}(x, x^{(k)}) \qquad \text{(ISRA)}$$

$$x^{(k+1)} = \frac{x^{(k)}}{H^T \mathbb{1}} \cdot H^T \frac{g}{Hx^{(k)}} = x^{(k)} - \frac{x^{(k)}}{H^T \mathbb{1}} \nabla\Phi(x^{(k)}) = \arg\min G^{KL}(x, x^{(k)}) \qquad \text{(EM-RL)}.$$

Therefore, we can regard ISRA and EM-RL from three different points of view: either as multiplicative algorithms based on gradient splitting, either as scaled gradient methods, or as MM methods. □

MM methods are directly related to scaling techniques when there exists a strictly convex quadratic surrogate function, written as

$$G(x, x^{(k)}) = \Phi(x^{(k)}) + \nabla\Phi(x^{(k)})^T(x - x^{(k)}) + \frac{1}{2}(x - x^{(k)})^T A^{(k)}(x - x^{(k)}).$$

Here $A^{(k)}$ is a symmetric positive definite matrix such that the minimum point of $G(\cdot, x^{(k)})$ belongs to the constraint set. Furthermore, from a practical point of view, $A^{(k)}$ has to be chosen as an easily solvable matrix, as, for example, a diagonal matrix. In this case, the MM method (18) writes also as

$$x^{(k+1)} = \underset{x \geq 0}{\mathrm{argmin}}\ G(x, x^{(k)}) = x^{(k)} - {A^{(k)}}^{-1} \nabla\Phi(x^{(k)}), \tag{19}$$

which clearly is a special instance of (7) with $D^{(k)} = A^{(k)}$ and $\alpha_k = \lambda_k = 1$.

Therefore, the MM strategy for scaling matrix selection consists in: 1) constructing a quadratic surrogate for the objective function $\Phi(x)$; 2) define $D^{(k)}$ as the Hessian of the quadratic surrogate. This approach has been considered for example in [12, 33]. Clearly, the implementation of the MM iteration (19) is effective when $A^{(k)}$ has a simple, for example diagonal, structure.

3. Proposed approach

In this section we present our approach, which is based on a combination of the two strategies described in the previous section. In particular, developing the ideas in [27], we explain the connections between SG and MM strategies, by showing how, in a quite general case, a surrogate $G(\cdot,\cdot)$ can be constructed upon a uniquely defined gradient splitting of the form (10). The crucial properties of the resulting surrogate, which is not necessarily quadratic, is that its domain is the strictly positive orthant and its unique minimum point (or an approximation of it) can be written as

$$x^{(k+1)} \simeq x^{(k)} - {D^{(k)}}^{-1} \nabla\Phi(x^{(k)})$$

where $D^{(k)}$ is a diagonal matrix with positive entries depending only on the components of $\nabla\Phi(x^{(k)})$.

Once a surrogate with these properties has been calculated, we propose to employ the associated scaling matrix $D^{(k)}$ with a suitable thresholding on its diagonal entries in the framework of variable metric FB methods (7). The thresholding technique enables to assure the uniformly boundedness of the eigenvalues of $D^{(k)}$, essential for the convergence results of (7). The main advantages of this strategy are: 1) it promotes both feasibility and objective function decrease; 2) it does not introduce additional expensive computations.

We point out that quadratic surrogates are also employed to define the variable metric in the FB algorithm [12], where the majorization property is crucial for proving the convergence of the iterates. Our point of view is different, since the convergence analysis of the FB methods where we want to apply our approach allows much more freedom to choose the scaling matrix. Indeed, as it will be better explained in Section 3.3, the practical implementation of our scaling strategy, taking into account of numerical and theoretical requirements, will result in an approximate majorization property.

3.1. Construction of the surrogate

In this section we revisit and extend the approach in [27] with the aim to calculate a surrogate for any function which can be written in the form

$$f_0(x) = \sum_{d=1}^{p} \sum_{i=1}^{m} \alpha_{d,i} h([Hx]_i + c_{d,i}, \zeta_d) \tag{20}$$

where $H \in \mathbb{R}^{n\times m}$ is a matrix with nonnegative entries such that $Hx > 0$ for any $x > 0$, $c_{d,i}$ are nonnegative constant parameters and

$$h(\sigma, t) = \begin{cases} \frac{\sigma^t - 1}{t} & \text{if } t \neq 0 \\ \log(\sigma) & \text{if } t = 0 \end{cases} . \tag{21}$$

The domain of $h(\cdot, t)$ is $[0, \infty)$ if $t > 0$ and $(0, \infty)$ if $t \leq 0$.

Examples: data discrepancy for Gaussian noise, Poisson noise and signal dependent Gaussian noise. The least–squares function (3) with background $b_g \geq 0$ can be written in the form (20) with the settings

$$\begin{aligned} p = 2 \quad c_{d,i} = b_g \quad \alpha_{1,i} = -g_i \quad \zeta_1 = 1, \\ \alpha_{2,i} = 1 \qquad \zeta_2 = 2, \end{aligned} \tag{22}$$

up to the additive constant $\sum_{1=1}^{m} (\frac{g_i^2}{2} + g_i) + \frac{m}{2}$.
Also the generalized Kullback–Leibler divergence (2) fits into the structure (20) up to an additive constant independent of x, by setting

$$\begin{aligned} p = 2 \quad c_{d,i} = b_g \quad \alpha_{1,i} = -g_i \quad \zeta_1 = 0, \\ \alpha_{2,i} = 1 \qquad \zeta_2 = 1. \end{aligned} \tag{23}$$

The additive constant is $m + \sum_{i=1}^{m} (g_i \log g_i - g_i)$.
A further example is the negative log-likelihood discrepancy corresponding to signal dependent Gaussian noise (see [12] end reference therein):

$$f_0(x) = \frac{1}{2} \sum_{i=1}^{m} \frac{([Hx]_i - g_i)^2}{a_i [Hx]_i + b_i} + \log(a_i [Hx]_i + b_i), \tag{24}$$

where a_i, b_i are positive parameters related to the noise model. It is easy to verify that the previous expression is equivalent to

$$f_0(x) = \frac{1}{2} \sum_{i=1}^{m} \left\{ \frac{1}{a_i} \left(([Hx]_i + \eta_i) + \frac{(g_i + \eta_i)^2}{[Hx]_i + \eta_i} - 2(g_i + \eta_i) \right) + \log([Hx]_i + \eta_i) + \log(a_i) \right\},$$

with $\eta_i = b_i / a_i$. Therefore, this nonconvex functional is a special case of (20) with

$$\begin{aligned} p = 3 \quad c_{d,i} = \eta_i \quad \alpha_{1,i} = \tfrac{1}{2a_i} \qquad \zeta_1 = 1, \\ \alpha_{2,i} = -\tfrac{(g_i + \eta_i)^2}{2a_i} \quad \zeta_2 = -1, \\ \alpha_{3,i} = \tfrac{1}{2} \qquad \zeta_3 = 0. \end{aligned} \tag{25}$$

The additive constant is $\frac{1}{2} \sum_{i=1}^{m} \log a_i + \frac{(g_i + \eta_i - 1)^2}{a_i}$ □

Useful properties of $h(\sigma, t)$ are summarized in the following lemma.

Lemma 3.1. *The function h defined in* (21) *has the following properties:*

1. *$h(\cdot, t)$ is convex with respect to the first argument if $t \geq 1$ and is concave if $t \leq 1$ (for $t = 1$ is linear, then it can be considered both convex and concave);*
2. *$h(\cdot, t)$ is continuous with respect to the first argument in its domain and monotone increasing;*
3. *$h(\sigma, \cdot)$ is continuous with respect to the second argument and monotone nondecreasing for any $\sigma > 0$.*

Proof. Claims 1, 2 are straightforward. Let us prove 3. The continuity of $h(\sigma, \cdot)$ with respect to the second argument follows from the known limit $\lim_{t\to 0^+}(\sigma^t - 1)/t = \log(\sigma)$. In order to prove the monotonicity, we write the explicit expression of the partial derivative with respect to t, for $t > 0$

$$\frac{\partial h}{\partial t}(\sigma, t) = \frac{\sigma^t(t\log(\sigma) - 1) + 1}{t^2};$$

the sign of the partial derivative is the sign of its numerator

$$\mu_\sigma(t) = \sigma^t(t\log(\sigma) - 1) + 1.$$

For any $\sigma > 0$ we have that $\mu'_\sigma(t) = \sigma^t t(\log\sigma)^2$ is negative for $t < 0$ and positive for $t > 0$. Then, the function $\mu_\sigma(t)$ has a minimum at $t = 0$. Since $\mu_\sigma(0) = 0$, we have that $\mu_\sigma(t) \geq 0$ $\forall t$. This implies that $\frac{\partial h}{\partial t}(\sigma, t) \geq 0$ for all $\sigma > 0$ and, therefore, $h(\sigma, \cdot)$ is monotone nondecreasing for any $\sigma > 0$. □

In the following, we present a general technique to define a specific surrogate for a function of the form (20). Before stating the theorem, we introduce the following notation

$$\omega_{d,i}(z) = \alpha_{d,i} h(z, \zeta_d) \tag{26}$$

so that f_0 in (20) writes also as

$$f_0(x) = \sum_{d=1}^{p}\sum_{i=1}^{m} \omega_{d,i}([Hx]_i + c_{d,i}). \tag{27}$$

We also denote by $\partial_{d,i,j}$ the partial derivative of $\omega_{d,i}([Hx]_i + c_{d,i})$ with respect to x_j at $\bar{x}$

$$\partial_{d,i,j} \equiv \left.\frac{\partial \omega_{d,i}([Hx]_i + c_{d,i})}{\partial x_j}\right|_{x=\bar{x}} = \alpha_{d,i} H_{ij} \left.\frac{\partial h(\sigma, \zeta_d)}{\partial \sigma}\right|_{\sigma=[H\bar{x}]_i + c_{d,i}} = \alpha_{d,i} H_{ij} ([H\bar{x}]_i + c_{d,i})^{\zeta_d - 1}. \tag{28}$$

Note that:

1. the last equality in (28) also holds when $\zeta_d = 0$;
2. when $H_{i,j} > 0$, in view of the monotonicity of $h(\cdot, \zeta_d)$, the sign of $\partial_{d,i,j}$ is the same of $\alpha_{d,i}$.

Based on the previous definitions we can write

$$\frac{\partial f_0(\bar{x})}{\partial x_j} = \sum_{d=1}^{p} \sum_{i=1}^{m} \partial_{d,i,j}$$

and, by grouping the positive and the negative terms, we have

$$[\nabla f_0(\bar{x})]_j = \underbrace{\sum_{(d,i)\in\mathcal{W}^+} \partial_{d,i,j}}_{\equiv [V_0(\bar{x})]_j} - \underbrace{\sum_{(d,i)\in\mathcal{W}^-} -\partial_{d,i,j}}_{\equiv [U_0(\bar{x})]_j} \tag{29}$$

where

$$\begin{aligned} \mathcal{W}^+ &= \{(d,i) \in \{1,...,p\} \times \{1,...,n\} : \alpha_{d,i} > 0\} \\ \mathcal{W}^- &= \{(d,i) \in \{1,...,p\} \times \{1,...,n\} : \alpha_{d,i} < 0\} \end{aligned}$$

Finally, we define $V_0(\bar{x}), U_0(\bar{x}) \in \mathbb{R}^n$ as the vectors whose components are $[V_0(\bar{x})]_j$ and $[U_0(\bar{x})]_j$, respectively.

We remark that with a little abuse of notation, if G fulfills Definition 2.1, we can consider as a surrogate any function of the form $G+c$, where c is any real number which does not depend on the first argument of the function G; this is because, in the framework of MM methods, we are mostly interested in the minimum points of G (see (18)) which do not change if we add a constant. For this reason, borrowing the notation from [27], we introduce the symbols $\overset{c}{=}, \overset{c}{\leq}$ to indicate that an equality (respectively inequality) holds up to an additive constant.

Theorem 3.1. *[27, Theorem 1]Let f_0 be a function with the structure* (20) *with $p \geq 2$ and let $x, \bar{x}$ be two points of its domain. Assume that $H_{ij} \geq 0$, $\bar{x}_j > 0$ for all $i, j = 1, ..., n$. Then, the function G_0 defined as*

$$G_0(x, \bar{x}) = \sum_{j=1}^{n} \bar{x}_j [V_0(\bar{x})]_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\max}\right) - \bar{x}_j [U_0(\bar{x})]_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\min}\right) \tag{30}$$

where

$$\zeta_{\max} = \max\{\zeta_1, ..., \zeta_d, 1\}, \quad \zeta_{\min} = \min\{\zeta_1, ..., \zeta_d, 1\}$$

is a convex surrogate of f_0 at $\bar{x}$ up to an additive constant.

Proof. The proof proceeds by finding a majorant for each term $\omega_{d,i}(z)$ defined in (26), considering the convex and concave cases separately.

If $\omega_{d,i}(z)$ is concave (i.e. either $\alpha_{d,i} > 0$ and $\zeta_d \leq 1$ or $\alpha_{d,i} < 0$ and $\zeta_d > 1$), for the properties of a differentiable concave function, we have

$$\omega_{d,i}(z) \leq \omega_{d,i}(\bar{z}) + \omega'_{d,i}(\bar{z})(z - \bar{z}),$$

where

$$\omega'_{d,i}(z) = \alpha_{d,i} \left. \frac{\partial h}{\partial \sigma}(\sigma, \zeta_d) \right|_{\sigma=z} = \alpha_{d,i} z^{\zeta_d - 1},$$

which, for $z = [Hx]_i + c_{d,i}$ and $\bar{z} = [H\bar{x}]_i + c_{d,i}$ gives

$$\begin{aligned}
\omega_{d,i}([Hx]_i + c_{d,i}) &\overset{c}{\leq} \omega'_{d,i}([H\bar{x}]_i + c_{d,i})([Hx]_i + c_{d,i}) \overset{c}{=} \omega'_{d,i}([H\bar{x}]_i + c_{d,i})([Hx]_i) \\
&= \sum_{j=1}^{n} \underbrace{\alpha_{d,i}([H\bar{x}]_i + c_{d,i})^{\zeta_d - 1} H_{ij}}_{\partial_{d,i,j}} x_j \\
&= \sum_{j=1}^{n} \partial_{d,i,j} x_j \overset{c}{=} \sum_{j=1}^{n} \partial_{d,i,j} \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 1\right).
\end{aligned}$$

We now consider the case when $\omega_{d,i}(z) = \alpha_{d,i} h(z, \zeta_d)$ is convex (i.e. either $\alpha_{d,i} > 0$ and $\zeta_d > 1$ or $\alpha_{d,i} < 0$ and $\zeta_d \leq 1$). A simple algebra gives

$$\begin{aligned}
\omega_{d,i}([Hx]_i + c_{d,i}) &= \alpha_{d,i} h\left(\sum_{j=1}^{n} H_{ij} x_j + c_{d,i}, \zeta_d\right) \\
&= \alpha_{d,i} h\left(\sum_{j=1}^{n} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} \frac{x_j}{\bar{x}_j}([H\bar{x}]_i + c_{d,i}) + \frac{c_{d,i}}{[H\bar{x}]_i + c_{d,i}}([H\bar{x}]_i + c_{d,i}), \zeta_d\right)
\end{aligned}$$

Since the $n+1$ coefficients $\frac{c_{d,i}}{[H\bar{x}]_i + c_{d,i}}, \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}}, j = 1, .., n$ sum to one, the Jensen's inequality yields

$$\begin{aligned}
\omega_{d,i}([Hx]_i + c_{d,i}) &\leq \sum_{j=1}^{n} \alpha_{d,i} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} h\left(\frac{x_j}{\bar{x}_j}([H\bar{x}]_i + c_{d,i}), \zeta_d\right) + \\
&\quad + \alpha_{d,i} \frac{c_{d,i}}{[H\bar{x}]_i + c_{d,i}} h\left([H\bar{x}]_i + c_{d,i}, \zeta_d\right) \\
&\overset{c}{=} \sum_{j=1}^{n} \alpha_{d,i} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} h\left(\frac{x_j}{\bar{x}_j}([H\bar{x}]_i + c_{d,i}), \zeta_d\right).
\end{aligned}$$

If $\zeta_d \neq 0$ the right-hand-side of the previous inequality can be written as

$$\begin{aligned}\sum_{j=1}^{n} \alpha_{d,i} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} h\left(\frac{x_j}{\bar{x}_j}([H\bar{x}]_i + c_{d,i}), \zeta_d\right) &= \sum_{j=1}^{n} \alpha_{d,i} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} \frac{1}{\zeta_d}\left[\left(\frac{x_j}{\bar{x}_j}([H\bar{x}]_i + c_{d,i})\right)^{\zeta_d} - 1\right] \\ &\overset{c}{=} \sum_{j=1}^{n} \alpha_{d,i} H_{ij} ([H\bar{x}]_i + c_{d,i})^{\zeta_d - 1} \frac{\bar{x}_j}{\zeta_d}\left(\frac{x_j}{\bar{x}_j}\right)^{\zeta_d} \\ &= \sum_{j=1}^{n} \partial_{d,i,j} \frac{\bar{x}_j}{\zeta_d}\left(\frac{x_j}{\bar{x}_j}\right)^{\zeta_d} \\ &\overset{c}{=} \sum_{j=1}^{n} \partial_{d,i,j} \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_d\right).\end{aligned}$$

When $\zeta_d = 0$, we have that

$$h\left(\frac{x_j}{\bar{x}_j}([H\bar{x}]_i + c_{d,i}), 0\right) = h\left(\frac{x_j}{\bar{x}_j}, 0\right) + h([H\bar{x}]_i + c_{d,i}, 0)$$

which gives

$$\begin{aligned}\sum_{j=1}^{n} \alpha_{d,i} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} h\left(\frac{x_j}{\bar{x}_j}[H\bar{x}]_i + c_{d,i}, 0\right) &\overset{c}{=} \sum_{j=1}^{n} \alpha_{d,i} \frac{H_{ij}\bar{x}_j}{[H\bar{x}]_i + c_{d,i}} h\left(\frac{x_j}{\bar{x}_j}, 0\right) \\ &= \sum_{j=1}^{n} \partial_{d,i,j} \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 0\right).\end{aligned}$$

In both cases we obtain that

$$\omega_{d,i}([Hx]_i + c_{d,i}) \overset{c}{\leq} \sum_{j=1}^{n} \partial_{d,i,j} \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_d\right). \tag{31}$$

Now, we notice that, by definition (28), since $\bar{x}_j$ and H_{ij} are nonnegative, the quantities $\partial_{d,i,j}$ have the same sign of $\alpha_{d,i}$. Then, by exploiting the monotonicity of $h(\sigma, \cdot)$ with respect to its second argument, we can further majorize (31), obtaining

$$\omega_{d,i}([Hx]_i + c_{d,i}) \overset{c}{\leq} \begin{cases} \sum_{j=1}^{n} \partial_{d,i,j} \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\max}\right) & \text{if } \alpha_{d,i} > 0, \\ \sum_{j=1}^{n} \partial_{d,i,j} \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\min}\right) & \text{if } \alpha_{d,i} < 0. \end{cases} \tag{32}$$

Finally we have

$$\begin{aligned} f_0(x) &= \sum_{d=1}^{p}\sum_{i=1}^{m}\omega_{d,i}([Hx]_i + c_{d,i}) \\ &= \sum_{(d,i)\in\mathcal{W}^+}\omega_{d,i}([Hx]_i + c_{d,i}) + \sum_{(d,i)\in\mathcal{W}^-}\omega_{d,i}([Hx]_i + c_{d,i}) \\ &\overset{c}{\leq} \sum_{(d,i)\in\mathcal{W}^+}\sum_{j=1}^{n}\partial_{d,i,j}\bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\max}\right) + \sum_{(d,i)\in\mathcal{W}^-}\sum_{j=1}^{n}\partial_{d,i,j}\bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\min}\right) \\ &= \sum_{j=1}^{n}\underbrace{\left(\sum_{(d,i)\in\mathcal{W}^+}\partial_{d,i,j}\right)}_{\equiv(V_0)_j(\bar{x})}\bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\max}\right) - \sum_{j=1}^{n}\underbrace{\left(\sum_{(d,i)\in\mathcal{W}^-}-\partial_{d,i,j}\right)}_{\equiv(U_0)_j(\bar{x})}\bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, \zeta_{\min}\right), \end{aligned}$$

which gives the result. □

We observe that the assumption $p \geq 2$ guarantees that there exist at least two values of the second argument of h different from each other; therefore it cannot happen that ζ_{max} is equal to ζ_{min}.

In order to derive the MM method (18) associated with the surrogate (30), we consider the gradient of G_0 with respect to its first argument (note that $G_0(\cdot, \bar{x})$ is separable):

$$\begin{aligned} \frac{\partial}{\partial x_j}G_0(x, \bar{x}) &= \bar{x}_j[V_0(\bar{x})]_j\frac{1}{\bar{x}_j}\left.\frac{\partial h(\sigma, \zeta_{\max})}{\partial\sigma}\right|_{\sigma=\frac{x_j}{\bar{x}_j}} - \bar{x}_j[U_0(\bar{x})]_j\frac{1}{\bar{x}_j}\left.\frac{\partial h(\sigma, \zeta_{\min})}{\partial\sigma}\right|_{\sigma=\frac{x_j}{\bar{x}_j}} \\ &= [V_0(\bar{x})]_j\left(\frac{x_j}{\bar{x}_j}\right)^{\zeta_{\max}-1} - [U_0(\bar{x})]_j\left(\frac{x_j}{\bar{x}_j}\right)^{\zeta_{\min}-1}. \end{aligned}$$

Solving the equation $\nabla_j G_0(x, \bar{x}) = 0$ with respect to the first argument gives

$$x = \bar{x}\cdot\left(\frac{U_0(\bar{x})}{V_0(\bar{x})}\right)^{\frac{1}{\zeta_{\max}-\zeta_{\min}}},$$

consisting in the unique minimum point of $G_0(\cdot, \bar{x})$, which has strictly positive entries as long as $\bar{x}$ is strictly positive.

Iterating the previous formula leads to the following MM method in multiplicative form

$$x^{(k+1)} = x^{(k)}\cdot\left(\frac{U_0(x^{(k)})}{V_0(x^{(k)})}\right)^{\zeta}, \quad \zeta = \frac{1}{\zeta_{\max}-\zeta_{\min}}. \tag{33}$$

When $\zeta = 1$, iteration (33) corresponds exactly to the SG method (16) with scaling matrix (17). This case includes the Kullback-Leibler divergence (23) and the least squares (22); in

particular, the surrogates and corresponding MM methods reduce to EM-LR and ISRA.

When $\zeta \neq 1$, as for example in the case (24)-(25), we propose the following approximation

$$\begin{aligned} x^{(k+1)} &= x^{(k)} \cdot \left(1 + \frac{-\nabla f_0(x^{(k)})}{V_0(x^{(k)})}\right)^{\zeta} \\ &\simeq x^{(k)} \cdot \left(1 + \zeta \frac{-\nabla f_0(x^{(k)})}{V_0(x^{(k)})}\right) \\ &= x^{(k)} - \zeta \frac{x^{(k)}}{V_0(x^{(k)})} \cdot \nabla f_0(x^{(k)}) \end{aligned}$$

where the first equality follows from (29) and the approximate equality is justified by the first order Taylor expansion $(1+s)^{\zeta} \simeq 1 + \zeta s$. In this case, the convergence of the method has to be carefully analyzed.

In conclusion, motivated by all the above discussion, we propose the following scaling strategy for problem (8) where the objective function has the form (20): 1) compute the decomposition (29); 2) define the scaling matrix

$$D^{(k)^{-1}} = \operatorname{diag}\left(\zeta \frac{x^{(k)}}{V_0(x^{(k)})}\right), \tag{34}$$

where a thresholding procedure has to be applied to guarantee convergence assumptions for the related FB method (7).

Remark . The same arguments of Theorem 3.1 can be adapted to handle the problem

$$\min_{x \geq x^{low}} f_0(x)$$

where $f_0(x)$ can be written in the form (20) and $x^{low} \in \mathbb{R}^n$ represents lower bounds for the variable x. Indeed, using the change of variable $y = x - x^{low}$, the above problem is equivalent to minimize $f_0(y + x^{low})$ subject to $y \geq 0$, which can be still expressed as in (20). Under the assumption that $[Hx^{low}]_i + c_{d,i} \geq 0$, $\forall i = 1, ..., m$, the proof of Theorem 3.1 is still valid and leads to the surrogate

$$G_0(x, \bar{x}) = \sum_{j=1}^{n} (\bar{x}_j - x_j^{low})[V_0(\bar{x})]_j h\left(\frac{x_j - x_j^{low}}{\bar{x}_j - x_j^{low}}, \zeta_{\max}\right) - (\bar{x}_j - x_j^{low})[U_0(\bar{x})]_j h\left(\frac{x_j - x_j^{low}}{\bar{x}_j - x_j^{low}}, \zeta_{\min}\right).$$

The corresponding variant of the scaling strategy (34) is

$$D^{(k)^{-1}} = \operatorname{diag}\left(\zeta \frac{(x^{(k)} - x^{low})}{V_0(x^{(k)})}\right).$$

□

We observe that, besides the assumption (20) on the structure of the objective function, most of the arguments leading to the scaling matrix above rely on the nonnegativity of the entries of the matrix H. In the next section, we describe a procedure to derive a surrogate also without the nonnegativity assumption on H_{ij}, which applies to functions which are not included in the formulation (20).

3.2. Generalizations

In this section we extend our approach to a more general case, with the aim to devise a technique, based on a suitable gradient splitting, to compute a surrogate for functions which can not be represented in the form (20). In particular, we focus on the following case

$$f_1(x) = \sum_{l=1}^{r} \phi_l(\|A^{(l)}x - b^{(l)}\|^2), \tag{35}$$

where $\phi_l : \mathbb{R} \to \mathbb{R}$ are differentiable, concave, monotone increasing functions and $A^{(l)}$ is an $\ell \times n$ matrix, possibly having negative entries. The motivation of this extension is that several interesting functionals can be expressed as in (35). For example, when H has negative entries, the least squares function (3) can not be included in the discussion of the previous section, but it can be represented as in (35) by setting $r = 1$, $\phi_1(s) = \frac{1}{2}s$, $b^{(1)} = g - b_g$ and $A^{(1)} = H$. Some further examples are described below.

Example: data discrepancy for Cauchy noise. The maximum likelihood criterion leads to the following data discrepancy function for data corrupted by Cauchy noise [35]:

$$\mathcal{D}(Hx; g) = \sum_{i=1}^{m} \log(\gamma^2 + ([Hx]_i - g_i)^2),$$

where γ is a given scalar parameter related to the noise distribution. This function is a special instance of (35) with the settings

$$r = m, \qquad \phi_l(s) = \log(\gamma^2 + s), \quad [A^{(l)}]_j = H_{lj},\ j = 1, ..., n, \quad b^{(l)} = g_l.$$

We observe that, when H has nonnegative entries, a surrogate for the above function can be computed also by applying the procedure proposed in [12, formula (36)]- [33, Table 1] (see also [21, Section 4.3]). However, this approach can not be extended to functions of the form (35) where $A^{(l)}$ has negative entries, as in the following examples. □

Examples: Tikhonov and edge preserving regularization. In the framework of image restoration, a family of very popular regularization functions is based on the discrete gradient operator, which associates to each pixel the vector whose components are the differences with respect to its neighbors. For example, for a 2D $N \times N$ image, the discrete gradient operator at the l-th pixel is the matrix $\nabla^{(l)} \in \mathbb{R}^{2\times n}$, where $n = N^2$, with all zero entries except $[\nabla^{(l)}]_{1,l} = [\nabla^{(l)}]_{2,l} = -1$, $[\nabla^{(l)}]_{1,l+1} = [\nabla^{(l)}]_{1,l+N} = 1$, assuming that some boundary conditions are set.

The Tikhonov regularization of order 1 is then obtained from (35) with $\phi_l(s) = \frac{1}{2}s$, $A^{(l)} = \nabla^{(l)}$ and $b^{(l)} = 0$.

When $\phi_l(\cdot) = \sqrt{\cdot + \delta^2}$, for some fixed $\delta \geq 0$, and $A^{(l)} = \nabla^{(l)}$, formula (35) corresponds to the Hyper-surface (HS) regularization functional [3], which, introducing the set of indexes $\mathcal{N}_l = \{l+1, l+N\}$, writes also as

$$f_1(x) = \sum_{l=1}^{n} \sqrt{\sum_{u\in\mathcal{N}_l} (x_u - x_l)^2 + \delta^2}. \tag{36}$$

For small values of δ, the HS function can be regarded as a differentiable approximation of the Total Variation (TV) function, which is recovered for $\delta = 0$.

The HS functional itself can be considered as a special case of a function of the form

$$f_1(x) = \sum_{l=1}^{n} \sqrt{\sum_{u\in\mathcal{N}_l} \left(\frac{(x_u - x_l)^2}{\epsilon_{u,l}}\right) + \delta^2} \tag{37}$$

where $\mathcal{N}_l$ is some index set and $\epsilon_{u,l}$ are weighting parameters. Clearly, $f_1(x)$ in (37) has the form (35), provided that the nonzero elements of $A^{(l)}$ are defined according to the indexes in $\mathcal{N}_l$.

When $\mathcal{N}_l$ contains the indexes of all the 8 nearest neighbors of the l-th pixel, a typical choice for the weights is $\epsilon_{u,l} = 1$ for vertical and horizontal, and $\epsilon_{u,l} = \sqrt{2}$ for diagonal neighbors: the function (37) corresponding to these settings is known in the literature as the Markov Random Field (MRF) regularization [1]. □

In the following, we show that the convex function

$$G_1(x, \bar{x}) = \sum_{j=1}^{n} \bar{x}_j [V_1(\bar{x})]_j h\left(\frac{x_j}{\bar{x}_j}, 2\right) - \bar{x}_j [U_1(\bar{x})]_j h\left(\frac{x_j}{\bar{x}_j}, 1\right), \tag{38}$$

which is built upon a suitable gradient decomposition $\nabla f_1(x) = V_1(x) - U_1(x), V_1(x), U_1(x) \geq 0$, is a surrogate of the function (35). Our analysis is based on a simple decomposition for matrices whose properties are summarized in the following Lemma. The proof is omitted since it is straightforward.

Lemma 3.2. *Let A be any $\ell \times n$ matrix and define $P, Q \in \mathbb{R}^{\ell \times n}$ as follows:*

$$P_{ij} = \begin{cases} A_{ij} & \text{if } A_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}, \quad Q_{ij} = \begin{cases} -A_{ij} & \text{if } A_{ij} < 0 \\ 0 & \text{otherwise} \end{cases}, \quad \begin{array}{l} i = 1, ..., \ell \\ j = 1, ..., n, \end{array} \tag{39}$$

Therefore we have the decompositions

$$A = P - Q, \quad A^T A = \underbrace{P^T P + Q^T Q}_{\equiv S} - \underbrace{(P^T Q + Q^T P)}_{\equiv R}$$

where the components of each matrix in the summations are nonnegative and, in addition, S, R are symmetric. Moreover, the following properties hold:

1. *$diag(R) = 0$;*
2. *$S + R = \tilde{A}^T \tilde{A}$, where $\tilde{A} = P + Q$, i.e. $\tilde{A}_{ij} = |A_{ij}|$;*
3. *$S + R$ is symmetric, positive semidefinite.*

For sake of simplicity, we consider first the simple case $\ell = 1$, $\phi_1(s) = \frac{1}{2}s$, then we will apply this preliminary result to the general case.

Theorem 3.2. *Let the function $f_1(x)$ be defined as*

$$f_1(x) = \frac{1}{2}\|Ax - b\|^2$$

and define P, Q, S and R as in Lemma 3.2. Moreover, let Σ be the diagonal $\ell \times \ell$ matrix such that $\Sigma_{jj} = sign(b_j)$ and let $\tilde{P}, \tilde{Q} \in \mathbb{R}^{\ell \times n}$ be the two matrices of the decomposition $\Sigma A = \tilde{P} - \tilde{Q}$ defined as in (39). Define the function $G_1(x, \bar{x})$ as in (38), where

$$V_1(x) = 2Sx + \tilde{Q}^T \Sigma b \quad \text{and} \quad U_1(x) = (S + R)x + \tilde{P}^T \Sigma b. \tag{40}$$

Then, $\nabla f_1(x) = V_1(x) - U_1(x)$ and the majorization condition $f_1(x) \leq G(x, \bar{x})$ holds for all positive n-vectors $x, \bar{x}$.

Proof. We first rewrite the decomposition $A^T A = S - R$ as

$$A^T A = S - R = 2S - (S + R). \tag{41}$$

Then, we observe that Σ is an orthogonal symmetric matrix and, therefore, we have the following splitting of $A^T b$ as a difference of two nonnegative vectors:

$$A^T b = A^T \Sigma^T \Sigma b = (\tilde{P} - \tilde{Q})^T \Sigma b = \tilde{P}^T \Sigma b - \tilde{Q}^T \Sigma b. \tag{42}$$

Then, the gradient decomposition (40) directly follows from the above expressions, since $\nabla f_1(x) = A^T A x - A^T b$.

From Lemma 3.2, $S + R$ is symmetric, positive semidefinite. Therefore, for any $x, \bar{x} \in \mathbb{R}^n$ we have

$$0 \leq (x - \bar{x})^T (S + R)(x - \bar{x})$$

which writes also as

$$-x^T (S + R)x \leq \bar{x}^T (S + R)\bar{x} - 2\bar{x}^T (S + R)x.$$

Then, from (41) we obtain

$$\frac{1}{2} x^T A^T A x \quad \leq \quad (x^T S x - \bar{x}^T (S + R)x + \frac{1}{2} \bar{x}^T (S + R)\bar{x}) \quad \stackrel{c}{\leq} \quad x^T S x - \bar{x}^T (S + R)x \tag{43}$$

The quadratic term in the right-hand-side of inequality (43) can be developed as

$$x^T S x = x^T (P^T P + Q^T Q)x = x^T P^T P x + x^T Q^T Q x.$$

Let us consider the term $x^T P^T P x$. For sake of simplicity, we first assume that P has no zero rows (otherwise one should redefine P suppressing the null rows). We have

$$x^T P^T P x = \sum_{i=1}^{\ell} [Px]_i^2 \stackrel{c}{=} 2 \sum_{i=1}^{\ell} h([Px]_i, 2).$$

Then, proceeding as in the proof of Theorem 3.1 (Jensen's inequality), we obtain

$$\begin{aligned}
x^T P^T P x \quad &\stackrel{c}{\leq} \quad 2 \sum_{i=1}^{\ell} \sum_{j=1}^{n} \frac{P_{ij} \bar{x}_j}{[P\bar{x}]_i} h\left(\frac{x_j}{\bar{x}_j} [P\bar{x}]_i, 2 \right) \\
&\stackrel{c}{=} \quad 2 \sum_{i=1}^{\ell} \sum_{j=1}^{n} P_{ij} \bar{x}_j [P\bar{x}]_i h\left(\frac{x_j}{\bar{x}_j}, 2 \right) \\
&= \quad 2 \sum_{j=1}^{n} \left(\sum_{i=1}^{\ell} P_{ij} [P\bar{x}]_i \right) \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 2 \right) \\
&= \quad 2 \sum_{j=1}^{n} [P^T P \bar{x}]_j \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 2 \right)
\end{aligned} \tag{44}$$

Observe that the previous formula holds also if some rows of P are zero: in this case the corresponding value $[P\bar{x}]_i$ is zero and it does not contribute in the summation (44). The same arguments apply also to the term $x^T Q^T Q x$. Therefore, exploiting additivity, we can conclude that

$$x^T S x \overset{c}{\leq} 2\sum_{j=1}^{n} \left(\underbrace{[P^T P\bar{x}]_j + [Q^T Q\bar{x}]_j}_{[S\bar{x}]_j} \right) \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 2\right) \tag{45}$$

Consider now the linear term in (43):

$$\bar{x}^T (S+R)x = \sum_{j=1}^{n} [(S+R)\bar{x}]_j x_j \overset{c}{=} \sum_{j=1}^{n} [(S+R)\bar{x}]_j \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 1\right) \tag{46}$$

Using (42), (45) and (46), we obtain

$$\begin{aligned} f_1(x) &\overset{c}{=} \frac{1}{2} x^T A^T A x - x^T A^T b \\ &\overset{c}{\leq} x^T S x + x^T \tilde{Q}^T \Sigma b - \bar{x}^T (S+R)x - x^T \tilde{P}^T \Sigma b \\ &\overset{c}{\leq} \sum_{j=1}^{n} 2[S\bar{x}]_j \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 2\right) + [\tilde{Q}^T \Sigma b]_j \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 1\right) + \\ &\quad -[(S+R)\bar{x}]_j \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 1\right) - [\tilde{P}^T \Sigma b]_j \bar{x}_j h\left(\frac{x_j}{\bar{x}_j}, 1\right) \end{aligned}$$

Finally, the result follows by using the monotonicity of $h(\cdot,\cdot)$ with respect to its second argument. □

The following corollary is a direct consequence of Theorem 3.2.

Corollary 3.1. *Let $f_1(x)$ be defined as in (35), where $\phi_l(\cdot)$ are differentiable, concave, monotone increasing functions. Moreover, let the matrices $S^{(l)}$, $R^{(l)}$, $\Sigma^{(l)}$, $\tilde{P}^{(l)}$, $\tilde{Q}^{(l)}$ be defined as Theorem 3.2. Consider the function $G_1(x,\bar{x})$ as in (38), where*

$$V_1(x) = 2\sum_{l=1}^{r} \phi_l'(\|A^{(l)} - b^{(l)}\|^2)(2S^{(l)}x + \tilde{Q}^{(l)T}\Sigma^{(l)}b^{(l)}) \tag{47}$$

and

$$U_1(x) = 2\sum_{l=1}^{r} \phi_l'(\|A^{(l)} - b^{(l)}\|^2)((S^{(l)} + R^{(l)})x + \tilde{P}^{(l)T}\Sigma^{(l)}b^{(l)}). \tag{48}$$

Then, $\nabla f_1(x) = V_1(x) - U_1(x)$ and the majorization condition $f_1(x) \leq G(x,\bar{x})$ holds for all positive n-vectors $x, \bar{x}$.

Proof. Applying the chain rule we obtain

$$\nabla f_1(x) = \sum_{l=1}^{r} 2\phi'(\|A^{(l)}x - b^{(l)}\|^2)(A^{(l)T}A^{(l)}x - A^{(l)T}b)$$

and using the decomposition of $A^{(l)T}A^{(l)}$ and $A^{(l)T}b$ directly gives the gradient decomposition (47)–(48). Since ϕ_l is concave the following inequality holds:

$$\phi_l(s) \leq \phi_l(\bar{s}) + \phi_l'(\bar{s})(s - \bar{s}).$$

Moreover, by the monotonicity assumption, we have $\phi_l'(\bar{s}) > 0$ for any $\bar{s}$. Summing up the previous inequalities with $s = \|A^{(l)}x - b^{(l)}\|^2$ and $\bar{s} = \|A^{(l)}\bar{x} - b^{(l)}\|^2$ for $l = 1, ..., r$ gives

$$f_1(x) = \sum_{l=1}^{r} \phi_l(\|A^{(l)}x - b^{(l)}\|^2) \overset{c}{\leq} \sum_{l=1}^{r} \phi_l'(\|A^{(l)}\bar{x} - b^{(l)}\|^2)\|A^{(l)}x - b^{(l)}\|^2.$$

Finally, applying Theorem 3.2 to any term $\|A^{(l)}x - b^{(l)}\|^2$ gives the result. □

Examples. Here we provide the explicit expression of the surrogate for the examples mentioned at the beginning of this section.

The vector $V_1(x)$ for the data discrepancy corresponding to Cauchy noise results in $V_1(\bar{x}) = 2H^T\bar{y}$, where $\bar{y}_i = [H\bar{x}]_i/(\gamma^2 + ([H\bar{x}]_i - g_i)^2$. It is interesting to observe that the corresponding scaling strategy is the same proposed in [21], motivated there only from a SG point of view. The numerical experience in [21] shows the good numerical performances of this choice.

For the Tikhonov regularization of order 1, i.e. for $A^{(l)} = \nabla^{(l)}$ and assuming periodic boundary conditions, it holds that $S^{(l)}$ is a square diagonal matrix of order $N^2 = n$, where the only non-zero entries are: $[S^{(l)}]_{l,l} = 2$, $[S^{(l)}]_{l+1,l+1} = [S^{(l)}]_{l+N,l+N} = 1$, whereas the non-zero entries of $R^{(l)}$ are $[R^{(l)}]_{l,l+1} = [R^{(l)}]_{l+1,l} = 1$ and $[R^{(l)}]_{l,l+N} = [R^{(l)}]_{l+N,l} = 1$. As a consequence, we have

$$\begin{aligned}
[V_1(\bar{x})]_j &= 2[S^{(j)} + S^{(j-1)} + S^{(j-N)}]_{j,j}\bar{x}_j = 8\bar{x}_j \\
[U_1(\bar{x})]_j &= [(S^{(j)} + R^{(j)} + S^{(j-1)} + R^{(j-1)} + S^{(j-N)} + S^{(j-N)})\bar{x}]_j \\
&= 2\bar{x}_j + \bar{x}_{j+1} + \bar{x}_{j+N} + \bar{x}_j + \bar{x}_{j-1} + \bar{x}_j + \bar{x}_{j-N}. \qquad (49)
\end{aligned}$$

For the HS regularization term, in the case of a 2D image and periodic boundary conditions, we obtain that the entries of the terms $V_1(\bar{x})$ and $U_1(\bar{x})$ in $G_1(x, \bar{x})$ can be written as

$$
\begin{aligned}
[V_1(\bar{x})]_j &= \frac{4\bar{x}_j}{\sqrt{Z_j(\bar{x})}} + \frac{2\bar{x}_j}{\sqrt{Z_{j-1}(\bar{x})}} + \frac{2\bar{x}_j}{\sqrt{Z_{j-N}(\bar{x})}}, \\
[U_1(\bar{x})]_j &= \frac{2\bar{x}_j + \bar{x}_{j+1} + \bar{x}_{j+N}}{\sqrt{Z_j(\bar{x})}} + \frac{\bar{x}_j + \bar{x}_{j-1}}{\sqrt{Z_{j-1}(\bar{x})}} + \frac{\bar{x}_j + \bar{x}_{j-N}}{\sqrt{Z_{j-N}(\bar{x})}}.
\end{aligned} \tag{50}
$$

where $Z_l(x) = \|A^{(l)}x\|^2 + \delta^2$. This splitting is very similar to the one derived in [36] from a pure SG strategy.

The definition of a surrogate for the HS/TV function has been investigated by several authors (see for example [33, 37, 38]). However, several of the proposed surrogate are not separable, and this can introduce some difficulties in the implementation of the corresponding MM method.

For the general MRF functional (37), rearranging the terms in the definition of the surrogate, we obtain the following expression:

$$
\begin{aligned}
G_1(x,\bar{x}) \;=\; & \sum_{j=1}^{n} \bar{x}_j \underbrace{2\bar{x}_j \left(\sum_{u\in\mathcal{N}_j} \left(\frac{1}{\epsilon_{j,u}^2 \sqrt{Z_{j,u}(\bar{x})}} + \frac{1}{\epsilon_{u,j}^2 \sqrt{Z_{u,j}(\bar{x})}} \right) \right)}_{\equiv [V_1(\bar{x})]_j} h\left(\frac{x_j}{\bar{x}_j}, 2\right) + \\
& - \bar{x}_j \underbrace{\left(\sum_{u\in\mathcal{N}_j} (\bar{x}_j + \bar{x}_u) \left(\frac{1}{\epsilon_{j,u}^2 \sqrt{Z_{j,u}(\bar{x})}} + \frac{1}{\epsilon_{u,j}^2 \sqrt{Z_{u,j}(\bar{x})}} \right) \right)}_{\equiv [U_1(\bar{x})]_j} h\left(\frac{x_j}{\bar{x}_j}, 1\right),
\end{aligned} \tag{51}
$$

where $Z_{u,i}(x) = (x_u - x_i)^2/\epsilon_{u,i}^2$. □

We conclude this section by observing that many image restoration problems can be written as

$$
\min_{x\geq 0} \Phi(x) \equiv f_0(x) + \mu f_1(x) \tag{52}
$$

where the fit-to-data term f_0 and the regularization function f_1 can be represented as in (20) and (35) respectively, while μ is a nonnegative parameter. The combination of the material presented in this section with that in Section 3.1 allows to define the following surrogate for the objective function Φ:

$$
G(x,\bar{x}) = \sum_{j=1}^{n} \bar{x}_j [V(\bar{x})]_j h\left(\frac{x_j}{\bar{x}_j}, \rho_{max}\right) - \bar{x}_j [U(\bar{x})]_j h\left(\frac{x_j}{\bar{x}_j}, \rho_{min}\right) \tag{53}
$$

where

$$
\rho_{max} = \max\{\zeta_1, ..., \zeta_p, 2\}, \quad \rho_{min} = \max\{\zeta_1, ..., \zeta_p, 1\}
$$

and

$$V(x) = V_0(x) + \mu V_1(x), \quad U(x) = U_0(x) + \mu U_1(x)$$

Clearly, the surrogate (53) is based on the gradient decomposition $\nabla\Phi(x) = V(x) - U(x)$, which is uniquely defined by (29) and (47)–(48).

Moreover, the multiplicative MM algorithm associated to (53) writes as

$$x^{(k+1)} = x^{(k)} \cdot \left(\frac{U(x^{(k)})}{V(x^{(k)})} \right)^{\rho}, \quad \rho = \frac{1}{\rho_{\max} - \rho_{\min}}. \tag{54}$$

and, when $\rho = 1$, it corresponds to a scaled gradient method. Motivated by the same consideration reported at the end of Section 3.1, we then propose the following scaling strategy (combined by a thresholding procedure) for problem (52)

$$D^{(k)^{-1}} = \operatorname{diag}\left(\rho \frac{x^{(k)}}{V(x^{(k)})} \right). \tag{55}$$

265 In Table 1 we summarize the ingredients to build the matrix (55) for the different fit-to-data and regularization functions mentioned in the previous sections.

3.3. Adaptations and applications

The scaling matrix choice (55) provides the inspiration for defining a scaling matrix to be employed in the FB iteration (7). However, this idea must be better refined. First of all, if the steplength parameters λ_k, α_k are allowed to take values different than one, the strict positivity of the iterates is no more guaranteed. This implies that, at some iterate k, the point $x^{(k)}$ may have some zero components. Therefore, $D^{(k)}$ in (55) is not even defined. Moreover, as we mentioned in the introduction, the convergence method (7) is proved under the condition that the eigenvalues of the scaling matrices sequence are bounded or converge to a fixed value at a given rate, regardless of how these matrices are defined.

Therefore, a simple and practical way to overcome these issues consists in thresholding the components in (55), leading to the following rule for scaling matrix selection

$$D^{(k)^{-1}} = \operatorname{diag}\left(\max\left(\min\left(\rho \frac{x_i^{(k)}}{V(x_i^{(k)})}, L_k \right), \frac{1}{L_k} \right) \right), \tag{56}$$

where $L_k \geq 1$ are given scalar parameters. In particular, (56) can be framed in the convergence analysis developed in [10, 21] if $L_k \equiv L$, for some $L \geq 1$, for all $k \geq 0$. Moreover,

when Φ is convex, the convergence conditions proposed in [13, 23, 15, 20] are fulfilled if $L_k^2 = 1 + \mathcal{O}(1/k^2)$, whose practical realization is

$$L_k = \sqrt{1 + \frac{a}{(k+1)^2}}, \tag{57}$$

for some fixed $a \geq 0$.

3.3.1. Variable metric FB methods with extrapolation

Variable metric techniques can be included also in the inertial/extrapolated version of FB methods, whose iteration can be written as

$$\begin{aligned} z^{(k)} &= \mathrm{P}_{Y,D^{(k)}}(x^{(k)} + \beta_k(x^{(k)} - x^{(k-1)})) \\ x^{(k+1)} &= \mathrm{prox}_{\gamma_k\Psi,D^{(k)}}(z^{(k)} - \gamma_k {D^{(k)}}^{-1}\nabla\Phi(z^k)), \end{aligned} \tag{58}$$

where Y denotes a closed convex set where $\nabla\Phi$ is Lipschitz continuous and contains $\mathrm{dom}(\Psi)$. Method (58) is a variant of FISTA [24, 39, 40, 41] and it applies to (4) restricted to the case when Φ is convex with Lipschitz-continuous gradient. As far as its parameters, γ_k can be adaptively computed via a backtracking procedure, while the parameter of the inertial step is defined as $\beta_k = \frac{t_{k-1}-1}{t_k}$ for $k > 0$ and $\beta_0 = 0$, with $\{t_k\}_{k\in\mathbb{N}}$ satisfying the condition $t_{k-1}^2 + t_k - t_k^2 \geq 0$, $t_k \geq 1$.

Convergence of the iterates (58) can be proved under the condition (57), and, in this case, the convergence rate of the objective function values is $\mathcal{O}\left(\frac{1}{k^2}\right)$ [17, 18]. Therefore, the same scaling techniques designed for (7) can be applied also to (58). Indeed, the numerical experience in [17] shows that also the extrapolated method can significantly benefit of suitable variable metric strategies.

4. Numerical experiments

In this section, we evaluate the impact of scaling strategies on the practical performance of the FB method (7) in solving both convex and nonconvex optimization problems of the form (52) arising from imaging real-life applications. Before detailing such applications, we present the setting employed in our implementation of the algorithm (7) for the parameters λ_k and α_k.

As concerns the iteration (7), we adopt the approach proposed in [10, 13, 15] where λ_k is adaptively computed by means of a backtracking procedure based on an Armijo acceptance

condition. For clarity of exposure, we resume in the following proposition the convergence results of the considered approach. For the proof of this proposition, refer to [15]. More precisely, in this paper also the special case where the proximal step is inexactly computed is considered. Furthermore, under very similar assumptions, in [21] convergence results for a non convex Φ are obtained.

Proposition 4.1. *Let assume $\{x^{(k)}\}$ be the sequence generated by the iteration (7). Under the following assumptions*

- $\alpha_k \in [\alpha_{min}, \alpha_{max}]$, $0 < \alpha_{min}$,
- $\{D^{(k)}\}$ *is a sequence of symmetric positive definite matrices with bounded eigenvalues:* $\frac{1}{L} \leq \delta_i(D^{(k)}) \leq L$, $i = 1, ..., n$, *for all* $k \geq 0$, $L \geq 1$,
- λ_k *satisfies the condition*

$$F(x^{(k+1)} + \lambda_k d^{(k)}) \leq F(x^{(k)}) + c\lambda_k(\nabla\Phi(x^{(k)})^T d^{(k)} + \frac{\gamma}{\alpha_k}\|d^{(k)}\|_{D_k} + \Psi(x^{(k)} + d^{(k)}) - \Psi(x^{(k)}))$$

with $c \in (0, 1)$ *and* $\gamma \in [0, 1]$.

Then, a limit point of $\{x^{(k)}\}$ is a stationary point for the problem (4).
Furthermore, under the following additional assumptions:

- Φ *is a convex function and the solution set of (4) is not empty,*
- *the sequence $\{D^{(k)}\}$ satisfies the following additional condition*

$$D^{(k+1)} \preceq (1 + \xi_k)D^{(k)}, \qquad \xi_k \geq 0, \qquad \sum_{k=0}^{\infty} \xi_k < \infty \quad ,$$

then, the sequence $\{x^{(k)}\}$ converges to a solution x^* of the problem (4).
Furthermore, when $\nabla\Phi$ is Lipschitz-continuous, we have that $F(x^{(k)}) - F(x^*) = \mathcal{O}(\frac{1}{k})$.

The choice of the other steplength parameter, α_k, is crucial to obtain good performances. The non-restrictive hypothesis on α_k allows to select it by means of strategies known in the literature to accelerate the performance of standard first order methods. In particular, we mention the well known Barzilai-Borwein rules proposed in the seminal paper [42], which gave rise to a variety of further studies (see for example [43, 44, 45, 46, 47, 48]), to the more recent variants and adaptations [49, 50]. Here we adopt a variant of the rules proposed in [50], which takes into account both the presences of constraints and of a nontrivial scaling

matrix multiplying the gradient. More precisely, given the scaling matrix $D^{(k)}$, at each iteration (7) the following two values are computed

$$\begin{aligned} \alpha_k^{BB1} &= \underset{\alpha\in\mathbb{R}}{\operatorname{argmin}} \ \|\alpha^{-1} s^{(k-1)}_{\mathcal{I}_{k-1}} - ({D^{(k)}}^{-1})_{\mathcal{I}_{k-1},\mathcal{I}_{k-1}} y^{(k-1)}_{\mathcal{I}_{k-1}}\|_{(D^{(k)})_{\mathcal{I}_{k-1},\mathcal{I}_{k-1}}} \\ \alpha_k^{BB2} &= \underset{\alpha\in\mathbb{R}}{\operatorname{argmin}} \ \|s^{(k-1)}_{\mathcal{I}_{k-1}} - \alpha({D^{(k)}}^{-1})_{\mathcal{I}_{k-1},\mathcal{I}_{k-1}} y^{(k-1)}_{\mathcal{I}_{k-1}}\|_{(D^{(k)})_{\mathcal{I}_{k-1},\mathcal{I}_{k-1}}}. \end{aligned} \tag{59}$$

where $s^{(k-1)} = x^{(k)} - x^{(k-1)}$, $y^{(k-1)} = \nabla\Phi(x^{(k)}) - \nabla\Phi(x^{(k-1)})$ and $\mathcal{I}_{k-1}$ is a set of indexes defined as $\mathcal{I}_{k-1} = \mathcal{N} - \mathcal{J}_{k-1}$, being $\mathcal{N} = \{1, ..., n\}$, $\mathcal{J}_{k-1} = \{i \in \mathcal{N} : (x_i^{(k-1)} = 0 \wedge x_i^{(k)} = 0)\}$ (the notation $s_{\mathcal{I}}$ indicates the vector obtained by the components of s whose index is in the set $\mathcal{I}$ and $D_{\mathcal{I},\mathcal{I}}$ is the submatrix of D defined by the intersection of rows and columns with indexes in $\mathcal{I}$). The formulas above impose a quasi-Newton condition on the matrix $\alpha({D^{(k)}}^{-1})_{\mathcal{I}_{k-1},\mathcal{I}_{k-1}}$, which is the submatrix of $D^{(k)}$ restricted to the constraints which have been inactive in the last two iterations. An easy computation shows that the above formulas write as

$$\begin{aligned} \alpha_k^{BB1} &= \frac{{s^{(k-1)}}^T D^{(k)} s^{(k-1)}}{{s^{(k-1)}}^T y^{(k-1)}} \\ \alpha_k^{BB2} &= \frac{{s^{(k-1)}_{\mathcal{I}_{k-1}}}^T y^{(k-1)}_{\mathcal{I}_{k-1}}}{{y^{(k-1)}_{\mathcal{I}_{k-1}}}^T ({D^{(k)}}^{-1})_{\mathcal{I}_{k-1},\mathcal{I}_{k-1}} y^{(k-1)}_{\mathcal{I}_{k-1}}}. \end{aligned} \tag{60}$$

Then, the value of α_k is computed by alternating the two values as described in Algorithm ABB$_{min}$, where α_{min} and α_{max}, with $0 < \alpha_{min} \leq \alpha_{max}$ are thresholding constants. More details about stepsize selection can be found in [49, 50].

All the numerical experiments have been carried out in a Matlab® R2016a environment and run on a PC equipped with a 1.60 GHz Intel Core i7 in a Windows 7 environment. The MATLAB routines are available at the website http://www.oasis.unimore.it/site/home/software.html.

4.1. Convex framework

We consider two applications, the first one is the edge-preserving image deconvolution with data corrupted by Poisson noise whereas the second one is the reconstruction of images from low sampling acquisition in Computed Tomography. The model problem is (52) where $f_0(x) = \mathcal{D}(Hx; g)$ is a convex fit-to-data term, f_1 is the HS regularization function described in Section 3.2 and μ is the regularization parameter balancing the weight of the regularization term f_1. The matrix H represents, in the first case, the convolution operator and a

Algorithm ABB$_{min}$ for steplength selection

IF $k = 0$

set $\alpha_0 \in [\alpha_{min}, \alpha_{max}]$, $\tau_1 \in (0,1)$, $\nu > 1$, $M_\alpha \in \mathbb{N}$;

ELSE

Compute α^{BB1} and α_{BB2} as in (60)

IF $\alpha_k^{BB1} <= 0$ THEN

$\alpha_k^{(1)} = \alpha_{max}$;

$\alpha_k^{(2)} = \alpha_{max}$;

ELSE

$\alpha_k^{(1)} = \max\left\{\alpha_{min}, \min\left\{\alpha_k^{BB1}, \alpha_{max}\right\}\right\}$;

$\alpha_k^{(2)} = \max\left\{\alpha_{min}, \min\left\{\alpha_k^{BB2}, \alpha_{max}\right\}\right\}$;

END

IF $\frac{\alpha_k^{(2)}}{\alpha_k^{(1)}} \leq \tau_k$

$\alpha_k = \min_{j=\max\{1,k-M_\alpha\},\ldots,k}\left\{\alpha_j^{(2)}\right\}$;

$\tau_{k+1} = \tau_k/\nu$;

ELSE

$\alpha_k = \alpha_k^{(1)}$;

$\tau_{k+1} = \nu\tau_k$;

END

END

discrete approximation of the CT acquisition system in the second application. In both cases, its components are nonnegative. Since the nonsmooth part of the problem consists in nonnegativity constraints, method (7) actually is a scaled gradient projection method, where the projection onto the constraint set can be computed in a straightforward way. In this framework, we denote by SGP the algorithm (7) applied to problem (52), equipped with the steplength selection rules described in Algorithm ABB_{min} (with $\alpha_{min} = 10^{-5}$, $\alpha_{max} = 10^5$, $\tau_1 = 0.5$, $M_\alpha = 3$, $\nu = 1.1$) and $D^{(k)}$ chosen as in (56). Moreover, since the problem is convex, we adopt the thresholding strategy (56)-(57) to ensure convergence (with $\rho = 1$). The nonscaled version of the same algorithm, i.e. $D^{(k)} = I$ for all $k \geq 0$ is referred as GP.

We compare the SGP scheme with the inertial method (58) in the implementation described in [17, 18]. In particular, we set $\beta_0 = 0$ and $\beta_k = \frac{k-1}{k-2.1}$. The inertial algorithm corresponding to the choice (56)–(57) is denoted by SFBEM, while its nonscaled version is referred as FBEM.

4.1.1. Application 1: deconvolution of images corrupted by Poisson noise

When the data are corrupted by Poisson noise, the data discrepancy is expressed by the (generalized) Kullback-Leibler divergence and the restored image is obtained as an approximation of the solution of the problem

$$\min_{x \geq 0} \sum_{i=1}^{m} \left(g_i \log \frac{g_i}{(Hx + b_g)_i} + (Hx + b_g)_i - g_i \right) + \mu f_1(x), \tag{61}$$

for a suitable value of the regularization parameter μ.

We simulate the blurring effect in data acquisition by convolving a clean image $\tilde{x}$ with a discretized Point Spread Function (PSF); then the blurred images are added by a background constant and perturbed with Poisson noise by the Matlab routine `imnoise`. Periodic boundary conditions are assumed in all cases; as a consequence of this, the convolution operator can be modeled as a $n \times n$ matrix H with a block-circulant-with-circulant-blocks (BCCB) structure and the associated matrix-vector products can be efficiently computed via the Fast Fourier Transform (FFT) algorithm.

In the following we detail the simulated test problems:

- `spacecraft`: the image size is 256×256; its pixels range between 0 and 2550; the object is convolved with a PSF simulating a ground-based telescope (downloaded from

`www.mathcs.emory.edu/nagy/RestoreTools/index.html`); finally, the background emission is simulated by adding to all pixels the constant $b_g = 10$; the relative distance between the original object and the blurred noisy data in ℓ_2 norm is 0.705, the simulated detected data g are in the interval $[5, 1135]$ and the regularization parameter μ is 3.353 10^{-4};

- `tubulins`: the size of the original object representing a micro-tubulin network inside a cell [51] is 512×512; its values are in $[0, 686]$, whereas those of the blurred and noisy image are in $[0, 446]$; the background is set equal to 1 and the relative distance between the original object and the blurred noisy data in ℓ_2 norm is 0.756; μ is set equal to 4 10^{-4}.

The blurred and noisy images for the two test problems are reported in Figure 1.

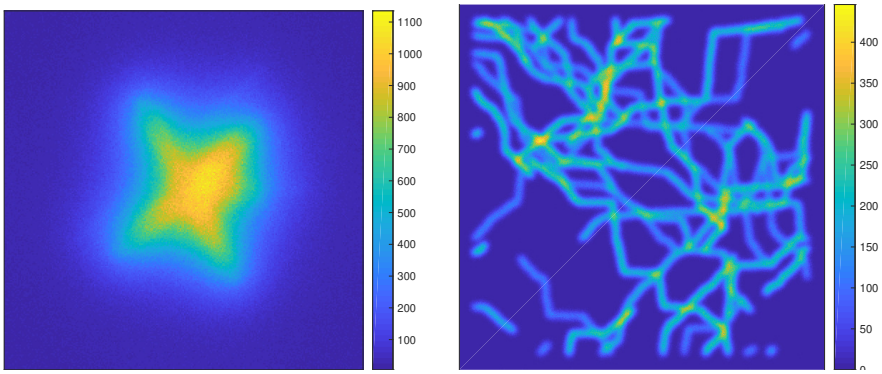

Figure 1: Blurred and noisy images for `spacecraft` (first panel) and `tubulins` (second panel).

We evaluate the effectiveness of the proposed selection of the steplength and the scaling matrix by comparing SGP and SFBEM with the corresponding nonscaled versions, GP and FBEM. In these experiments, the scaling matrix is defined as in (56)-(57), where $V(x^{(k)}) = V_0(x^{(k)}) + \mu V_1(x^{(k)})$, with $V_0(x^{(k)}) = H^T\mathbf{1}$ and $V_1(x^{(k)})$ defined as in (50). The constant a in (57) is set to 10^{10} for both SGP and SFBEM, while the starting value γ_0 is chosen as 0.125 and 2.5 for FBEM and SFBEM respectively.

As further benchmark, we include in our comparison also the multiplicative or MM method (54), where the terms of the gradient splitting are given in (23)-(29) and (50) (for the gradients of the discrepancy function and the HS regularization, respectively).

For all the test problems, the initial iterate $x^{(0)}$ is set as $\max(g, \epsilon)$, where ϵ is the machine precision and the value of δ in the HS term is chosen equal to $10^{-6} \max_i g_i$. For each test problem, we compute an high accuracy numerical approximation x_μ^* of the solution of (61), by running a huge number of iterations of SGP. In order to evaluate the effectiveness of the methods in reducing the objective function and converging to the solution of the minimization problem (61), at any iteration k of each of the considered methods we computed the relative error

$$F_k = \frac{F(x^{(k)}) - F^*}{F^*}, \quad F^* = F(x_\mu^*)$$

and the relative distance

$$e_k = \frac{\|x^{(k)} - x_\mu^*\|_2}{\|x_\mu^*\|_2}$$

with respect to the estimated minimum. Moreover, to evaluate the performance also from the point of view of image restoration, we compute the relative reconstruction error

$$E_k = \frac{\|x^{(k)} - \tilde{x}\|_2}{\|\tilde{x}\|_2}$$

with respect to the original object $\tilde{x}$.

Table 2 shows the numerical results obtained by running the methods until the relative error F_k is less than a prefixed value *tol* or a maximum number of 1500 iterations is reached. In particular, for different values of *tol*, we report the number k of iterations and the computational time (in seconds) needed by each method to satisfy $F_k \leq tol$, the corresponding relative minimization and reconstruction errors e_k, E_k . Figure 2 shows the plot of the sequences F_k, e_k and E_k with respect to the computational time for both test problems. From Table 2 and Figure 2 we can draw the following remarks:

- the variable metric methods have a very effective behaviour in the initial iterations, and this effectiveness is similar for all the schemes;
- according to the theoretical results, the inertial methods exhibits the $o\left(\frac{1}{k^2}\right)$ rate of convergence with respect to the function values; nevertheless, from a practical point of view, the faster convergence rate is really appreciable when high accuracy is required.

As a further consideration, we can observe that a stopping criterion based on the relative difference of the function values at two subsequent iterations may either provide an unsatisfactory solution for the slower methods as GP or induce unnecessary iterations for the faster

ones. Indeed for FBEM and SFBEM, the objective function curve becomes flat only close to the solution.

Figure 3 shows the image provided by the SGP and the SFBEM methods for the `spacecraft` dataset after 5 seconds and for the `tubulins` dataset after 10 seconds.

4.1.2. Application 2: 3D X-ray Computed Tomography

The 3D Computed Tomography (CT) operation process is based on the different levels of X-ray absorption by materials in an object, or tissues in the human body. Essentially, a CT apparatus consists of a X-ray source and a X-ray detector: a cone of X-rays is emitted by the source which rotates around the object of interest from a fixed number of angles. The rays, partially absorbed by the object, are projected on a detector and then recorded. Given the number of angles N_θ , the number of pixels in the detector N_p and the number of voxels of the object n, the image formation model for the X-ray CT can be discretized through the linear model $Hx = g$ where $H \in \mathbb{R}^{(N_p N_\theta)\times n}$ is the matrix describing the system geometry, $x \in \mathbb{R}^n$ is the vector of the linear attenuation coefficients of the object at each voxel and $g \in \mathbb{R}^{N_p N_\theta}$ is the nonnegative vector of the recorded projections.

Recently, low sampling acquisition in CT received growing attention in the medical community with respect to complete sample since the acquisition of a reduced set of data allows to speed-up the imaging process and to increase the patient safety thanks to a low-dose ionizing radiation. However, in this case, $N_p N_\theta < n$; hence the linear system $Hx = g$ has infinite possible solutions. This fact, together with the ill-conditioning of H, makes necessary to employ regularization techniques. In [52, 53], the authors propose to formulate the restoration problem as in (52), where $f_0(x) = \mathcal{D}(Hx + b_g; g)$ is the Kullback-Leibler or the least squares functional, according to the noise statistics, and $f_1(x)$ is the HS regularization.

We generated 4 different test problems by simulating the CT acquisition of the 3D Shepp–Logan phantom, discretized in $n = 61^3 = 226981$ voxels lexicographically ordered in the vector $\tilde{x}$. The projections have been computed as $\tilde{g} = H\tilde{x}$, where H is the projection matrix, obtained with the functions in the TVREG Matlab Toolbox; in particular, H represents a 3D geometry with random angles over an half sphere. The detector pixels are $N_p = 61^2$ while the number of angles N_θ varies in the set $\{19, 37\}$. The projections $\tilde{g}$ have been artificially corrupted by adding Gaussian or Poisson noise. In particular, we denote by

G1 the dataset with $N_\theta = 19$ and Gaussian noise;

G2 the dataset with $N_\theta = 37$ and Gaussian noise;

P1 the dataset with $N_\theta = 19$ and Poisson noise ($b_g = 1$);

P2 the dataset with $N_\theta = 37$ and Poisson noise ($b_g = 1$).

The value of δ in the HS term has been set as $10^{-6} \cdot \max(g)$ for all the datasets, while the regularization parameters μ are set to the following values: 8×10^{-4} for **G1**, 6.5×10^{-2} for **G2** and 3×10^{-2} for both **P1** and **P2**.

We compare the methods described in the previous section with the variable metric defined by (56)-(57) and the same parameters settings, except the following change:

- $L_k = \sqrt{1 + \dfrac{a}{(k+1)^2}}$ with $a = 10^{15}$ for SGP and SFBEM.

For each problem, we first computed a numerical approximation of the minimum F^* and the corresponding solution x^*_μ by running all methods for a huge number of iterations and selecting the output corresponding to the smallest value of the objective function.

Table 4 reports the values of F_k and E_k reached by the algorithms at three different temporal windows: at 5 s (simulating real-time execution), 20 s (over-time of few minutes) and 50 s (off-line execution). The number of iterations is also reported. Figures 4-7 show the behaviour of the sequences F_k and E_k with respect to the computational time for all the test problems considered. Finally, by way of example, Figures 8 and 9 show the reconstruction of the true object provided by GP, SGP and SFBEM at 5 and 20 seconds, for the dataset **G1** and the dataset **P1**, respectively. These figures shows the different quality of the reconstructions provided by scaled and nonscaled methods in a short time. Indeed, scaled methods produce good restorations where the image structure are well reconstructed in few iterations, while nonscaled methods need more computational time to give comparable results. We can also observe that SGP and SFBEM show comparable numerical performance.

4.2. Nonconvex framework: deconvolution of images corrupted by signal dependent Gaussian noise

We consider the image restoration problem described in [12] where the data are corrupted by a signal dependent Gaussian noise. Under this assumption, an estimate of the true image can be computed by solving the minimization problem (52) where f_0 is the data

discrepancy function introduced in (24) and f_1 is the discretization of the TV functional, given by (36) with $\delta = 0$. We point out that the backward step for the algorithm (7) can not be computed in a closed formula, but inexactly. The minimization problem related to the inexact computation of the proximal step has been addressed by the FISTA [24] method which has been stopped thanks to the implementable criterion suggested in [15]. In the notation used in [15] we set $\eta = 10^{-6}$. In this framework, we denote by VMILA the algorithm (7) applied to problem (52), equipped with the steplength selection rules described in Algorithm ABB_{min} (with $\alpha_{min} = 10^{-1}$, $\alpha_{max} = 10$, $\tau_1 = 0.5$, $M_\alpha = 3$, $\nu = 1.1$). The scaling matrix has been selected in two different ways in order to compare the effects of different variable metrics. In particular the following choices for $D^{(k)}$ has been considered.

a) $D^{(k)}$ has been chosen as in (56) with $\rho = \frac{1}{2}$, $L_k \equiv 10^2$, $\forall k$ and $V(x^{(k)}) = V_0(x^{(k)}) = H^T y^{(k)}$ with

$$y_i^{(k)} = (Hx^{(k)})_i \frac{a_i((Hx^{(k)})_i + g_i) + 2b_i}{2(a_i(Hx^{(k)})_i + b_i)^2} + \frac{a_i}{2(a_i(Hx^{(k)})_i + b_i)}.$$

b) $D^{(k)}$ has been fixed as ${D^{(k)}}^{-1} = \text{diag}\left(\max\left(\min\left((A_k)_{ii}, L\right), \frac{1}{L}\right)\right)$, where A_k is defined in [12, formula (36)] with $\epsilon = 0$ and $L = 10^2$.

The nonscaled version of the same algorithm, i.e. $D^{(k)} = I$ for all $k \geq 0$ is referred as ILA. We evaluate the performance of the VMILA and ILA methods in comparison with the variable-metric forward-backward (VMFB) algorithm [12] (the implementation is provided by the authors [54]). In particular, we analyze the test problem jet plane [54]. In Table 4 we report the number of iterations and the computational time needed by each method to satisfy $F_k \leq tol$, the corresponding relative minimization and reconstruction errors e_k, E_k. The notations VMILA_a and VMILA_b indicate the VMILA method equipped with the scaling matrices defined before in a) and b), respectively. Figure 10 shows the behaviour of the relative error on the objective function values with respect to both the iterations number and the computational time and the behaviour of the relative minimization error and the relative reconstruction error. From the analysis of both Table 4 and Figure 10 it is possible to conclude that VMILA and ILA outperform the VMFB algorithm in terms of number of iterations and computational time, by confirming the numerical results shown in [15, 21]. Moreover, by comparing the performance of VMILA with respect to ILA, the

benefits gained thanks to the presence of the first scaling matrix is quite evident, especially in the first part of the iterative process. On the other hand, the second scaling matrix seems to not guarantee the same effect.

In conclusion, the numerical experience, carried out in both the convex and the nonconvex framework, shows that

- variable metric methods based on the proposed scaling technique are valid tools for image reconstruction, especially when a medium accuracy solution is needed in a short time, as in real-time applications;
- the suggested approach to define scaling matrices for forward-backward algorithms is effective, quite general and it can be applied to many optimization problems arising from imaging applications.

5. Conclusions and future work

In this paper, combining the Split Gradient and Majorization-Minimization ideas, we proposed some new strategies to choose the scaling matrix for nonnegatively constrained problems, in the framework of variable metric FB methods. Our strategy employs only the first order information, and leads to a diagonal scaling matrix, therefore it does not require significant additional computations. The effectiveness of the proposed approach has been validated on image restoration problems. We believe that interesting issues to be investigated for future work concern the extension of this ideas to more general problems involving different kinds of constraints and objective functions.

Acknowledgements. This work was partially supported by the INdAM-GNCS institute.

References

[1] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1984) 721–741.

[2] M. Bertero, P. Boccacci, V. Ruggiero, Inverse Imaging with Poisson Data, 2053-2563, IOP Publishing, 2018. doi:10.1088/2053-2563/aae109.
URL http://dx.doi.org/10.1088/2053-2563/aae109

[3] P. Charbonnier, L. Blanc-Féraud, G. Aubert, M. Barlaud, Deterministic edge-preserving regularization in computed imaging, IEEE Trans. Image Proc. 6 (1997) 298–311.

[4] P. L. Combettes, J.-C. Pesquet, Proximal splitting methods in signal processing, in: H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, H. Wolkowicz (Eds.), Fixed-point algorithms for inverse problems in science and engineering, Springer Optimization and Its Applications, Springer, New York NY, 2011, pp. 185–212.

[5] P. L. Combettes, V. R. Wajs, Signal recovery by proximal forward-backward splitting, SIAM Multiscale Model. Simul. 4 (2005) 1168–1200.

[6] E. S. Levitin, B. T. Polyak, Constrained minimization methods, U.S.S.R. Comput. Math. Math. Phys. 6 (1966) 1–50.

[7] L. B. Lucy, An iterative technique for the rectification of observed distributions, Astronom. J. 79 (1974) 745–754.

[8] D. S. C. Biggs, M. Andrews, Acceleration of iterative image restoration algorithms, Appl. Optics 36 (1997) 1766–1775.

[9] H. Lantéri, M. Roche, O. Cuevas, C. Aime, A general method to devise maximum-likelihood signal restoration multiplicative algorithms with nonnegativity constraints, Signal Process. 81 (2001) 945–974.

[10] S. Bonettini, R. Zanella, L. Zanni, A scaled gradient projection method for constrained image deblurring, Inverse Problems 25 (1) (2009) 015002 (23pp).

[11] S. Bonettini, G. Landi, E. Loli Piccolomini, L. Zanni, Scaling techniques for gradient projection-type methods in astronomical image deblurring, International Journal of Computer Mathematics 90 (1) (2013) 9–29.

[12] E. Chouzenoux, J.-C. Pesquet, A. Repetti, Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function, J. Optim. Theory and Appl. 162 (1) (2014) 107–132.

[13] S. Bonettini, M. Prato, New convergence results for the scaled gradient projection method, Inverse Problems 31 (9) (2015) 095008.
URL http://stacks.iop.org/0266-5611/31/i=9/a=095008

[14] E. Chouzenoux, J.-C. Pesquet, A. Repetti, A block coordinate variable metric forward–backward algorithm, J. Glob. Optim. 66 (3) (2016) 457–485.

[15] S. Bonettini, I. Loris, F. Porta, M. Prato, Variable metric inexact line–search based methods for nonsmooth optimization, SIAM Journal on Optimization 26 (2016) 891–921.

[16] S. Bonettini, A. Benfenati, V. Ruggiero, Scaling techniques for ϵ-subgradient methods, SIAM Journal on Optimization 26 (3) (2016) 1741–1772.

[17] S. Bonettini, F. Porta, V. Ruggiero, A variable metric forward–backward method with extrapolation, SIAM J. Sci. Comput. 38 (4) (2016) A2558–A2584.

[18] S. Bonettini, S. Rebegoldi, V. Ruggiero, Inertial variable metric techniques for the inexact forward–backward algorithm, SIAM Journal on Scientific Computing 40 (5) (2018) A3180–A3210.

[19] P. L. Combettes, B. C. Vũ, Variable metric quasi-Féjer monotonicity, Nonlinear Analysis: Theory, Methods, and Applications 78 (2013) 17–31.

[20] P. L. Combettes, B. C. Vũ, Variable metric forward-backward splitting with applications to monotone inclusions in duality, Optimization 63 (9) (2014) 1289–1318.

[21] S. Bonettini, I. Loris, F. Porta, M. Prato, S. Rebegoldi, On the convergence of a linesearch based proximal-gradient method for nonconvex optimization, Inverse Problems 55 (5) (2017) 055005.

[22] P. Frankel, G. Garrigos, J. Peypouquet, Splitting methods with variable metric for Kurdyka-Lojasiewicz functions and general convergence rates, J. Optim. Theory Appl. 165 (2015) 874–900.

[23] S. Salzo, The variable metric forward-backward splitting algorithm under mild differentiability assumptions, SIAM J. Optim. 27 (4) (2017) 2153–2181.

[24] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM Journal on Imaging Sciences 2 (2009) 183–202.

[25] F. Porta, M. Prato, L. Zanni, A new steplength selection for scaled gradient methods with application to image deblurring, J. Sci. Comp. 65 (2015) 895–919.

[26] H. Lantéri, M. Roche, C. Aime, Penalized maximum likelihood image restoration with positivity constraints: multiplicative algorithms, Inverse Problems 18 (2002) 1397–1419.

[27] Z. Yang, E. Oja, Unified development of multiplicative algorithms for linear and quadratic nonnegative factorization, IEEE Trans. on Neural Networks 22 (12) (2011) 1878–1891.

[28] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature (1999) 788–791.

[29] M. E. Daube-Witherspoon, G. Muehllehner, An iterative image space reconstruction algorithm suitable for volume ect, IEEE Trans. Med. Imaging 5 (1986) 61–66.

[30] W. H. Richardson, Bayesian–based iterative method of image restoration, J. Opt. Soc. Amer. A 62 (1) (1972) 55–59.

[31] M. Razaviyayn, M. Hong, Z.-Q. Luo, A unified convergence analysis of block successive minimization methods for nonsmooth optimization, SIAM J. Optim. 23 (2) (2013) 1126–1153.

[32] H. Erdogan, J. Fessler, Monotonic algorithms for transmission tomography., IEEE Trans. Med. Imag. 18 (9) (1999) 801–814.

[33] E. Chouzenoux, J.-C. Pesquet, A stochastic majorize-minimize subspace algorithm for online penalized least squares estimation, IEEE Transactions on Signal Processing 65 (18) (2017) 4770–4783.

[34] A. R. De Pierro, On the convergence of an EM–type algorithm for penalized likelihood estimation in emission tomography, IEEE Trans. Med. Imag. 14 (4) (1995) 762–765.

[35] F. Sciacchitano, Y. Dong, T. Zeng, Variational approach for restoring blurred images with Cauchy noise, SIAM J. Imaging Sci. 8 (2015) 1894–1922.

[36] R. Zanella, P. Boccacci, L. Zanni, M. Bertero, Efficient gradient projection methods for edge-preserving removal of Poisson noise, Inverse Problems 25 (4) (2009) 045010.

[37] J. P. Oliveira, J. M. Bioucas-Dias, M. A. T. Figueiredo, Adaptive total variation image deblurring: a majorization-minimization approach, Signal Process. 89 (9) (2009) 1683–1693.

[38] T. F. Chan, P. Mulet, On the convergence of the lagged diffusivity fixed point method in total variation image restoration, SIAM J. Numer. Anal. 36 (2) (1999) 354–367.

[39] Y. Nesterov, Introductory lectures on convex optimization : a basic course, Applied optimization, Kluwer Academic Publ., Boston, Dordrecht, London, 2004.

[40] A. Chambolle, C. Dossal, On the convergence of the iterates of the "Fast Iterative Shrinkage/Thresholding Algorithm", J. Optim. Theory Appl. 166 (3) (2015) 968–982. doi:10.1007/s10957-015-0746-4.

[41] H. Attouch, J. Peypouquet, The Rate of Convergence of Nesterov's Accelerated Forward-Backward Method is Actually Faster than $1/k^2$, SIAM J. Optim. 26 (3) (2016) 1824–1834.

[42] J. Barzilai, J. M. Borwein, Two-point step size gradient methods, IMA J. Numer. Anal. 8 (1988) 141–148.

[43] A. Friedlander, J. M. Martínez, B. Molina, M. Raydan, Gradient Method with Retards and Generalizations, SIAM J. Numer. Anal. 36 (1999) 275–289.

[44] Y. H. Dai, R. Fletcher, On the asymptotic behaviour of some new gradient methods, Math. Programming 103 (2005) 541–559.

[45] Y. H. Dai, W. H. Hager, K. Schittkowski, H. Zhang, The cyclic Barzilai-Borwein method for unconstrained optimization, IMA J. Numer. Anal. 26 (2006) 604–627.

[46] B. Zhou, L. Gao, Y. H. Dai, Gradient methods with adaptive step-sizes, Comput. Optim. Appl. 35 (1) (2006) 69–86.

[47] G. Frassoldati, G. Zanghirati, L. Zanni, New adaptive stepsize selections in gradient methods, J. Indutrial and Management Optimization 4 (2) (2008) 299–312.

[48] A. De Asmundis, D. di Serafino, H. Hager, G. Toraldo, H. Zhang, An efficient gradient method using the Yuan steplength, Comput. Optim. Appl. 59 (3) (2014) 541–563.

[49] D. di Serafino, V. Ruggiero, G. Toraldo, L. Zanni, On the steplength selection in gradient methods for unconstrained optimization, Appl. Math. Comput. 318 (2018) 176–195.

[50] S. Crisci, V. Ruggiero, L. Zanni, Steplength selection in gradient projection methods for boxconstrained quadratic programs, Applied Mathematics and Computation 356 (1) (2019) 312–327.

[51] F. Porta, R. Zanella, G. Zanghirati, L. Zanni, Limited-memory scaled gradient projection methods for real-time image deconvolution in microscopy, Communications in Nonlinear Science and Numerical Simulation 21 (2015) 112–127.

[52] M. Beister, D. Kolditz, W. A. Kalender, Iterative reconstruction methods in X-ray CT, Physica Medica 28 (2012) 94–108.

[53] E. Loli Piccolomini, V. L. Coli, E. Morotti, L. Zanni, Reconstruction of 3d x-ray ct images from reduced sampling by a scaled gradient projection algorithm, Comput. Optim. Appl. 71 (2018) 171–191.

[54] A. Repetti, E. Chouzenoux, RestoVMFB_Lab : Matlab toolbox for image restauration with the Variable Metric Forward-Backward algorithm, http://www-syscom.univ-mlv.fr/~chouzeno/Logiciel.html (2013).

Fit-to-data functions	$V_0(x^{(k)})$	Name
$\frac{1}{2}\|Hx-g\|^2$	$H^T H x^{(k)}$	*Least-squares discrepancy (Gaussian noise)*
$\sum_{i=1}^{m} g_i \log \frac{g_i}{[Hx]_i} + [Hx]_i - g_i$	$H^T \mathbf{1}$	*Kullback-Leibler divergence (Poisson noise)*
$\frac{1}{2}\sum_{i=1}^{m} \frac{([Hx]_i - g_i)^2}{a_i[Hx]_i + b_i} + \log(a_i[Hx]_i + b_i)$	$H^T y^{(k)}$, $y_i^{(k)} = [Hx^{(k)}]_i \frac{a_i([Hx^{(k)}]_i + g_i) + 2b_i}{2(a_i[Hx^{(k)}]_i + b_i)^2} + \frac{a_i}{2(a_i[Hx^{(k)}]_i + b_i)}$	*Data discrepancy for signal dependent Gaussian noise*
$\sum_{i=1}^{m} \log\left(\gamma^2 + ([Hx]_i - g_i)^2\right)$	$2H^T y^{(k)}$, $y_i^{(k)} = \frac{[Hx^{(k)}]_i}{\gamma^2 + ([Hx^{(k)}]_i - g_i)^2}$	*Data discrepancy for Cauchy noise*

Regularization functions	$V_1(x^{(k)})$	Name
$\sum_{l=1}^{n} \frac{1}{2}\|\nabla^{(l)} x\|^2$	$8x^{(k)}$	*Tikhonov regularization of order 1*
$\sum_{l=1}^{n} \sqrt{\|\nabla^{(l)} x\|^2 + \delta^2}$	$[V_1(x^{(k)})]_j = \frac{4x_j^{(k)}}{\sqrt{Z_j(x^{(k)})}} + \frac{2x_j^{(k)}}{\sqrt{Z_{j-1}(x^{(k)})}} + \frac{2x_j^{(k)}}{\sqrt{Z_{j-N}(x^{(k)})}}$, $Z_l(x) = \|\nabla^{(l)} x\|^2 + \delta^2$	*Hyper-surface (HS) regularization*
$\sum_{l=1}^{n} \sqrt{\sum_{u\in\mathcal{N}_l} \left(\frac{(x_u - x_l)^2}{\epsilon_{u,l}}\right) + \delta^2}$	$[V_1(x^{(k)})]_j = 2x_j^{(k)} \sum_{u\in\mathcal{N}_j} \frac{1}{\epsilon_{j,u}^2 \sqrt{Z_{j,u}(x^{(k)})}} + \frac{1}{\epsilon_{u,j}^2 \sqrt{Z_{u,j}(x^{(k)})}}$, $Z_{u,i}(x) = (x_u - x_i)^2/\epsilon_{u,i}^2$	*Markov Random Field (MRF) regularization*

Table 1: Elements to build the scaling matrix (55) for different fit-to-data and regularization functions frequently used in imaging ($\rho = 1$ in all cases except for signal dependent Gaussian noise where $\rho = \frac{1}{2}$).

Table 2: Computational results for the HS edge-preserving regularization problem.

`spacecraft`								
	$F_k \leq 0.05$				$F_k \leq 0.005$			
method	k	time	E_k	e_k	k	time	E_k	e_k
MM	221	1.50	0.41	0.29	1500	10.10	0.34	0.17
GP	248	1.77	0.44	0.34	1500	10.68	0.37	0.23
SGP	34	0.24	0.41	0.30	125	0.91	0.33	0.17
FBEM	81	0.73	0.44	0.34	194	1.77	0.35	0.19
SFBEM	30	0.29	0.41	0.30	98	0.89	0.33	0.16
`tubulins`								
	$F_k \leq 0.05$				$F_k \leq 0.001$			
method	k	time	E_k	e_k	k	time	E_k	e_k
MM	101	2.99	0.53	0.35	1500	43.80	0.44	0.18
GP	381	11.40	0.56	0.39	1500	45.97	0.52	0.33
SGP	27	0.83	0.49	0.29	266	8.12	0.44	0.14
FBEM	349	13.93	0.57	0.41	1500	59.00	0.45	0.17
SFBEM	20	0.86	0.54	0.37	122	4.83	0.44	0.13

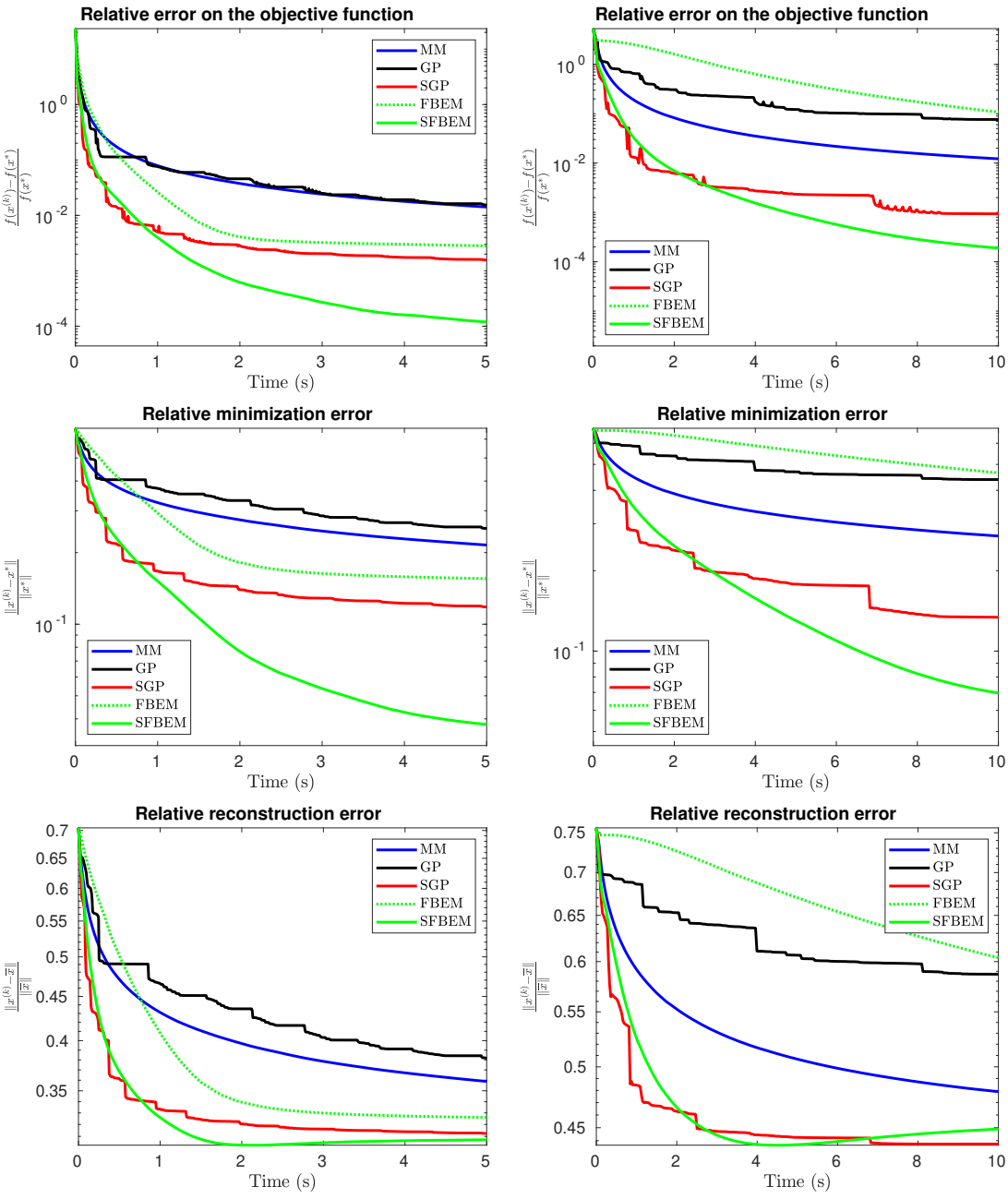

Figure 2: Behaviour of the relative error on the function values (first row), the minimization error (second row) and the reconstruction error (third row) for `spacecraft` (first column) and `tubulins` (second column) with respect to the execution time for the considered methods in solving the HS edge-preserving regularization problem.

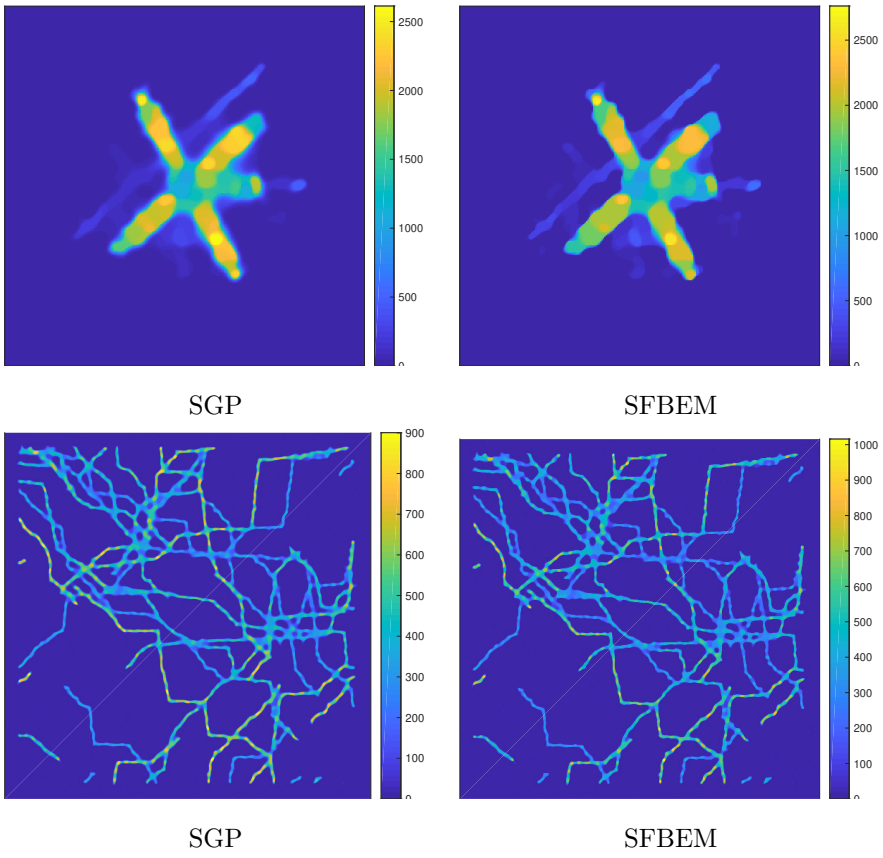

Figure 3: Reconstructed images provided by the SGP (first column) and the SFBEM (second column) methods for the *spacecraft* dataset after 5 seconds (first row) and for the *tubulins* dataset after 10 seconds (second row).

Table 3: Computational results at different times for the X-ray CT problems.

G1									
	5 s			20 s			50 s		
method	k	E_k	F_k	k	E_k	F_k	k	E_k	F_k
MM	22	0.34	84.566	43	0.26	25.177	205	0.17	1.8450
GP	19	0.35	32.156	37	0.24	3.356	185	0.16	0.0566
SGP	20	0.22	7.376	38	0.16	0.830	186	0.12	0.0287
FBEM	23	0.34	28.718	43	0.24	3.540	205	0.16	0.0521
SFBEM	21	0.20	4.686	40	0.16	0.726	190	0.12	0.0276
G2									
	5 s			20 s			50 s		
method	k	E_k	F_k	k	E_k	F_k	k	E_k	F_k
MM	21	0.31	5.326	41	0.21	1.554	199	0.11	0.1071
GP	19	0.28	1.412	37	0.18	0.222	177	0.12	0.0225
SGP	19	0.13	0.231	37	0.11	0.086	178	0.10	0.0002
FBEM	22	0.30	2.008	41	0.17	0.225	194	0.10	0.0012
SFBEM	20	0.15	0.280	38	0.10	0.058	182	0.10	0.0005
P1									
	5 s			20 s			50 s		
method	k	E_k	F_k	k	E_k	F_k	k	E_k	F_k
MM	23	0.29	1.265	44	0.22	0.339	212	0.12	0.0153
GP	20	0.36	1.329	39	0.35	1.288	190	0.22	0.1228
SGP	20	0.19	0.095	39	0.12	0.012	190	0.10	0.0004
FBEM	17	0.74	60.516	37	0.61	15.189	196	0.22	0.1680
SFBEM	20	0.19	0.113	39	0.12	0.012	185	0.10	0.0003
P2									
	5 s			20 s			50 s		
method	k	E_k	F_k	k	E_k	F_k	k	E_k	F_k
MM	21	0.27	2.438	41	0.18	0.598	201	0.09	0.0211
GP	19	0.38	3.713	37	0.23	0.632	181	0.14	0.0815
SGP	20	0.14	0.127	37	0.08	0.010	181	0.07	0.0002
FBEM	12	0.79	160.250	30	0.64	33.470	176	0.19	0.3084
SFBEM	18	0.16	0.246	35	0.09	0.020	172	0.07	0.0005

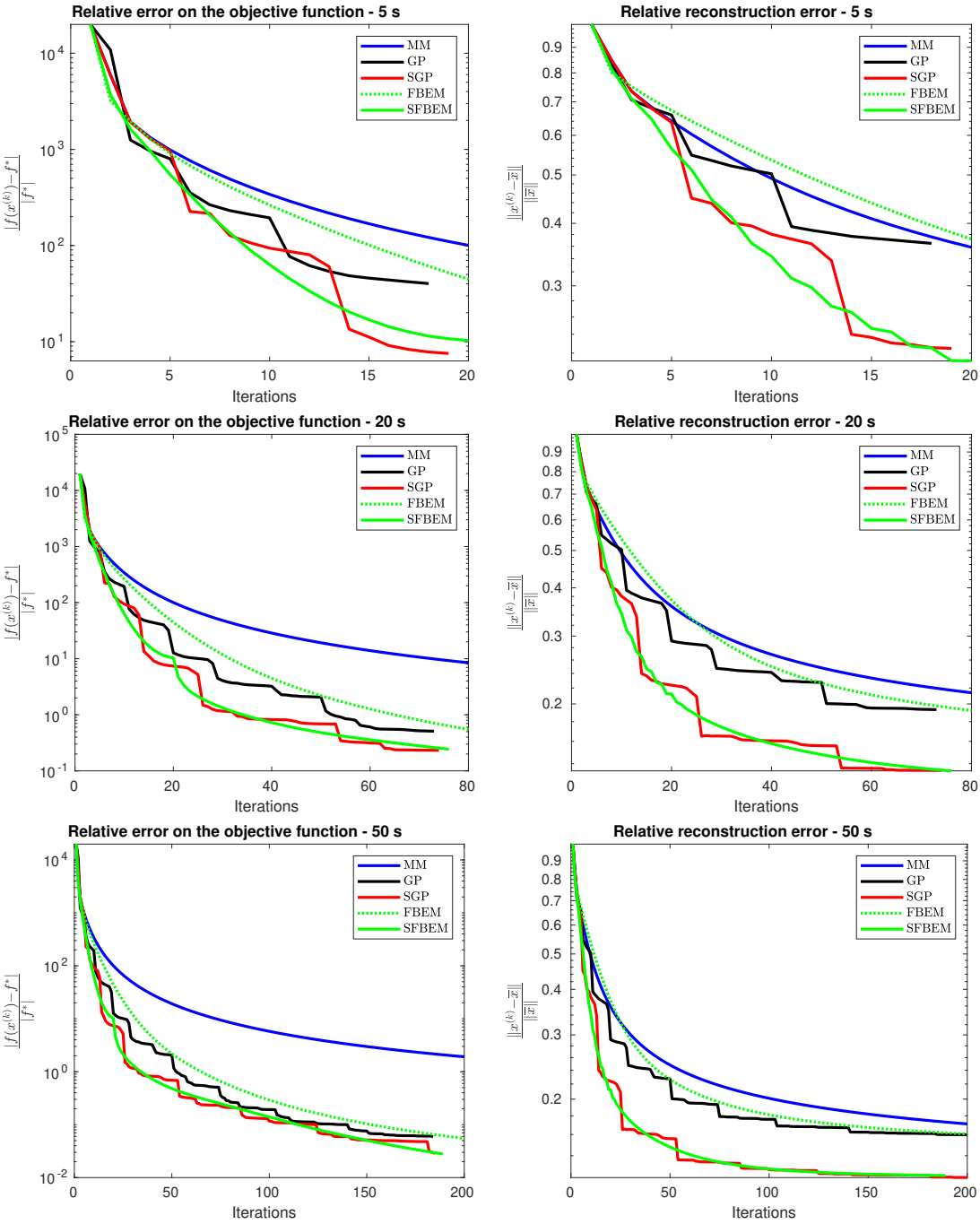

Figure 4: Relative error on the function values and relative reconstruction error generated by the considered methods for the dataset **G1** at three different temporal windows: 5 seconds (first row), 20 seconds (second row) and 50 seconds (third row).

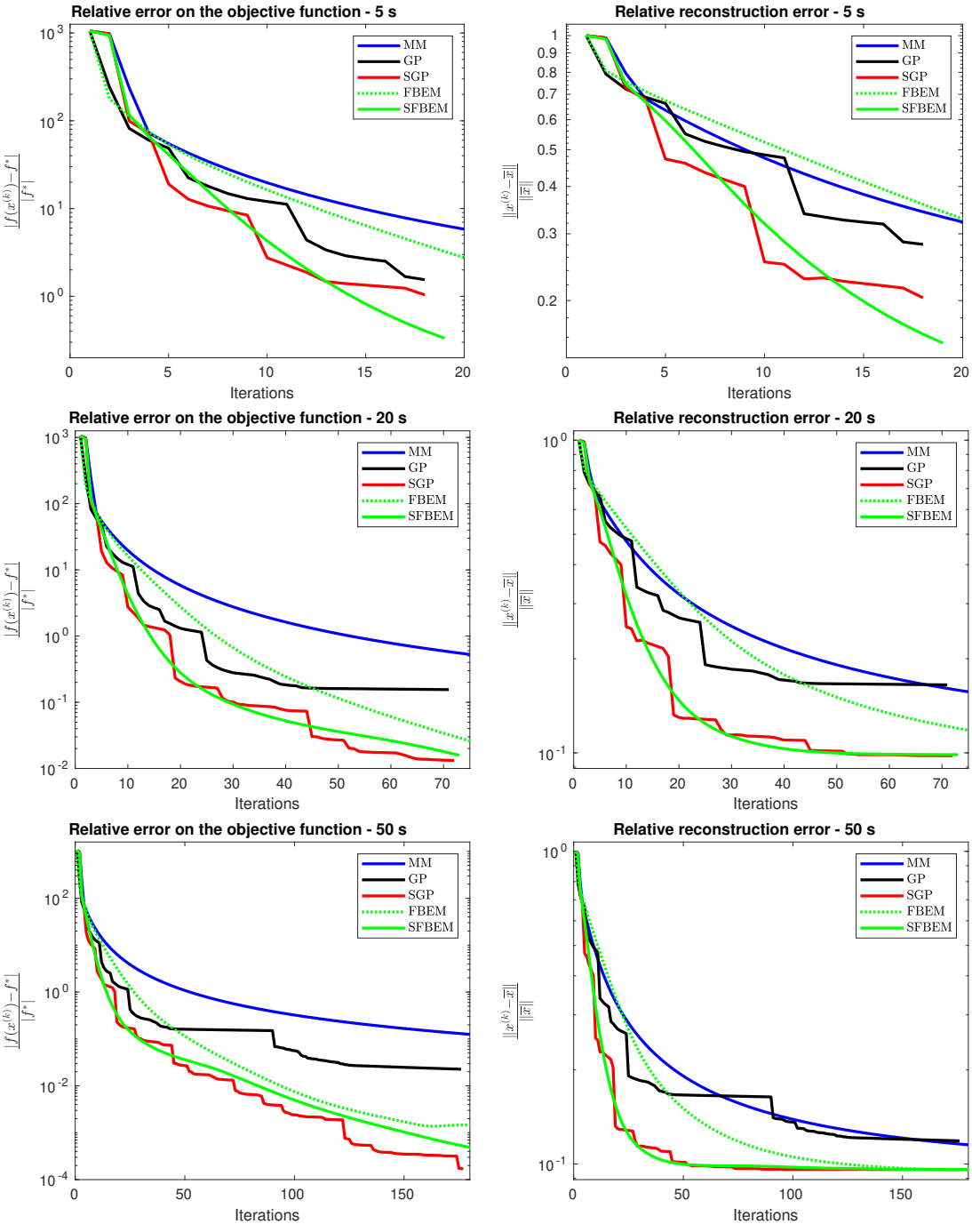

Figure 5: Relative error on the function values and relative reconstruction error generated by the considered methods for the dataset **G2** at three different temporal windows: 5 seconds (first row), 20 seconds (second row) and 50 seconds (third row).

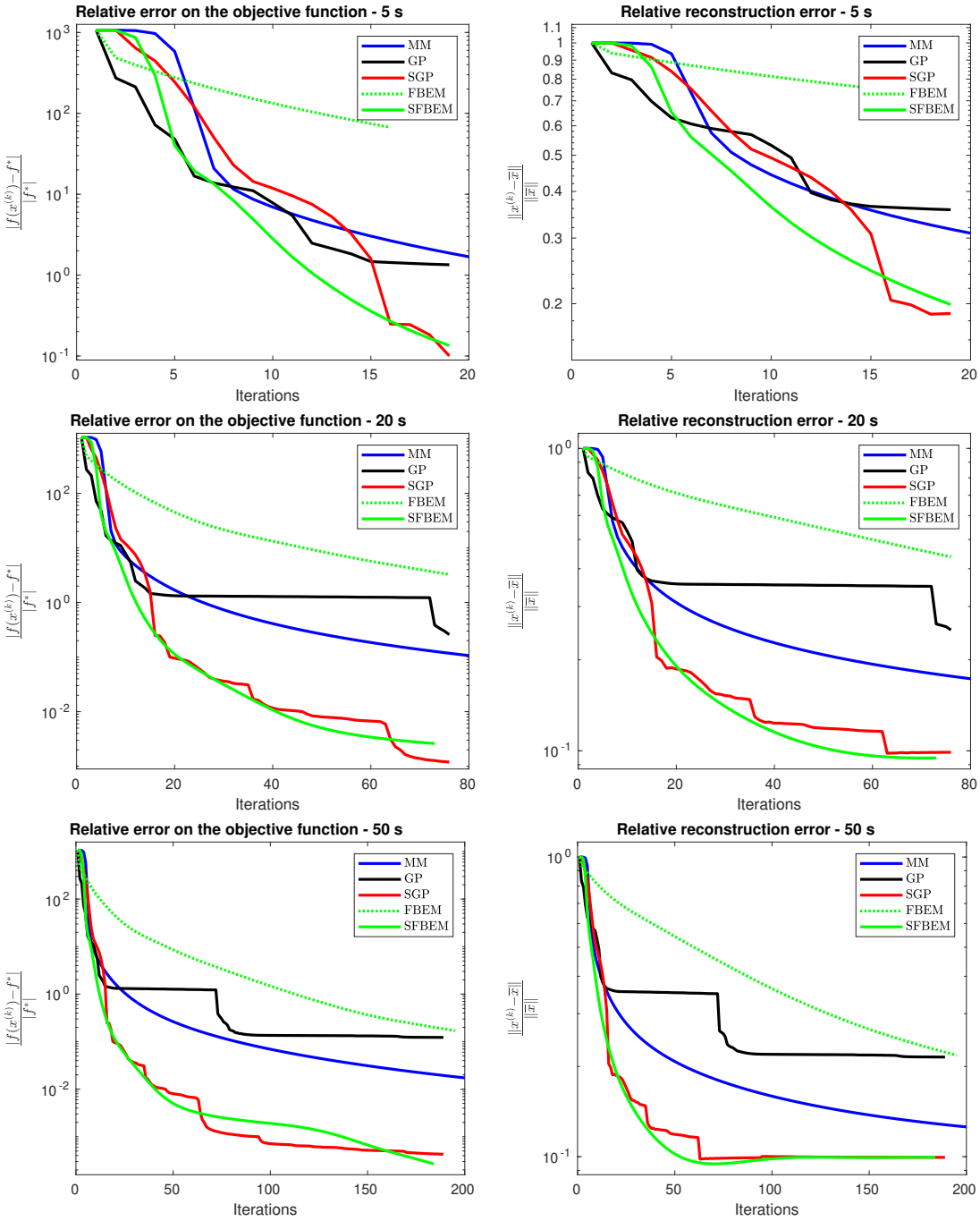

Figure 6: Relative error on the function values and relative reconstruction error generated by the considered methods for the dataset **P1** at three different temporal windows: 5 seconds (first row), 20 seconds (second row) and 50 seconds (third row).

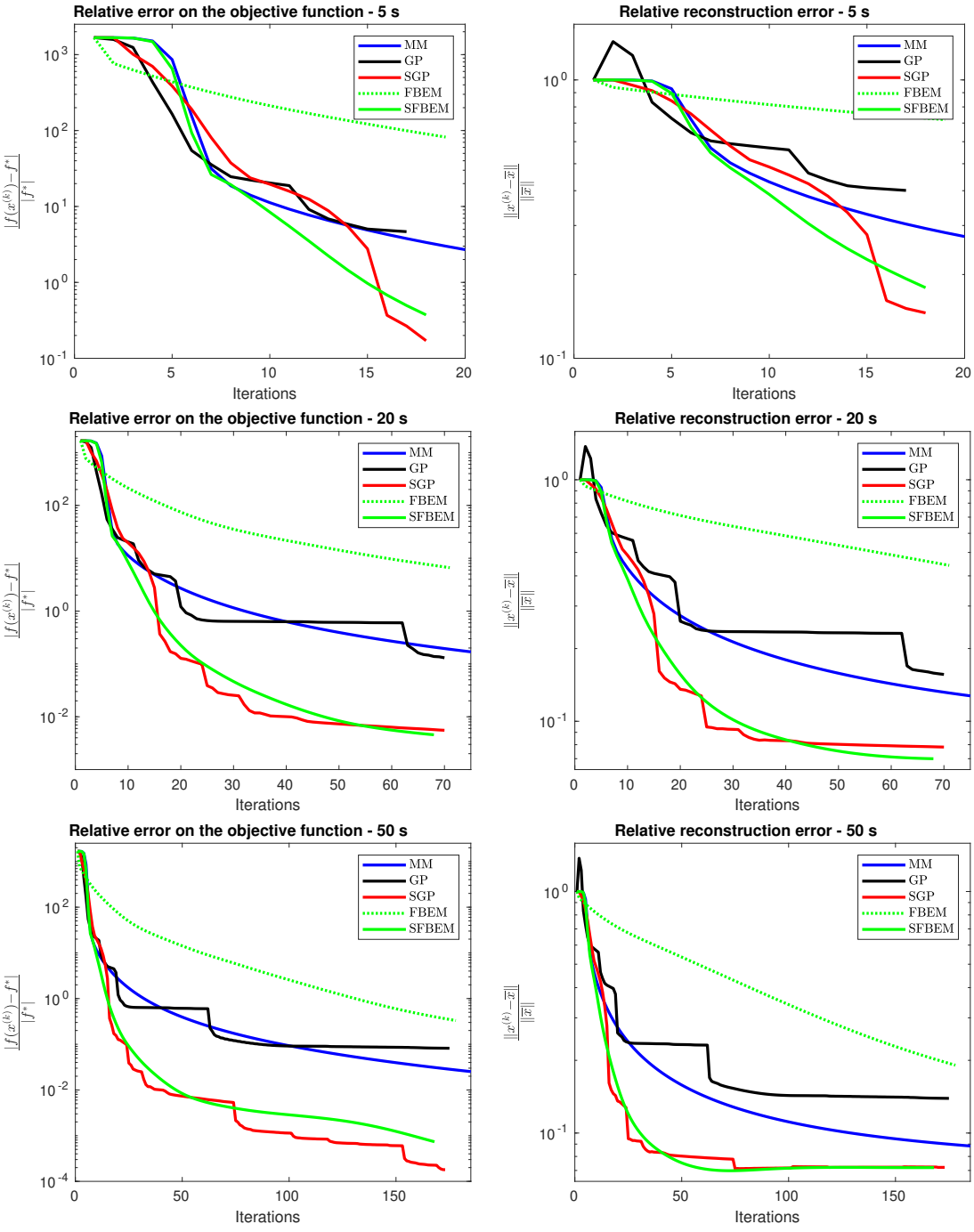

Figure 7: Relative error on the function values and relative reconstruction error generated by the considered methods for the dataset **P2** at three different temporal windows: 5 seconds (first row), 20 seconds (second row) and 50 seconds (third row).

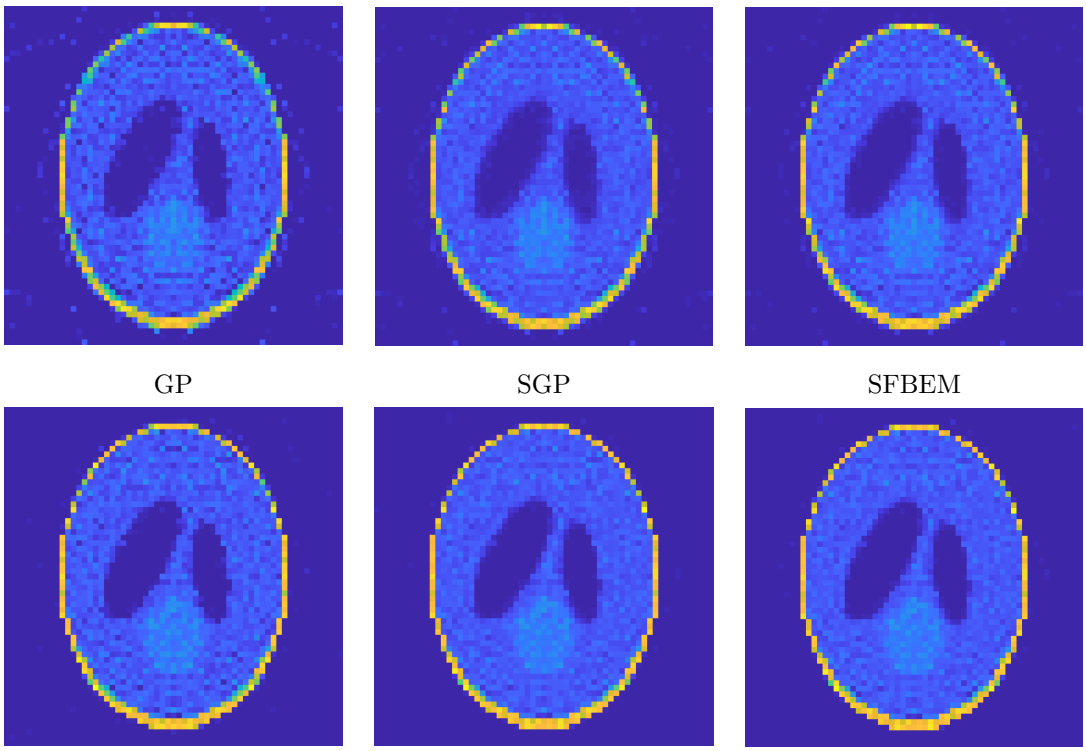

Figure 8: Reconstruction provided by GP, SGP and SFBEM after 5 seconds (first row) and after 20 seconds (second row) for the **G1** dataset.

Table 4: Computational results for the TV edge-preserving regularization problem in presence of signal dependent Gaussian noise.

	$F_k \leq 10^{-2}$				$F_k \leq 10^{-4}$				$F_k \leq 10^{-6}$			
method	k	time	E_k	e_k	k	time	E_k	e_k	k	time	E_k	e_k
VMILA_a	9	0.15	0.050	2.4e-02	58	0.68	0.048	3.1e-03	688	6.85	0.049	2.3e-05
VMILA_b	42	0.41	0.052	2.8e-02	273	2.70	0.048	6.0e-03	998	9.93	0.049	8.1e-04
ILA	44	0.33	0.053	2.9e-02	271	2.20	0.048	6.1e-03	925	7.60	0.049	9.2e-04
VMFB	203	3.65	0.052	2.8e-02	-	-	-	-	-	-	-	-

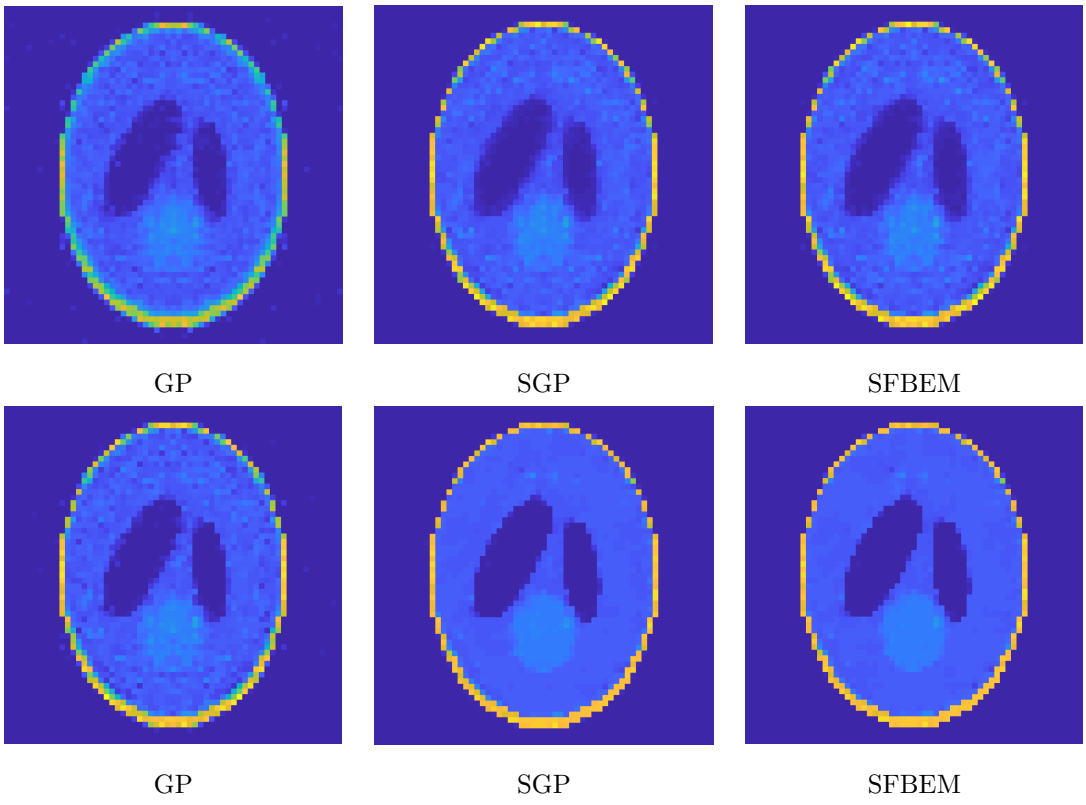

Figure 9: Reconstruction provided by GP, SGP and SFBEM after 5 seconds (first row) and after 20 seconds (second row) for the **P1** dataset.

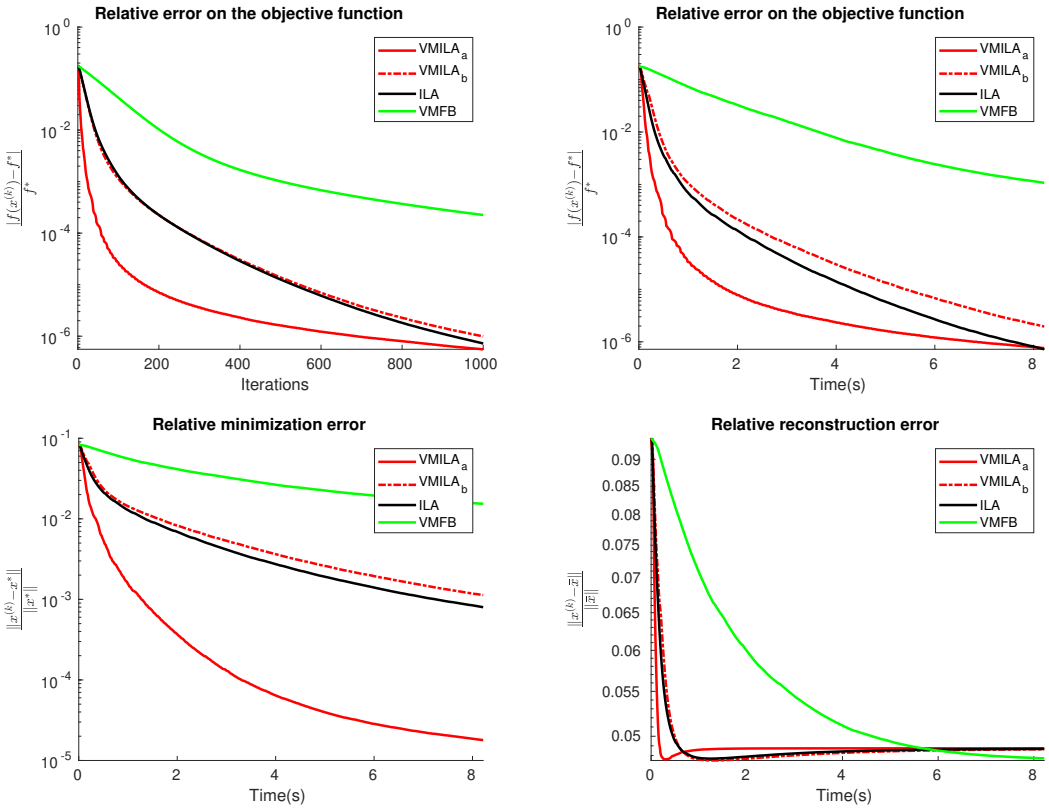

Figure 10: Behaviour of the relative error on the function values (first row), the minimization error and the reconstruction error for the jetplane test problem.