A Local MM Subspace Method for Solving Constrained Variational Problems in Image Recovery

Emilie Chouzenoux, Ségolène Martin, Jean-Christophe Pesquet

Noname manuscript No.
(will be inserted by the editor)

A Local MM Subspace Method for Solving Constrained Variational Problems in Image Recovery

Emilie Chouzenoux · Ségolène Martin · Jean-Christophe Pesquet

Received: date / Accepted: date

Abstract This article introduces a new Penalized Majorization-Minimization Subspace algorithm (P-MMS) for solving smooth, constrained optimization problems. In short, our approach consists of embedding a subspace algorithm in an inexact exterior penalty procedure. The subspace strategy, combined with a Majoration-Minimization step-size search, takes great advantage of the smoothness of the penalized cost function, while the penalty method allows to handle a wide range of constraints. The main drawback of exterior penalty approaches, namely ill-conditioning for large values of the penalty parameter, is overcome by using a trust-region-like technique. The convergence of the resulting algorithm is analyzed. Numerical experiments carried out on two large-scale image recovery applications demonstrate that, compared with state-of-the-art algorithms, the proposed method performs well in terms of computational time.

Keywords constrained optimization · smooth optimization · subspace acceleration · exterior penalty method · Majorization-Minimization · wavelet restoration · PET reconstruction

Emilie Chouzenoux acknowledges support from the European Research Council Starting Grant MAJORIS ERC-2019-STG-850925. Jean-Christophe Pesquet acknowledges support from the Teaching and Research Chair in Artificial Intelligence BRIDGEABLE.

Emilie Chouzenoux[1]
E-mail: emilie.chouzenoux@centralesupelec.fr

Ségolène Martin[1] (corresponding author)
E-mail: segolene.martin@centralesupelec.fr

Jean-Christophe Pesquet[1]
E-mail: jean-christophe.pesquet@centralesupelec.fr

[1]Université Paris-Saclay, Inria, CentraleSupélec, Centre de Vision Numerique, 9 rue Joliot Curie, Gif-sur-Yvette 91190, France

1 Introduction

Many challenges in image processing can be addressed by solving constrained optimization problems. These problems may be difficult to solve numerically in reasonable times because of their high dimension and of the involved constraints. These constraints may play diverse and crucial roles as they enforce prior knowledge about the solution. For example, constraints may have a regularization effect [30], promote sparsity [6], ensure consistency with the noise model through a data-fit term [2,52], or impose the fulfillment of geometrical properties as in deformable image matching [61].

In this paper, we propose a novel method for solving large-scale differentiable constrained problems formulated as

$$\underset{x\in C}{\text{minimize}} \;\; F(x), \tag{1}$$

where $F\colon \mathbb{R}^N \longrightarrow \mathbb{R}$ is a differentiable cost function and $C \subseteq \mathbb{R}^N$. In particular, our approach allows to tackle image recovery problems of high dimension (typically, at least 10^9 variables) efficiently. Although such problems are often formulated as the minimization of a nondifferentiable objective function – being the sum of a differentiable data-fidelity term and a nondifferentiable regularization term, e.g. ℓ_1 or total variation semi-norms – satisfying (and sometimes better) results can be obtained by adopting a smooth formulation, yielding problems expressed as (1). Let us mention in

particular two common methods to circumvent the non-smoothness possibly arising in the regularization term. The first one consists of using a smoothed approximation to the regularization function, which usually does not alter the recovery performance and can sometimes avoid undesirable effects such as staircasing [63]. The second possibility is to re-express the regularization term in the objective as a constraint, in the spirit of Ivanov approach [53].

The differentiability assumption gives us access to a wide range of algorithms for solving Problem (1), that we review hereafter. Let us first give a brief overview of standard methods for solving *unconstrained* differentiable optimization problems, i.e. Problem (1) with $C = \mathbb{R}^N$. These methods can be divided into two families, namely *line-search* and *trust region* methods (see [35] for a survey). Algorithms from both these classes are iterative; they generate sequences $(x_k)_{k\in\mathbb{N}}$ such that, for every $k \in \mathbb{N}$,

$$x_{k+1} = x_k + s_k, \tag{2}$$

where $s_k \in \mathbb{R}^N$. Line-search and trust region methods differ from the way they compute the *step* s_k at each iteration. On the one hand, in the line-search strategy, one chooses a direction $d_k \in \mathbb{R}^N$ and searches along this direction from the current iterate x_k for a new iterate with a (sufficiently) lower function value. The scale factor $\alpha_k \in \mathbb{R}$ quantifying the move along d_k, referred to as the *step-size*, is found by approximately solving the following scalar minimization problem:

$$\underset{\alpha\in\mathbb{R}}{\text{minimize}} \;\; F(x_k + \alpha d_k). \tag{3}$$

The resulting step is then defined as $s_k = \alpha_k d_k$. On the other hand, in the trust region strategy, the step s_k is computed by solving an inner optimization problem on the whole space $\mathbb{R}^N$ as follows. First, a quadratic model m_k of F around x_k is constructed from the information available at the current k-th iterate. This model is expected to provide a satisfactory approximation to the objective in a neighborhood of x_k, so that the step s_k is defined as an inexact solution to the *trust region subproblem*:

$$\begin{aligned} &\underset{s\in\mathbb{R}^N}{\text{minimize}} && m_k(s) \\ &\text{subject to} && \|s\|^2 \leq \Delta_k, \end{aligned} \tag{4}$$

where the radius $\Delta_k > 0$ of the trust region follows predefined rules. However, the aforementioned line-search and trust region strategies might not be suitable for solving large-scale problems. For the former, the direction d_k might be difficult to compute (e.g., Newton direction) or the decrease of the objective function too slow along iterations (e.g., gradient direction). For the latter, solving the full-space constrained optimization problem (4) at each iteration might be computationally demanding.

A successful strategy for overcoming the curse of dimensionality amounts to search for a compromise between a one-dimensional linesearch and an inner problem resolution in the entire space. The idea is to select s_k within a restricted low-dimensional subspace $\mathcal{S}_k$ of $\mathbb{R}^N$, so that, for every $k \in \mathbb{N}$,

$$s_k \in \mathcal{S}_k, \tag{5}$$

yielding the so-called *subspace methods*. It is worth mentioning that the state-of-the-art line-search methods can be viewed as a special type of such methods, as they actually define the direction d_k as a linear combination of other directions of interest. For example, the well-known nonlinear conjugate gradient (NLCG) [46] and limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) [56] algorithms can be interpreted as subspace methods, with $\mathcal{S}_k$ respectively defined as

$$\mathcal{S}_k = \text{Span}\{g_k, x_k - x_{k-1}\}, \tag{6}$$

and

$$\begin{aligned} \mathcal{S}_k = \text{Span}\{-g_k, x_k - x_{k-1}, \ldots, x_{k-m+1} - x_{k-m}, \\ g_k - g_{k-1}, \ldots, g_{k-m+1} - g_{k-m}\}, \end{aligned} \tag{7}$$

for some $m \in \mathbb{N}^*$, where $g_k = \nabla F(x_k)$. In this work, we will be interested in more general subspace methods where the whole subspace $\mathcal{S}_k$ is explored [27,42], i.e. for which a multidimensional step-size is computed instead of the scalar step-size α_k. Specifically, one defines the step-size $u_k \in \mathbb{R}^{M_k}$, where M_k is the dimension of $\mathcal{S}_k$, as an approximate solution to

$$\underset{u\in\mathbb{R}^{M_k}}{\text{minimize}} \;\; F(x_k + D_k u), \tag{8}$$

with $D_k \in \mathbb{R}^{N\times M_k}$ the matrix of directions which span $\mathcal{S}_k$, and then update x_k by

$$x_{k+1} = x_k + D_k u_k. \tag{9}$$

A practical and efficient way to compute the step u_k, which we will consider later, is the Majorization-Minimization procedure proposed in [26]. In particular, approaches of the form (9) have shown their interest in terms of computation time on several image processing applications [47,87]. The subspace strategy can also be applied to trust region methods, in such case the trust region subproblem (4) is modified as follows:

$$\begin{aligned} &\underset{s\in\mathbb{R}^N}{\text{minimize}} && m_k(s) \\ &\text{subject to} && \|s\|^2 \leq \Delta_k \text{ and } s \in \mathcal{S}_k, \end{aligned} \tag{10}$$

therefore reducing the dimension of the inner problem from N to M_k. This strategy was successfully used in many problems, including image processing problems [73,80,81,82].

Interestingly, subspace methods can also be used to tackle *constrained* differentiable optimization problems, corresponding to (1) with $C \subsetneq \mathbb{R}^N$. In particular, the case of linear constraints can be handled efficiently by relying on specific techniques, similar to the ones used in full-space trust region methods [60] or unidirectional line-search methods [29]. For instance, active-set methods [65, Chap. 16] have been used for line-search strategies, defining x_{k+1} via a direction computed from a reduced problem associated only with the inactive variables. Based on these ideas, Shanno and Marsten [72] proposed a conjugate gradient method for solving problems with linear equality constraints and bound constraints. Byrd et al. [19] adapted the L-BFGS method [56] to minimize a smooth function under bound constraints leading to the extensively used L-BFGS-B algorithm. Furthermore, rescaling strategies [35, Sec. 6.7] have been applied to subspace trust region methods for bound-constrained optimization. For instance, in [16], Branch et al. performed a change of variables to eliminate the bound constraint in the trust region subproblem and their algorithm was applied to image deblurring in [79].

In contrast, there are fewer works on how to use subspace algorithms for handling nonlinear constraints. In the literature, attempts to use line-search or trust region methods (without a subspace strategy) for problems with general constraints of the form

$$C = \{x \in \mathbb{R}^N \mid c_i(x) \leq 0,\ \forall i \in \{1, \dots, r\}\}, \tag{11}$$

where, for every $i \in \{1, \dots, r\}$, $c_i : \mathbb{R}^N \longrightarrow \mathbb{R}$ is continuously differentiable, usually involve the setup of sophisticated procedures. These procedures are essentially based on barrier procedure [17], exterior penalty [34] strategies, Sequential Quadratic Programming (SQP) approaches [50], or mixed penalty-SQP strategies [45, 37]. To our knowledge, mostly SQP methods have been combined with a subspace strategy. SQP consists of solving a sequence of quadratic programming subproblems. Each of them aims at minimizing a quadratic model of the objective subject to a linearization of the constraints. A typical subspace SQP line-search [65, Sec. 18.4] approach solves the subproblem:

$$\begin{aligned} \underset{d \in \mathbb{R}^N}{\text{minimize}} \quad & m_k(d) \\ \text{subject to} \quad & \nabla c_i(x_k)^\top d + c_i(x_k) \leq 0,\ i \in \{1, \dots, r\}, \\ & d \in \mathcal{S}_k, \end{aligned} \tag{12}$$

where m_k is a quadratic model of F at the current iterate x_k. The next iterate is then computed as $x_{k+1} = x_k + \alpha_k d_k$, where d_k is a solution to (12) and $\alpha_k \in \mathbb{R}$ is determined via classical rules. Lee et al. [55] analyzed this subspace SQP line-search algorithm for the particular choice of subspace:

$$\mathcal{S}_k = \text{Span}\{-g_k, d_1, \dots, d_{k-1}, -\nabla c_i(x_k)\}, \tag{13}$$

with $i = \text{argmax}\{|c_i(x_k)|,\ i \in \{1, \dots r\}\}$. On the other hand, in a subspace SQP trust region approach [65, Sec. 18.5], the solution s_k to the following subproblem is computed:

$$\begin{aligned} \underset{s \in \mathbb{R}^N}{\text{minimize}} \quad & m_k(s) \\ \text{subject to} \quad & \nabla c_i(x_k)^\top s + c_i(x_k) \leq 0,\ i \in \{1, \dots, r\}, \\ & \|s\| \leq \Delta_k, \\ & s \in \mathcal{S}_k, \end{aligned} \tag{14}$$

and the next iterate corresponds to $x_{k+1} = x_k + s_k$. However the trust region strategy (14) is more complex to implement than the line-search one (12) because the approximate feasible region and the trust region may be disjoint. Celis et al. [22] proposed to relax problem (14), in the case $\mathcal{S}_k = \mathbb{R}^N$, into the so-called CDT problem, by replacing the linear inequalities with

$$\| \left(\nabla c_i(x_k)^\top s + c_i(x_k)\right)_- \|^2 \leq \zeta_{k,i},\ i \in \{1, \dots, r\}, \tag{15}$$

where $(\zeta_{k,i})_{1 \leq i \leq r} \in (\mathbb{R}^+)^r$. Grapiglia et al. [51] studied a subspace version of CDT, but one drawback of their method is that the subspace dimension increases with iterations. In general, although SQP methods converge locally at a superlinear rate, they remain complex to implement as the subproblems may be infeasible. Well-known undesirable practical convergence behaviors [66] must also be circumvented. In addition, even in the special case when the constraint set C is closed convex, these approaches do not take advantage of the projection onto C when it is available. To our knowledge, the only subspace methods to do so are projected NLCG [74,85] in the same spirit as projected gradient algorithms [13]. However, such algorithms remain limited to this category of constraints and cannot cope with simple precomposition with linear operators.

The method we investigate in this paper is an adaptation of the Majorization-Minimization Subspace algorithm (MMS) proposed in [27] to constrained problems. In a nutshell, our method is a subspace method following the scheme (9), coupled with techniques reminiscent of trust region techniques, embedded into an exterior penalty framework. Our approach offers several advantages over the previously mentioned ones: (i) it can take

into account a wide variety of constraints, and in particular, constraints for which the projection has closed-form are easily handled; (ii) the choice of the subspace is very flexible; (iii) the step search at each iteration is easy to implement, in particular it does not require solving a constrained problem thus avoiding feasibility issues. Our contribution is twofold. We first propose a modification of the MMS algorithm for solving efficiently penalized problems. The new MMS algorithm uses a local majoration strategy similar to trust region approaches, to overcome the ill-conditioning that usually arises in penalty methods. The convergence of the algorithm is analyzed. Secondly, the penalty method in which the new MMS algorithm is nested is allowed to be inexact, each subproblem being solved with a fixed tolerance. The convergence of the global method towards a solution to the constrained problem (1) is proved.

The rest of the paper is organized as follows: Section 2 presents the optimization problem tackled through the paper. In Section 3, we describe the original MMS algorithm, and we propose a new accelerated variant of it in Section 4. Section 5 is dedicated to the inexact exterior penalty method. Finally, in Section 6, comparisons between our approach and state-of-the-art algorithms are conducted on two image recovery applications.

2 Considered optimization problem

2.1 Notation and definitions

In this paper, $\mathbb{R}^N$ denotes the N-dimensional Euclidean space endowed with the standard scalar product $\langle \cdot, \cdot \rangle$ and the norm $\| \cdot \|$. We denote by $\mathcal{S}^N$ the set of symmetric matrices of $\mathbb{R}^{N \times N}$ and $\mathcal{S}^N_+$ the set of symmetric semi-definite positive matrices of $\mathbb{R}^{N \times N}$. The Loewner order relation is denoted by $\preceq$, i.e., for every $A \in \mathcal{S}^N$ and $B \in \mathcal{S}^N$, $A \preceq B$ if, for every $x \in \mathbb{R}^N$, $x^\top A x \leq x^\top B x$. We refer to the identity matrix of $\mathbb{R}^{N \times N}$ as I_N. For every nonempty set $C \subseteq \mathbb{R}^N$, d_C is the distance function to C with respect to the Euclidean norm, and ι_C is the indicator function of C, namely $\iota_C(x) = 0$ if $x \in C$, $\iota_C(x) = +\infty$ otherwise.

The following definition will be at the core of the developments in this paper:

Definition 1 (Exterior penalty function) Let C be a nonempty subset of $\mathbb{R}^N$. We say $R \colon \mathbb{R}^N \longrightarrow [0, +\infty)$ is an *exterior penalty function for constraint* $x \in C$ if

$$\operatorname{Argmin} R = C \quad \text{and} \quad \min R = 0. \tag{16}$$

As their name suggests it, such penalty functions assign a nonnegative cost to every point which is exterior to C.

Example 1 Let $\psi \colon [0, +\infty) \to [0, +\infty)$ be a strictly increasing function such that $\psi(0) = 0$. Let the constraint set C be the lower zero-level set of a function $f \colon \mathbb{R}^N \longrightarrow \mathbb{R}$. Then functions of the form

$$(\forall x \in \mathbb{R}^N) \quad R(x) = \psi\big(\max(0, f(x))\big), \tag{17}$$

are exterior penalty functions for the constraint $f(x) \leq 0$ [10,57]. In addition,

- if f and ψ are convex, then R is convex;
- if f and ψ are differentiable and $\psi'(0) = 0$, then R is differentiable.

In particular, when $\psi = (\cdot)^2$, we recover the widely used quadratic penalty [44,21].

Remark 1 One could think of using exact penalization methods. However, these techniques require more restrictive assumption since either they use non-differentiable penalties, or the objective and the constraint functions of the problem are assumed to be twice differentiable [38,10].

A useful property of exterior penalty functions is the following one:

Proposition 1 *Let $r \in \mathbb{N}^*$ and let $(C_i)_{1 \leq i \leq r}$ be subsets of $\mathbb{R}^N$. For every $i \in \{1, \ldots, r\}$, let $R_i \colon \mathbb{R}^N \longrightarrow \mathbb{R}$ be an exterior penalty function for constraint $x \in C_i$. Then $\sum_{i=1}^r R_i$ is an exterior penalty function for the constraint $x \in \bigcap_{i=1}^r C_i$.*

2.2 Optimization problem

Throughout this paper, we will use functions F and R satisfying the following hypotheses.

Assumption 1.

(i) *$F \colon \mathbb{R}^N \longrightarrow \mathbb{R}$ is a differentiable function.*

(ii) *C is a nonempty subset of $\mathbb{R}^N$, which is decomposed as $C = \bigcap_{i=1}^r C_i$, where $r \in \mathbb{N}^*$ and $(C_i)_{1 \leq i \leq r}$ are subsets of $\mathbb{R}^N$.*

(iii) *$R = \sum_{i=1}^r R_i$ where, for every $i \in \{1, \ldots, r\}$, $R_i \colon \mathbb{R}^N \longrightarrow \mathbb{R}$ is a differentiable exterior penalty function for constraint $x \in C_i$.*

Our main goal will be to solve the following constrained optimization problem.

Problem $(\mathcal{P})$ Suppose that Assumption 1 is satisfied. We want to find

$$\widehat{x} \in \operatorname*{Argmin}_{x \in C} F(x). \tag{18}$$

Following the framework of exterior penalty methods, we will tackle this problem by considering the penalized problem hereunder defined for every $\gamma \in (0, +\infty)$.

Problem $(\mathcal{P}_\gamma)$ Suppose that Assumption 1 is satisfied. We want to find

$$\widehat{x}_\gamma \in \operatorname*{Argmin}_{x\in\mathbb{R}^N} F_\gamma(x), \tag{19}$$

where

$$(\forall x \in \mathbb{R}^N) \quad F_\gamma(x) = F(x) + \gamma R(x). \tag{20}$$

3 Majorization-Minimization subspace algorithm

Let the penalty parameter $\gamma > 0$ be fixed in the whole section, and let F and $(R_i)_{1\leq i\leq N}$ be functions fulfilling Assumption 1. We seek to solve the penalized problem $(\mathcal{P}_\gamma)$. Among available methods, the Majorization-Minimization (MM) principle offers a generic framework to solve this problem and can be efficiently combined with a subspace acceleration [78,42,83]. Hereafter, we describe the use of the Majorization-Minimization subspace algorithm, initially proposed in [26], for this particular problem.

3.1 Majoration-Minimization principle

Definition 2 Let $f : \mathbb{R}^N \longrightarrow \mathbb{R}$ be a differentiable function and $x' \in \mathbb{R}^N$. A function $x \mapsto Q(x, x')$ is said to be a *global tangent majorant* of f at x' if, for every $x \in \mathbb{R}^N$,

$$f(x) \leq Q(x, x') \quad \text{and} \quad f(x') = Q(x', x'). \tag{21}$$

Moreover $x \mapsto Q(x, x')$ is said to be a global tangent *quadratic* majorant of f at x' if

$$\begin{aligned}(\forall x \in \mathbb{R}^N) \quad Q(x, x') = f(x') + \nabla f(x')^\top (x - x') \\ + \frac{1}{2}(x - x')^\top A(x')(x - x'),\end{aligned} \tag{22}$$

with $A(x') \in \mathcal{S}^N_+$. In that case, $A(x')$ is called the *curvature matrix* of function Q at point x'.

The following proposition, which is readily proved, provides a useful tool for building quadratic majorants of functions involving a precomposition with a linear operator.

Proposition 2 *Let $\psi\colon \mathbb{R}^P \longrightarrow \mathbb{R}$ be a differentiable function, let $L \in \mathbb{R}^{P\times N}$ and let $c \in \mathbb{R}^P$. Assume that for every $y' \in \mathbb{R}^P$, there exists $A(y') \in \mathcal{S}^P_+$ such that*

$$\begin{aligned}(\forall y \in \mathbb{R}^P) \quad \psi(y) \leq \psi(y') + \nabla\psi(y')^\top (y - y') \\ + \frac{1}{2}(y - y')^\top A(y')(y - y').\end{aligned} \tag{23}$$

Then function Ψ defined as

$$(\forall x \in \mathbb{R}^N) \quad \Psi(x) = \psi(Lx + c) \tag{24}$$

satisfies the following majoration property at every $x' \in \mathbb{R}^N$:

$$\begin{aligned}(\forall x \in \mathbb{R}^N) \quad \Psi(x) \leq \Psi(x') + \nabla\Psi(x')^\top (x - x') \\ + \frac{1}{2}(x - x')^\top B(x')(x - x')\end{aligned} \tag{25}$$

with

$$B(x') = L^\top A(Lx' + c)L \in \mathbb{R}^{N\times N}. \tag{26}$$

In the following, we will assume that the functions F and $(R_i)_{1\leq i\leq r}$, involved in Problem $(\mathcal{P}_\gamma)$, have quadratic majorants:

Assumption 2.
(i) For every $x \in \mathbb{R}^N$, F has a quadratic tangent majorant at x with curvature matrix $A_F(x)$.
(ii) For every $i \in \{1, \ldots, r\}$ and for every $x \in \mathbb{R}^N$, R_i has a quadratic tangent majorant at x with curvature matrix $A_{R_i}(x)$.

As a consequence, for every $x \in \mathbb{R}^N$, the penalty function R has a quadratic tangent majorant at x with curvature

$$A_R(x) = \sum_{i=1}^{r} A_{R_i}(x), \tag{27}$$

and F_γ defined in (20) has a quadratic tangent majorant at x with curvature

$$A_{F_\gamma}(x) = A_F(x) + \gamma A_R(x). \tag{28}$$

Example 2 Let us consider a wide class of penalty functions that we will subsequently use in our numerical experiments (see Section 6). Let $L \in \mathbb{R}^{P\times N}$ and let C' be a nonempty closed convex set of $\mathbb{R}^P$. Suppose that the constraint set is

$$C = L^{-1}(C') = \{x \in \mathbb{R}^N \mid Lx \in C'\}. \tag{29}$$

Let us define the exterior penalty function relative to the constraint set (29) as

$$(\forall x \in \mathbb{R}^N) \quad R(x) = \mathrm{d}^2_{C'}(Lx). \tag{30}$$

Since $\psi = \mathrm{d}^2_{C'}$ has a 2-Lipschitz gradient, (23) holds with, for every $y' \in \mathbb{R}^P$, $A(y') = 2\mathrm{I}_P$. It then follows from Proposition 2 that a quadratic majorant of function R can be built with (constant) curvature

$$(\forall x \in \mathbb{R}^N) \quad A_R(x) = 2L^\top L. \tag{31}$$

MM approaches are iterative methods for the minimization of an objective function $f\colon \mathbb{R}^N \longrightarrow \mathbb{R}$ based on a surrogate such as (22). At each iteration of the algorithm, a global tangent majorant of the objective at the current iterate is minimized, yielding the following update rule:

$$\begin{cases} x_0 \in \mathbb{R}^N, \\ (\forall k \in \mathbb{N}) \quad x_{k+1} \in \underset{x\in\mathbb{R}^N}{\operatorname{Argmin}}\ Q(x, x_k). \end{cases} \tag{32}$$

When applied to the objective F_γ, the iterative procedure (32) corresponds to the half-quadratic algorithm. The convergence guarantees for this algorithm have been established for instance in [64,4] under mild boundedness assumptions on the curvature matrices sequence $\big(A_{F_\gamma}(x_k)\big)_k$. However, the computational complexity for (32) is rather high, as it involves the inversion of an $N \times N$ matrix at each iteration. We thus propose to resort to a subspace accelerated version of it that we describe next.

3.2 Majorization-Minimization subspace algorithm

The MM subspace algorithm (MMS) takes advantage of both the MM principle, introduced in Sec. 3.1 and subspace acceleration strategies. In short, the MMS method that we describe in this section is a subspace minimization algorithm coupled with a quadratic MM step-size search [26,27]. This allows us to avoid tedious $N \times N$ matrix inversions involved in the update of (32), while still preserving fast practical convergence [28].

3.2.1 Subspace minimization

A general subspace method moves the current point x_k at each iteration $k \in \mathbb{N}$ along a subspace of dimension $M_k \in \{1, \ldots, N\}$, by generating

$$(\forall k \in \mathbb{N}) \quad x_{k+1} = x_k + D_k u_k, \tag{33}$$

where

$$(\forall k \in \mathbb{N}) \quad D_k = [d_k^1, \ldots, d_k^{M_k}] \tag{34}$$

and, for every $j \in \{1, \ldots, M_k\}$, $d_k^j \in \mathbb{R}^N$. D_k is the search direction matrix and $u_k \in \mathbb{R}^{M_k}$ is a multivariate step-size. In the present study, we will adopt the same requirement on D_k as in [26], that is d_k^1 is assumed to be *gradient-related* (see more details in Sec. 4.2). The following example describes relevant choices for D_k encompassed by our analysis.

Example 3 Let $k \in \mathbb{N}$ and let us introduce the shorter notation $g_k = \nabla F_\gamma(x_k)$.

(i) If $d_k^1 = -g_k$, then we retrieve the classical gradient descent direction.

(ii) We can also use a preconditioned gradient direction, $d_k^1 = -H_k g_k$, where H_k is a symmetric positive definite matrix of $\mathbb{R}^{N\times N}$. For instance, if the Hessian of the objective function is invertible at the current iterate, H_k can be chosen equal to the inverse of this Hessian (or an approximation of it). We then obtain the classical Newton (or quasi-Newton) direction.

(iii) One can also use for d_k^1, the truncated Newton direction obtained by solving approximately the linear system $\nabla^2 F_\gamma(x_k)d + g_k = 0$ with the Conjugate Gradient algorithm. Such choice was adopted, for instance, in the SESOP-TN method from [86].

(iv) When the subspace is spanned by the gradient direction and the difference between two past iterates (also called *memory term*), i.e.

$$D_k = \begin{cases} -g_k & k = 0 \\ [-g_k, x_k - x_{k-1}] & \text{elsewhere,} \end{cases} \tag{35}$$

we recover the *memory gradient* subspace introduced in [59], which is closely related to the search direction employed in the NLGC algorithm [46,86].

(v) A generalization of the latter consists in adding further past directions by setting for some $m \in \mathbb{N}^*$, and $k \geq m$:

$$D_k = [-g_k, x_k - x_{k-1}, \ldots, x_{k-m+1} - x_{k-m}], \tag{36}$$

yielding the *supermemory gradient subspace* from [36].

(vi) Matrix

$$D_k = [-g_k, x_k - x_{k-1}, \ldots, x_{k-m+1} - x_{k-m}, \\ g_k - g_{k-1}, \ldots, g_{k-m+1} - g_{k-m}], \tag{37}$$

with $m \in \mathbb{N}^*$, and $k \geq m$, spans the subspace inherent to the limited memory quasi-Newton L-BFGS algorithm [56], when setting the memory size to m. L-BFGS method amounts to approximate the Newton step by a step in the above subspace, composed of $(2m + 1)$ directions including the m last differences between the passed iterates and the m last differences between the past gradients.

Once the method for building the subspace D_k has been selected, one must define a suitable step-size strategy. Two approaches are typically employed. In the first approach, used in NLCG and L-BFGS for instance, a vector $s_k \in \mathbb{R}^N$ belonging to the subspace spanned by the columns of D_k is selected in a deterministic way (e.g., conjugation rules in NLCG). Then, the next iterate is $x_{k+1} = x_k + \alpha_k s_k$ where $\alpha_k > 0$ is obtained through a standard one-dimensional linesearch procedure using rules such as Armijo, Goldstein, or Wolfe

ones [65]. This approach is appealing because of its low complexity cost, but it requires an ad-hoc procedure for determining s_k. Moreover, a one-dimensional step-size search might not allow to exploit optimally the whole subspace generated by D_k, which can limit the performance of the method. In contrast, a multidimensional step-size search consists of finding a step $u_k \in \mathbb{R}^{M_k}$ minimizing f_k defined by

$$(\forall u \in \mathbb{R}^M) \quad f_k(u) = F_\gamma(x_k + D_k u). \tag{38}$$

Solving the aforementioned minimization problem can be time-consuming, so a compromise must be reached between accuracy, convergence stability, and complexity. Typical strategies in the literature of subspace optimization involve a Newton or truncated Newton method [62,86], however they go with limited convergence guarantees.

As shown in [26], the MM framework allows to design a practical, fast step search, while providing better theoretical properties for convergence than Newton-based methods. The use of MM procedure for the step-size search in (33) yields MMS method, that we describe hereafter.

3.2.2 MMS algorithm

MMS defines the stepsize u_k in (33) by minimizing a quadratic tangent majorant for f_k, defined hereabove, at $u' = 0$. According to Assumption 2, such a majorant reads for every $k \in \mathbb{N}$,

$$(\forall u \in \mathbb{R}^{M_k}) \quad q_k(u) = F_\gamma(x_k) + \nabla F_\gamma(x_k)^\top D_k u + \frac{1}{2} u^\top B_k u, \tag{39}$$

where $B_k \in \mathbb{R}^{M_k \times M_k}$ is defined as

$$B_k = D_k^\top A_{F_\gamma}(x_k) D_k, \tag{40}$$

and $A_{F_\gamma}(x_k)$ is the majorant curvature matrix introduced in (28). The MM step search strategy then reduces to minimizing the quadratic function q_k over $\mathbb{R}^{M_k}$, therefore leading to a low-complexity closed form computation for the step-size u_k at each iteration. The MMS method for minimizing F_γ is detailed in Algorithm 1, where † stands for the pseudo-inverse operation. The convergence properties of MMS are studied for instance in [28,26,27]. Its good practical performance, in comparison with NLCG and L-BFGS in particular, have been illustrated on various image restoration problems in [26,27].

Algorithm 1: MMS$(x_0, \gamma, \varepsilon)$

Inputs: $(\gamma, \varepsilon) \in (0, +\infty)^2$, $x_0 \in \mathbb{R}^N$.
for $k = 0, 1, \ldots...$ **do**
  Construct D_k according to (34),
  Compute B_k according to (40),
  $u_k = -[B_k]^\dagger D_k^\top \nabla F_\gamma(x_k)$,
  $x_{k+1} = x_k + D_k u_k$,
  if $\|\nabla F_\gamma(x_{k+1})\| \leq \varepsilon$ **then**
    exit loop // stop algorithm if given precision ε on the norm of the gradient is reached
    return x_{k+1}.
  end
end

4 Proposed local variant of MMS

The penalty function present in Problem $(\mathcal{P}_\gamma)$ may have a large curvature, which can jeopardize the good accuracy of the majorant function inherent to the MM strategy and thus significantly slowdown the convergence of the algorithm [71,25]. This issue had been overcome by the authors of [71], in the particular case of the MM memory gradient (3MG) method, by empirically substituting local majorations for global majorations, in a similar spirit as trust-region approaches [84,76,16]. In what follows, we formalize and generalize such a local approach to accelerate any MM subspace algorithm, and we demonstrate the convergence of the resulting iterative approach to a critical point of Problem $(\mathcal{P}_\gamma)$.

4.1 Description of the algorithm

According to theoretical results concerning penalty methods, large values of the penalty parameter γ should be chosen for the solutions to Problem $(\mathcal{P}_\gamma)$ to lie close to the constraint set. However, as stressed out in [35, 41,54], large values of γ tend to overemphasize the role of the penalty function R with respect to the objective function F. Consequently, the penalized problem becomes ill-conditioned, hence difficult to solve. This harshly impacts the convergence profile of MMS as observed in [71]. This slowdown is directly related to a large spectral norm of the curvature matrix B_k, defined by (40) at each iteration k, which leads to small update steps.

We thus propose a novel approach to cope with the aforementioned issue by making use of a local majorant

for F_γ — by opposition to a global one. We define a local majorant as follows.

Definition 3 Let $f : \mathbb{R}^N \longrightarrow \mathbb{R}$ be a differentiable function, let $\mathcal{D} \subseteq \mathbb{R}^N$, and let $x' \in \mathcal{D}$. A function $x \mapsto Q(x, x')$ is said to be a *local tangent majorant* of f at x' on the *trust region* $\mathcal{D}$ if, for every $x \in \mathcal{D}$, the majorizing property (21) is satisfied.

The local MM framework shares similarities with trust region methods [35], the objective model identifying here with the quadratic majorant and the trust region with the domain over which the majorant is valid. This kind of local majoration strategy has already been successfully used for the proximal-based MM algorithm in [25], leading to a significant speed up. In the case of penalized problems of the form of Problem $(\mathcal{P}_\gamma)$, the use of local majorants is promising. Indeed, when γ is large enough, constraints are likely to be satisfied after some iterations. Then, the penalty term has no more effect locally and therefore the majorant of F_γ can be reduced to the sole contribution of F.

At each iteration k and given x_k, we thus propose to make a finite number of curvature trials for our local majorants of F_γ. For each trial, an MMS update is performed using the corresponding curvature. If the resulting iterate lies in the local domain where the majorizing property (21) is verified, it is accepted and defined as x_{k+1}. Otherwise the curvature of the local majorant is strictly increased (in the sense of Loewner order). Designing this test allows us to guarantee the same descent properties than in the global case as will be shown in next subsection. To construct the sequence of increasing curvatures, we rely on the following definitions.

Definition 4 Let $\mathbf{I} \subseteq \{1, \ldots, r\}$. For every $x \in \mathbb{R}^N$, we define the curvature $A_{F_\gamma}^{(\mathbf{I})}(x) \in \mathcal{S}_+^N$ as

$$A_{F_\gamma}^{(\mathbf{I})}(x) = A_F(x) + \gamma \sum_{i \in \mathbf{I}} A_{R_i}(x), \tag{41}$$

and the associated local domain as

$$\mathscr{D}^{(\mathbf{I})} = \left\{ x \in \mathbb{R}^N \mid (\forall i \notin \mathbf{I}) \;\; x \in C_i \right\}. \tag{42}$$

Note that, for every $x \in \mathbb{R}^N$, $A_{F_\gamma}^{(\mathbf{I})}(x) \preceq A_{F_\gamma}(x)$, where $A_{F_\gamma}(x)$ is defined in (28). In particular, when $\mathbf{I} = \{1, \ldots, r\}$, we have, for every $x \in \mathbb{R}^N$, $A_{F_\gamma}^{(\mathbf{I})}(x) = A_{F_\gamma}(x)$ and $\mathscr{D}^{(\mathbf{I})} = \mathbb{R}^N$. In other words, we recover a global majorant.

Let us now explain how to build local majorants of F_γ at x_k, for a given $k \in \mathbb{N}$. Let

$$\mathbf{I}_k = \{i \in \{1, \ldots, r\} \mid x_k \notin C_i\}. \tag{43}$$

A suitable strategy for majorizing F_γ at x_k then consists of choosing a local majorant which only penalizes the remaining constraints $(C_i)_{i \in \mathbf{I}_k}$. Such a local majorant can be given by

$$Q_k^{(\mathbf{I}_k)} : x \mapsto F_\gamma(x_k) + \nabla F_\gamma(x_k)^\top (x - x_k) + \frac{1}{2}(x - x_k)^\top A_{F_\gamma}^{(\mathbf{I}_k)}(x_k)(x - x_k), \tag{44}$$

on the trust region $\mathscr{D}^{(\mathbf{I}_k)}$, that is

$$(\forall x \in \mathscr{D}^{(\mathbf{I}_k)}) \quad F_\gamma(x) \leq Q_k^{(\mathbf{I}_k)}(x). \tag{45}$$

Consequently, $q_k^{(\mathbf{I}_k)}$ defined as

$$(\forall u \in \mathbb{R}^M) \quad q_k^{(\mathbf{I}_k)}(u) = F_\gamma(x_k) + \nabla F_\gamma(x_k)^\top D_k u + \frac{1}{2} u^\top B_k^{(\mathbf{I}_k)} u, \tag{46}$$

with

$$B_k^{(\mathbf{I}_k)} = D_k^\top A_{F_\gamma}^{(\mathbf{I}_k)}(x_k) D_k, \tag{47}$$

is a local majorant of f_k defined in (38), at $u' = 0$, on the trust region

$$\left\{ u \in \mathbb{R}^{M_k} \mid x_k + D_k u \in \mathscr{D}^{(\mathbf{I}_k)} \right\}. \tag{48}$$

Remark 2 From the definition of $A_{F_\gamma}^{(\mathbf{I}_k)}(x_k)$ in (41), for every $u \in \mathbb{R}^{M_k}$, $q_k^{(\mathbf{I}_k)}(u) \leq q_k(u)$, which means that the local majorants are upper bounded by the global one.

Our resulting local MMS procedure is detailed in Algorithm 2.

Algorithm 2: $\mathrm{MMS}^{\mathrm{loc}}(x_0, \gamma, \varepsilon)$

Inputs: $(\gamma, \varepsilon) \in (0, +\infty)^2$, $x_0 \in \mathbb{R}^N$.
for $k = 0, 1, \ldots$ **do**
  Construct D_k according to (34),
  $x_{k,0} = x_k$, // initialize the current point to x_k
  $\mathbf{I}_{k,0} = \{i \in \{1, \ldots, r\} \mid x_{k,0} \notin C_i\}$,
    // initialize index set with the indices of constraints not satisfied by $x_{k,0}$
  for $\ell = 1, \ldots$ **do**
    Compute $B_k^{(\mathbf{I}_{k,\ell-1})}$ according to (47),
    $u_{k,\ell} = -\left[B_k^{(\mathbf{I}_{k,\ell-1})}\right]^{\dagger} D_k^{\top} \nabla F_\gamma(x_k)$,
    $x_{k,\ell} = x_k + D_k u_{k,\ell}$,
    $\mathbf{I}_{k,\ell} = \mathbf{I}_{k,\ell-1} \cup \{i \in \{1, \ldots, r\} \mid x_{k,\ell} \notin C_i\}$,
      // increment index set with the indices of constraints not satisfied by $x_{k,\ell}$
    if $\mathbf{I}_{k,\ell} = \mathbf{I}_{k,\ell-1}$ **then**
      exit loop, // accept the update $x_{k,\ell}$ if the local majoration property is valid
    end
  end
  $x_{k+1} = x_{k,\ell}$, // define x_{k+1} as $x_{k,\ell}$
  if $\|\nabla F_\gamma(x_{k+1})\| < \varepsilon$ **then**
    exit loop // stop algorithm if given precision ε on the norm of the gradient is reached
    return x_{k+1}.
  end
end

At each iteration $k \in \mathbb{N}$ and, at each MMS trial $\ell \in \mathbb{N}$ of Algorithm 2, a local majorant of f_k is constructed on a trust region which depends on the constraints satisfied at the previous trials $x_{k,0}, \ldots, x_{k,\ell-1}$. The procedure stops when $x_{k,\ell}$ violates no new constraints compared to the previous trials $(x_{k,i})_{0 \leq i \leq \ell-1}$. In other words, $x_{k,\ell}$ belongs to the trust region of the local majorant from which it was computed. The fact that the sequence $(\mathbf{I}_{k,\ell})_{\ell \in \mathbb{N}}$ is monotonically increasing ensures that the method is well defined: in the worst case we end up using the global majorant (i.e., $\mathbf{I}_{k,\ell} = \{1, \ldots, r\}$) and thus the standard MMS iteration is retrieved.

4.2 Convergence analysis of the proposed algorithm

We analyse here the convergence of Algorithm 2. Our main convergence result corresponds to Theorem 1 and does not require any convexity assumption on F_γ.

4.2.1 Assumptions

In addition to Assumption 1-2, our convergence analysis of Algorithm 2 relies on similar assumptions as the ones commonly adopted for MMS [27].

We first make the following assumption ensuring the boundedness of the sequence generated by Algorithm 2.

Assumption 3. *F_γ is coercive on $\mathbb{R}^N$.*

The next assumption controls the eigenvalues of the curvature function $x \mapsto A_{F_\gamma}(x)$.

Assumption 4. *We have the following boundedness properties on the curvature functions $x \mapsto A_F(x)$ and $x \mapsto A_{F_\gamma}(x)$:*
(i) There exists $\eta > 0$ such that, for every $x \in \mathbb{R}^N$,

$$(\forall v \in \mathbb{R}^N) \quad \eta \|v\|^2 \leq v^\top A_F(x) v. \tag{49}$$

(ii) For every nonempty compact set $\mathcal{B} \subseteq \mathbb{R}^N$, there exists $\nu > 0$ such that for every $x \in \mathcal{B}$,

$$(\forall v \in \mathbb{R}^N) \quad v^\top A_{F_\gamma}(x) v \leq \nu \|v\|^2. \tag{50}$$

Note that this last assumption is mild, since Assumption 4(i) can be satisfied by adding an elastic term to F, and Assumption 4(ii) is straightforwardly satisfied when $x \mapsto A_{F_\gamma}(x)$ is continuous, which is often the case. Let us also remark that Assumption 4(i) ensures that for every $\mathbf{I} \subseteq \{1, \ldots, r\}$ and for every $x \in \mathbb{R}^N$,

$$(\forall v \in \mathbb{R}^N) \quad \eta \|v\|^2 \leq v^\top A_{F_\gamma}^{(\mathbf{I})}(x) v. \tag{51}$$

In particular for $\mathbf{I} = \{1, \ldots, r\}$ we obtain

$$(\forall v \in \mathbb{R}^N) \quad \eta \|v\|^2 \leq v^\top A_{F_\gamma}(x) v, \tag{52}$$

which is a standard assumption in the analysis of MM and MMS methods [4,27,28].

Moreover, we assume that the set of directions $(D_k)_{k \in \mathbb{N}}$ fulfills the following condition.

Assumption 5. *There exist $(\mu_0, \mu_1) \in (0, +\infty)^2$ such that for every $k \in \mathbb{N}$, the first column d_k^1 of direction matrix $D_k \in \mathbb{R}^{N \times M_k}$ (with $1 \leq M_k \leq N$) satisfies:*

$$\nabla F_\gamma(x_k)^\top d_k^1 \leq -\mu_0 \|\nabla F_\gamma(x_k)\|^2, \tag{53}$$

$$\|d_k^1\| \leq \mu_1 \|\nabla F_\gamma(x_k)\|. \tag{54}$$

Assumption 5 is satisfied by classical subspace directions, including all those listed in Example 3 (see also [26] for a discussion on this assumption).

We finally assume that the penalized function F_γ verifies the following Kurdyka-Lojasiewicz (KL) property.

Assumption 6. *For all $\tilde{x} \in \mathbb{R}^N$, there exist $\beta \in (0, +\infty)$, $\zeta \in (0, +\infty)$ and $\theta \in [0, 1)$ (all depending of γ) such that for every $x \in \mathbb{R}^N$ verifying $|F_\gamma(x) - F_\gamma(\tilde{x})| \leq \zeta$,*

$$\|\nabla F_\gamma(x)\| \geq \beta |F_\gamma(x) - F_\gamma(\tilde{x})|^\theta. \tag{55}$$

In recent years, this property has become a fundamental tool for convergence analysis in a nonconvex setting [11,12]. Functions satisfying the KL property include semi-algebraic functions and real analytic functions. In particular, the set of semi-algebraic functions is stable under various operations (sum, product, composition...) and contains polynomials, square root, distance to a semi-algebraic set, etc. The KL assumption on F_γ is not restrictive for standard choices of F and R. As we will see in the applications in Section 6, this assumption is met for a wide range of functions used in the context of image recovery. For instance, the standard Gaussian log-likelihood $x \mapsto \|Hx - y\|^2$ with $H \in \mathbb{R}^{N \times M}$ and $y \in \mathbb{R}^M$, and classical regularizations such as ℓ_1 or total variation are semi-algebraic. Concerning the penalty function R, if we suppose it to be of the form given in Example 1, it is semi-algebraic as soon as ψ and f are. Therefore, constraints that can be taken into account include polynomial constraints and distance to balls or ellipsoids.

4.2.2 Convergence proof

Our proof follows the same structure as the convergence proof of MMS [27]. We adapt the descent lemmas to the local case and conclude using [27, Theorem 3].

Lemma 1 *Suppose that Assumptions 1, 2, and 4 are satisfied. Let $x_0 \in \mathbb{R}^N$ be given. Then, the sequence $(x_k)_{k\in\mathbb{N}}$ produced by Algorithm 2 with $\varepsilon = 0$ satisfies for every $k \in \mathbb{N}$,*

$$F_\gamma(x_k) - F_\gamma(x_{k+1}) \geq \frac{\eta}{2}\|x_{k+1} - x_k\|^2, \tag{56}$$

where η is defined in (49).

Proof. Since Algorithm 2 is well-defined, there exists $\ell \in \mathbb{N}^*$ such that $x_{k+1} = x_{k,\ell} = x_k + D_k u_{k,\ell}$, $u_{k,\ell} \in \mathbb{R}^{M_k}$. By definition of $u_{k,\ell}$ we have

$$D_k^\top \nabla F_\gamma(x_k) = -B_k^{(\mathbf{I}_{k,\ell-1})} u_{k,\ell}, \tag{57}$$

for some index set $\mathbf{I}_{k,\ell-1} \subseteq \{1, \dots, r\}$ built from the previous trials $x_k = x_{k,0}, \dots, x_{k,\ell-1}$. Using the expression (46) of $q_k^{(\mathbf{I}_{k,\ell-1})}$,

$$\begin{aligned} F_\gamma(x_k) - q_k^{(\mathbf{I}_{k,\ell-1})}(u_{k,\ell}) &= -\nabla F_\gamma(x_k)^\top D_k^\top u_{k,\ell} - \frac{1}{2}(u_{k,\ell})^\top B_k^{(\mathbf{I}_{k,\ell-1})} u_{k,\ell} \\ &= (u_{k,\ell})^\top B_k^{(\mathbf{I}_{k,\ell-1})} u_{k,\ell} - \frac{1}{2}(u_{k,\ell})^\top B_k^{(\mathbf{I}_{k,\ell-1})} u_{k,\ell} \\ &= \frac{1}{2}(u_{k,\ell})^\top B_k^{(\mathbf{I}_{k,\ell-1})} u_{k,\ell}. \end{aligned} \tag{58}$$

We now use the fact that $q_k^{(\mathbf{I}_{k,\ell-1})}$ is a local tangent majorant of f_k at $u' = 0$ on the trust region

$$\left\{ u \in \mathbb{R}^M \mid x_k + D_k u \in \mathscr{D}^{(\mathbf{I}_{k,\ell-1})} \right\}.$$

By definition of ℓ, $u_{k,\ell}$ belongs to the above set, hence

$$\begin{aligned} q_k^{(\mathbf{I}_{k,\ell-1})}(u_{k,\ell}) &\geq f_k(u_{k,\ell}) \\ &= F_\gamma(x_k + D_k u_{k,\ell}) \\ &= F_\gamma(x_{k+1}). \end{aligned} \tag{59}$$

Finally, from Assumption 4(i), (58), and (59),

$$\begin{aligned} F_\gamma(x_k) - F_\gamma(x_{k+1}) &\geq \frac{1}{2}(u_{k,\ell})^\top B_k^{(\mathbf{I}_{k,\ell-1})} u_{k,\ell} \\ &= \frac{1}{2}(D_k u_{k,\ell})^\top A_{F_\gamma}^{(\mathbf{I}_{k,\ell-1})}(x_k) D_k u_{k,\ell} \\ &\geq \frac{\eta}{2}\|D_k u_{k,\ell}\|^2 \\ &= \frac{\eta}{2}\|x_{k+1} - x_k\|^2. \end{aligned} \tag{60}$$

□

Corollary 1 *Suppose that Assumptions 1-4 are satisfied. Let $x_0 \in \mathbb{R}^N$ be given. Then the sequence $(x_k)_{k\in\mathbb{N}}$ generated by Algorithm 2 with $\varepsilon = 0$ is bounded.*

Proof. From Lemma 1, $(F_\gamma(x_k))_{k\in\mathbb{N}}$ is nonincreasing, thus for every $k \in \mathbb{N}$, $F_\gamma(x_k) \leq F_\gamma(x_0)$. We deduce that $(F_\gamma(x_k))_{k\in\mathbb{N}}$ is bounded from above and the boundedness of $(x_k)_{k\in\mathbb{N}}$ follows from Assumption 3. □

Lemma 2 *Suppose that Assumptions 1, 2, 4 and 5 hold. Let $x_0 \in \mathbb{R}^N$ be given. Then, there exists $\nu > 0$ such that the sequence $(x_k)_{k\in\mathbb{N}}$ produced by Algorithm 2 with $\varepsilon = 0$ satisfies, for every $k \in \mathbb{N}$,*

$$F_\gamma(x_k) - F_\gamma(x_{k+1}) \geq \frac{1}{2}\frac{\mu_0^2}{\nu\mu_1^2}\|\nabla F_\gamma(x_k)\|^2, \tag{61}$$

where μ_0 and μ_1 are defined in Assumption 5.

Proof. Fix $k \in \mathbb{N}$. First, note that if the first direction in D_k satisfies $d_k^1 = 0$, then Assumption 5 implies that $\nabla F_\gamma(x_k) = 0$. Thus $x_{k+1} = x_k$ and (61) is satisfied. We shall now assume that $d_k^1 \neq 0$. In the same way as in the previous proof, there exists $\ell \in \mathbb{N}^*$ such that $x_{k+1} = x_{k,\ell} = x_k + D_k u_{k,\ell}$. Let

$$(\forall t \in \mathbb{R}) \quad h_k(t) = q_k([t, 0, \ldots, 0]), \tag{62}$$

where q_k is defined in (39). Expanding h_k yields

$$h_k(t) = F_\gamma(x_k) + t\nabla F_\gamma(x_k)^\top d_k^1 + \frac{t^2}{2}(d_k^1)^\top A_{F_\gamma}(x_k) d_k^1. \tag{63}$$

Because of Assumption 4(i), h_k is a strictly convex quadratic function which achieves its minimum at

$$\widehat{t}_k = \frac{-\nabla F_\gamma(x_k)^\top d_k^1}{(d_k^1)^\top A_{F_\gamma}(x_k) d_k^1}, \tag{64}$$

with minimum value

$$h_k(\widehat{t}_k) = F_\gamma(x_k) + \frac{1}{2}\widehat{t}_k \nabla F_\gamma(x_k)^\top d_k^1. \tag{65}$$

Recall that by definition, $u_{k,\ell}$ minimizes $q_k^{(\mathbf{I}_{k,\ell-1})}$, defined in (46). It follows from Remark 2 that

$$\begin{aligned} q_k^{(\mathbf{I}_{k,\ell-1})}(u_{k,\ell}) &= \min_{u \in \mathbb{R}^{M_k}} q_k^{(\mathbf{I}_{k,\ell-1})}(u) \\ &\leq \min_{u \in \mathbb{R}^{M_k}} q_k(u) \\ &\leq h_k(\widehat{t}_k) \\ &= F_\gamma(x_k) + \frac{1}{2}\widehat{t}_k \nabla F_\gamma(x_k)^\top d_k^1, \end{aligned} \tag{66}$$

which can be rewritten as

$$F_\gamma(x_k) - q_k^{\mathbf{I}^{(\ell)}}(u_{k,\ell}) \geq -\frac{1}{2}\widehat{t}_k \nabla F_\gamma(x_k)^\top d_k^1. \tag{67}$$

In addition, since $q_k^{(\mathbf{I}_{k,\ell-1})}$ is a local tangent majorant of f_k at $u' = 0$ on a trust region containing $u_{k,\ell}$,

$$\begin{aligned} q_k^{(\mathbf{I}_{k,\ell-1})}(u_{k,\ell}) &\geq f_k(u_{k,\ell}) \\ &= F_\gamma(x_k + D_k u_{k,\ell}) \\ &= F_\gamma(x_{k+1}). \end{aligned} \tag{68}$$

Coupling (67) and (68), we obtain:

$$F_\gamma(x_k) - F_\gamma(x_{k+1}) \geq -\frac{1}{2}\widehat{t}_k \nabla F_\gamma(x_k)^\top d_k^1. \tag{69}$$

Let us now use Assumptions 4 and 5. From Lemma 1, $(x_k)_{k\in\mathbb{N}}$ belongs to a compact set $\mathcal{B} \subseteq \mathbb{R}^N$. Let ν be given by Assumption 4(ii) for this compact set. Firstly, according to (54),

$$\widehat{t}_k \leq -\frac{\nabla F_\gamma(x_k)^\top d_k^1}{\nu ||d_k^1||^2} \leq -\frac{\nabla F_\gamma(x_k)^\top d_k^1}{\nu \mu_1^2 \|\nabla F_\gamma(x_k)\|^2}. \tag{70}$$

Moreover, according to (53),

$$\left(\nabla F_\gamma(x_k)^\top d_k^1\right)^2 \geq \mu_0^2 \|\nabla F_\gamma(x_k)\|^4. \tag{71}$$

Combining (69), (70), and (71), the desired inequality is obtained

$$F_\gamma(x_k) - F_\gamma(x_{k+1}) \geq \frac{1}{2}\frac{\mu_0^2}{\nu\mu_1^2}\|\nabla F_\gamma(x_k)\|^2. \tag{72}$$

□

Finally the following theorem can be deduced as a consequence of the previous lemmas and [27, Theorem 3], which was established using the KL Assumption 6.

Theorem 1 *Suppose that Assumptions 1-6 are satisfied. Let $x_0 \in \mathbb{R}^N$ be given. Then, for every $\epsilon \in (0, +\infty)$ Algorithm 2 stops after a finite number of iterations K_ϵ. In addition, there exists a critical point $\widehat{x}$ of F_γ such that $x_{K_\epsilon} \to \widehat{x}$ as $\epsilon \to 0$.*

5 Embedding the proposed algorithm in a sequential penalty framework

In the previous section, we have presented a new MM-based subspace algorithm for the resolution of penalized problems of the form $(\mathcal{P}_\gamma)$. Coming back to our main objective of solving the constrained problem $(\mathcal{P})$, we propose to embed Algorithm 2 in an exterior penalty method by progressively increasing the penalty parameter. This section describes the proposed iterative approach and provides a convergence proof for it.

5.1 Inexact exterior penalty method

Exterior penalty methods solve Problem $(\mathcal{P})$ by recasting it into a sequence of unconstrained subproblems of the form $(\mathcal{P}_\gamma)$, for $\gamma > 0$ growing to infinity [10,14]. The motivation is the following. Let $(\gamma_j)_{j\in\mathbb{N}}$ be a real sequence of positive reals such that $\lim_{j\to+\infty} \gamma_j = +\infty$. Then, for every $x \in \mathbb{R}^N$,

$$\gamma_j R(x) \underset{j\to+\infty}{\longrightarrow} \iota_C(x). \tag{73}$$

Thus, for some j sufficiently large, one may expect any solution to $(\mathcal{P}_{\gamma_j})$ to be close to a solution of $(\mathcal{P})$. This intuition turns out to be valid under the assumption that F and R are lower semi-continuous, which is covered by Assumption 1 (see [10,14,57]). More precisely, let $(\gamma_j)_{j\in\mathbb{N}}$ be set as stated before, and denote by x_j a solution to $(\mathcal{P}_{\gamma_j})$ for $j \in \mathbb{N}$. Then, $(\mathsf{x}_j)_{j\in\mathbb{N}}$ is bounded and any of its cluster points is a solution to $(\mathcal{P})$ [57, Chap. 10, Thm. 1]. However, for obvious practical reasons, the penalized subproblems $(\mathcal{P}_{\gamma_j})_{j\in\mathbb{N}}$ cannot be

solved exactly. It is actually not even desirable to solve them with high accuracy before reaching large values of γ_j as this would be of limited interest while inducing a high computation cost.

Following [49,44], we propose to solve the subproblems $(\mathcal{P}_{\gamma_j})_{j\in\mathbb{N}}$ with an increasing accuracy. More precisely, given $j \in \mathbb{N}$, an initial point $\mathsf{x}_j^{(0)} \in \mathbb{R}^N$, a penalty parameter $\gamma_j \in (0,+\infty)$ and a precision $\varepsilon_j \in (0,+\infty)$, Algorithm 2 generates a vector $\mathsf{x}_j \in \mathbb{R}^N$ such that

$$\|\nabla F_{\gamma_j}(\mathsf{x}_j)\| < \varepsilon_j. \tag{74}$$

The loop over $j \in \mathbb{N}$ in Algorithm 3 leads to an inexact exterior penalty method, for which we provide a convergence analysis in the remaining of this section. We denote by P-MMS$^{\mathrm{loc}}$ (resp. P-MMS) the penalized MMS algorithm with (resp. without) local acceleration.

Algorithm 3: P-MMS$^{(\mathrm{loc})}$

Inputs: $(\gamma_j)_{j\in\mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$, $(\varepsilon_j)_{j\in\mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$.
for $j = 0, 1, \ldots$ **do**
  Set initial point $\mathsf{x}_j^{(0)}$,
  $\mathsf{x}_j = \mathrm{MMS}^{(\mathrm{loc})}(\mathsf{x}_j^{(0)}, \gamma_j, \varepsilon_j)$, // find an inexact solution to $(\mathcal{P}_{\gamma_j})$
end
return x_j.

5.2 Assumptions

We introduce the following classical assumption on the sequence of penalty and precision parameters involved in our method.

Assumption 7. *The sequences of parameters $(\varepsilon_j)_{j\in\mathbb{N}}$ and $(\gamma_j)_{j\in\mathbb{N}}$ satisfy:*
(i) For every $j \in \mathbb{N}$, $\varepsilon_j > 0$ and $\lim_{j\to+\infty} \varepsilon_j = 0$.
(ii) $(\gamma_j)_{j\in\mathbb{N}}$ is nondecreasing sequence of positive reals and $\lim_{j\to+\infty} \gamma_j = +\infty$.

In addition, the convergence analysis hereafter is led under the following assumption on the involved functions.

Assumption 8. *There exists some $j^* \in \mathbb{N}$ that such, for every $j \geq j^*$, F_{γ_j} is convex.*

Note that the above assumption implies that F is convex on C, but it can be nonconvex on the whole space as shown next and illustrated in Figure 1.

Example 4 Assume that C is the closed ball of $\mathbb{R}^N$ with center 0 and radius $\rho \in (0,+\infty)$. Let $R = \mathrm{d}_C^2/2$ and let F be a continuously differentiable function which is convex on C and twice-differentiable on $\mathbb{R}^N \setminus C$ with

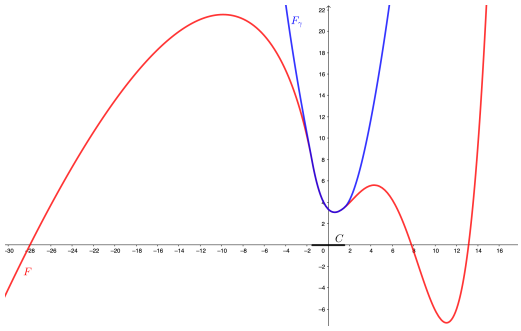


Fig. 1 Illustration of Example 4 in the one-dimensional case. C is the closed ball of $\mathbb{R}$ of center 0 and radius $\rho = 1.5$, and $R = \mathrm{d}_C^2/2$. F is defined as $F(x) = 14.4\sqrt{|x|} + \ln(1+|x|)x - x - 10$ if $x \leq -\rho$, and $F(x) = P(x)$ if $x > -\rho$, where $P(x) = ax^4 + bx^3 + cx^2 - x + e$, is the unique polynomial function such that $P(-\rho) = F(-\rho)$, $P'(-\rho) = F'(-\rho)$, $P'(0.6) = 0$ and $P'(11.07) = 0$. On this illustration, F corresponds to the red curve and F_γ with $\gamma = 2$ to the blue curve. One can note that F is convex on C and nonconvex on $\mathbb{R}$, but for $\gamma \geq 2$, F_γ is convex. Moreover F_γ is coercive for any $\gamma > 0$.

its Hessian $\nabla^2 F$ satisfying the following property: there exists $(\alpha,\delta) \in (0,+\infty)^2$ with $\alpha \geq \rho\delta$ such that

$$(\forall x \in \mathbb{R}^N \setminus C) \quad \nabla^2 F(x) - \frac{\alpha}{\|x\|}\Big(I_N - \frac{xx^\top}{\|x\|^2}\Big) \succeq -\delta I_N. \tag{75}$$

Then F_γ is convex if

$$\gamma\rho \geq \alpha. \tag{76}$$

Indeed (75) means that G defined as

$$(\forall x \in \mathbb{R}^N) \quad G(x) = F(x) - \alpha\|x\| + \frac{\delta}{2}\|x\|^2 \tag{77}$$

is a convex function on every convex subset of $\mathbb{R}^N \setminus C$. We have then

$$(\forall x \in \mathbb{R}^N \setminus C) \quad F_\gamma(x) = G(x) + \Delta(x), \tag{78}$$

where

$$\begin{aligned}\Delta(x) &= \alpha\|x\| - \frac{\delta}{2}\|x\|^2 + \frac{\gamma}{2}\mathrm{d}_C^2(x)\\ &= \alpha\|x\| - \frac{\delta}{2}\|x\|^2 + \frac{\gamma}{2}(\|x\| - \rho)^2.\end{aligned} \tag{79}$$

For every $x \in \mathbb{R}^N \setminus C$,

$$\nabla^2\Delta(x) = (\gamma - \delta)I_N + \frac{\alpha - \gamma\rho}{\|x\|}\Big(I_N - \frac{xx^\top}{\|x\|^2}\Big) \tag{80}$$

If (76) holds, the minimum eigenvalue of $\nabla^2\Delta(x)$ is

$$\begin{aligned}\gamma - \delta + (\alpha - \gamma\rho)/\|x\| &\geq \gamma - \delta + (\alpha - \gamma\rho)/\rho\\ &= \alpha/\rho - \delta \geq 0,\end{aligned} \tag{81}$$

which shows that Δ is convex on every convex subset of $\mathbb{R}^N \setminus C$. We deduce from (78) that F_γ shares the same property. Since F is $\mathcal{C}^1$ and convex on C and $\nabla \mathrm{d}_C^2 = 2(\mathrm{Id} - \mathrm{proj}_C)$ is continuous, F_γ is also $\mathcal{C}^1$ and convex on C. F_γ is thus convex on $\mathbb{R}^N$.
Note that a sufficient condition for (75) to be satisfied with $\alpha = \rho\delta$ is

$$(\forall x \in \mathbb{R}^N \setminus C) \quad \lambda_{\min}\big(\nabla^2 F(x)\big) \geq -\delta \left(1 - \frac{\rho}{\|x\|}\right), \quad (82)$$

where $\lambda_{\min}\big(\nabla^2 F(x)\big)$ is the minimum eigenvalue of the Hessian of F at x.

5.3 Convergence analysis

Our convergence result is given in Theorem 2 and exhibits some similarity to the one obtained in [57, Chap. 10, Thm. 1] for the classical exact exterior penalty method.

Lemma 3 *Suppose that Assumptions 1 and 2 hold. Let $(\varepsilon_j)_{j\in\mathbb{N}}$ and $(\gamma_j)_{j\in\mathbb{N}}$ be two sequences satisfying Assumption 7. Suppose that Assumptions 3-6 and 8 are satisfied by $(F_{\gamma_j})_{j\in\mathbb{N}}$. Then the sequence $(\mathsf{x}_j)_{j\in\mathbb{N}}$ generated by Algorithm 3 is bounded.*

Proof. For every $j \in \mathbb{N}$, since Assumptions 1-6 are satisfied by F_{γ_j}, the convergence results established in Section 4.2.2 apply. Thus Algorithm 2 produces, in a finite number of iterations, a vector $\mathsf{x}_j \in \mathbb{R}^N$ verifying (74). This shows that Algorithm 3 is well-defined.

Let us now prove the boundedness of sequence $(\mathsf{x}_j)_{j\in\mathbb{N}}$. By definition of sequence $(\mathsf{x}_j)_{j\in\mathbb{N}}$ and because of the convexity of functions $(F_{\gamma_j})_{j\geq j^*}$, the Cauchy-Schwarz inequality yields for every $j \geq j^*$ and $z \in \mathbb{R}^N$,

$$\begin{aligned} F_{\gamma_j}(\mathsf{x}_j) - F_{\gamma_j}(z) &\leq \langle \nabla F_{\gamma_j}(\mathsf{x}_j), \mathsf{x}_j - z\rangle, \\ &< \|\nabla F_{\gamma_j}(\mathsf{x}_j)\| \|\mathsf{x}_j - z\|, \\ &\leq \varepsilon_j \|\mathsf{x}_j - z\|. \end{aligned} \quad (83)$$

We then argue by contradiction. Assume that $(\mathsf{x}_j)_{j\in\mathbb{N}}$ is unbounded. Then there exists a subsequence of nonzero vectors $(\mathsf{x}_{j_q})_{q\in\mathbb{N}}$ and $\overline{\mathsf{x}} \in \mathbb{R}^N$ such that

$$\lim_{q\to+\infty} \|\mathsf{x}_{j_q}\| = +\infty, \quad (84)$$

$$\lim_{q\to+\infty} \frac{\mathsf{x}_{j_q}}{\|\mathsf{x}_{j_q}\|} = \overline{\mathsf{x}}, \quad (85)$$

$$\|\overline{\mathsf{x}}\| = 1. \quad (86)$$

Let $z \in C$. According to (83), for every $q \in \mathbb{N}$ such that $j_q \geq j^*$,

$$\begin{aligned} \frac{F_{\gamma_{j_q}}(\mathsf{x}_{j_q})}{\|\mathsf{x}_{j_q}\|} &< \frac{F_{\gamma_{j_q}}(z)}{\|\mathsf{x}_{j_q}\|} + \varepsilon_{j_q} \left\| \frac{\mathsf{x}_{j_q}}{\|\mathsf{x}_{j_q}\|} - \frac{z}{\|\mathsf{x}_{j_q}\|} \right\| \\ &= \frac{F(z)}{\|\mathsf{x}_{j_q}\|} + \varepsilon_{j_q} \left\| \frac{\mathsf{x}_{j_q}}{\|\mathsf{x}_{j_q}\|} - \frac{z}{\|\mathsf{x}_{j_q}\|} \right\|, \end{aligned} \quad (87)$$

where we have used the fact that $z \in C$, hence $R(z) = 0$, to obtain the equality. Let us take the infimum limit on both sides. It follows from (84), (85), and Assumption 7(i), that

$$\liminf_{q\to+\infty} \frac{F_{\gamma_{j_q}}(\mathsf{x}_{j_q})}{\|\mathsf{x}_{j_q}\|} \leq 0. \quad (88)$$

Let $q_0 \in \mathbb{N}$. By using (84), (85) and the convexity (and continuity) of $F_{\gamma_{j^*}}$,

$$\begin{aligned} &F_{\gamma_{j^*}}\left(z + \|\mathsf{x}_{j_{q_0}}\|\overline{\mathsf{x}}\right) \\ &= \lim_{q\to+\infty} F_{\gamma_{j^*}}\left(\left(1 - \frac{\|\mathsf{x}_{j_{q_0}}\|}{\|\mathsf{x}_{j_q}\|}\right) z + \frac{\|\mathsf{x}_{j_{q_0}}\|}{\|\mathsf{x}_{j_q}\|}\mathsf{x}_{j_q}\right) \\ &\leq \liminf_{q\to+\infty} \left[\left(1 - \frac{\|\mathsf{x}_{j_{q_0}}\|}{\|\mathsf{x}_{j_q}\|}\right) F_{\gamma_{j^*}}(z) + \frac{\|\mathsf{x}_{j_{q_0}}\|}{\|\mathsf{x}_{j_q}\|} F_{\gamma_{j^*}}(\mathsf{x}_{j_q})\right] \\ &\leq \liminf_{q\to+\infty} \left[\left(1 - \frac{\|\mathsf{x}_{j_{q_0}}\|}{\|\mathsf{x}_{j_q}\|}\right) F(z) + \frac{\|\mathsf{x}_{j_{q_0}}\|}{\|\mathsf{x}_{j_q}\|} F_{\gamma_{j_q}}(\mathsf{x}_{j_q})\right]. \end{aligned} \quad (89)$$

The last inequality (89) is a consequence of the monotonicity of $(\gamma_j)_{j\in\mathbb{N}}$ and of the equality of $F_{\gamma_{j^*}}(z)$ and $F(z)$. By using (88), we deduce that

$$F_{\gamma_{j^*}}\left(z + \|\mathsf{x}_{j_{q_0}}\|\overline{\mathsf{x}}\right) \leq F(z). \quad (90)$$

Thus the sublevel set $\{x \in \mathbb{R}^N \mid F_{\gamma_{j^*}}(x) \leq F(z)\}$ contains the unbounded set $\{z + \|\mathsf{x}_{j_{q_0}}\|\overline{\mathsf{x}},\ q_0 \in \mathbb{N}\}$, which contradicts the coercivity of $F_{\gamma_{j^*}}$. □

Theorem 2 *Under the same assumptions as in Lemma 3, the sequence $(\mathsf{x}_j)_{j\in\mathbb{N}}$ generated by Algorithm 3 has at least one cluster point. In addition, every of its cluster points is a solution to Problem $(\mathcal{P})$.*

Proof. According to Lemma 3, $(\mathsf{x}_j)_{j\in\mathbb{N}}$ is bounded. Let $(\mathsf{x}_{j_q})_{q\in\mathbb{N}}$ be a subsequence of $(\mathsf{x}_j)_{j\in\mathbb{N}}$ which converges to a point $\mathsf{x}^* \in \mathbb{R}^N$. We shall show that $\mathsf{x}^* \in C$ and that, for every $z \in C$, $F(\mathsf{x}^*) \leq F(z)$.

Let us first prove that $\mathsf{x}^* \in C$. Let $z \in C$ and $q \in \mathbb{N}$ such that $j_q \geq j^*$. Following the same reasoning as for (83), and using the equality $F_{\gamma_{j_q}}(z) = F(z)$, we have

$$F_{\gamma_{j_q}}(\mathsf{x}_{j_q}) < F(z) + \varepsilon_{j_q}\|\mathsf{x}_{j_q} - z\|. \quad (91)$$

Since the sequence $(\mathsf{x}_{j_q})_{q\in\mathbb{N}}$ is bounded and Assumption 7(i) holds, we deduce that there exists a constant $\nu \in \mathbb{R}$ such that

$$(\forall q \in \mathbb{N}) \quad F_{\gamma_{j_q}}(\mathsf{x}_{j_q}) \leq \nu. \quad (92)$$

In addition, since F is continuous and $(\mathsf{x}_j)_{j\in\mathbb{N}}$ is bounded, $(F(\mathsf{x}_{j_q}))_{q\in\mathbb{N}}$ is bounded from below by a constant $\zeta \in \mathbb{R}$ and the following inequality holds:

$$\begin{aligned} (\forall q \in \mathbb{N}) \quad F_{\gamma_{j_q}}(\mathsf{x}_{j_q}) &= F(\mathsf{x}_{j_q}) + \gamma_{j_q} R(\mathsf{x}_{j_q}) \\ &\geq \zeta + \gamma_{j_q} R(\mathsf{x}_{j_q}). \end{aligned} \quad (93)$$

By combining (92) and (93), we deduce that

$$(\forall q \in \mathbb{N}) \quad 0 \leq R(\mathsf{x}_{j_q}) \leq \frac{\nu - \zeta}{\gamma_{j_q}}. \tag{94}$$

Next, passing to the limit when $q \to +\infty$ in (94) and using Assumption 7(ii), we obtain by continuity of R,

$$R(\mathsf{x}^*) = 0, \tag{95}$$

which, according to Definition 1, is equivalent to $\mathsf{x}^* \in C$.

Let us now show that, for every $z \in C$, $F(\mathsf{x}^*) \leq F(z)$. Equation (91) implies that

$$\begin{aligned} F(\mathsf{x}_{j_q}) &\leq F(z) + \varepsilon_{j_q}\|\mathsf{x}_{j_q} - z\| - \gamma_{j_q} R(\mathsf{x}_{j_q}) \\ &\leq F(z) + \varepsilon_{j_q}\|\mathsf{x}_{j_q} - z\|. \end{aligned} \tag{96}$$

Passing to the limit when $q \to +\infty$ in (96), we obtain by continuity of F,

$$F(\mathsf{x}^*) \leq F(z), \tag{97}$$

which completes the proof. □

Corollary 2 *Under the same assumptions as in Lemma 3, if Problem $(\mathcal{P})$ admits a unique solution $\mathsf{x}^* \in \mathbb{R}^N$, then the sequence $(\mathsf{x}_j)_{j\in\mathbb{N}}$ generated by Algorithm 3 converges to x^*.*

Remark 3

(i) Our convergence proof remains valid if, at each iteration $j \in \mathbb{N}$ of the proposed inexact penalty method, the subproblem $(\mathcal{P}_{\gamma_j})$ is solved by any method returning an inexact minimizer $\mathsf{x}_j \in \mathbb{R}^N$ such that (74) holds.

(ii) The whole theory is conducted here with a single sequence of penalty parameters for the global penalty function R. However, in the case where there are r penalty functions that we wish to weight differently with sequences of parameters $(\gamma_j^{(1)})_{j\in\mathbb{N}}, \ldots, (\gamma_j^{(r)})_{j\in\mathbb{N}}$, a similar proof may be carried.

5.4 Initial guess at each iteration

The choice of the initial point $\mathsf{x}_j^{(0)}$ at each iteration can influence the convergence speed of the algorithm, so one should define it carefully. Ideally, the latter must be determined in such a way that $\|\nabla F_{\gamma_j}(\mathsf{x}_j^{(0)})\|$ is as small as possible, with

$$\nabla F_{\gamma_j}(\mathsf{x}_j^{(0)}) = \nabla F(\mathsf{x}_j^{(0)}) + \gamma_j \nabla R(\mathsf{x}_j^{(0)}). \tag{98}$$

We propose a heuristic algorithm initialization, relying on the two previous iterates, x_{j-1} and x_{j-2}, and on the penalty parameters $\gamma_{j-2},\ \gamma_{j-1},\ \gamma_j$, when $j \geq 2$. In a neighborhood $\mathcal{N}(x^*)$ of the optimum x^* for Problem $(\mathcal{P})$, we suppose that

$$\begin{aligned} (\forall x \in \mathcal{N}(x^*)) \quad \nabla F_\gamma(x) &\simeq \nabla F(x^*) + \gamma \nabla R(x) \\ &\simeq \nabla F(x^*) + 2\gamma(x - x^*). \end{aligned} \tag{99}$$

The latter approximation is valid if x^* lies on the boundary of C, $x \notin C$, and

$$(\forall x \in \mathcal{N}(x^*) \setminus C) \quad R(x) \simeq \|x - x^*\|^2 \tag{100}$$

(e.g., $R = \mathrm{d}_C^2$). Then, using the approximation (99) for the values (γ, x) in $\{(\gamma_{j-2}, \mathsf{x}_{j-2}), (\gamma_{j-1}, \mathsf{x}_{j-1})\}$ and assuming that $\nabla F_{\gamma_{j-2}}(\mathsf{x}_{j-2}) \simeq 0$ and $\nabla F_{\gamma_{j-1}}(\mathsf{x}_{j-1}) \simeq 0$, we obtain

$$\mathsf{x}_{j-1} - \mathsf{x}_{j-2} \simeq \frac{1}{2}\nabla F(x^*)\left(\frac{1}{\gamma_{j-2}} - \frac{1}{\gamma_{j-1}}\right). \tag{101}$$

Similarly, to get $\nabla F_{\gamma_j}(\mathsf{x}_j^{(0)}) \simeq 0$, we should have

$$\mathsf{x}_j^{(0)} - \mathsf{x}_{j-1} \simeq \frac{1}{2}\nabla F(x^*)\left(\frac{1}{\gamma_{j-1}} - \frac{1}{\gamma_j}\right). \tag{102}$$

Gathering (101) and (102) leads to the following choice for the initial point:

$$\mathsf{x}_j^{(0)} = \mathsf{x}_{j-1} + \frac{\frac{1}{\gamma_{j-1}} - \frac{1}{\gamma_j}}{\frac{1}{\gamma_{j-2}} - \frac{1}{\gamma_{j-1}}}(\mathsf{x}_{j-1} - \mathsf{x}_{j-2}). \tag{103}$$

6 Numerical experiments

We consider two applicative scenarios with the aim to highlight the computational efficiency of our approach on large-scale image recovery problems, as well as to illustrate the variety of constraints that can be handled. All algorithms are implemented in Python 3 and the computations are performed on a desktop having an Intel Xeon 3.2 GHz processor and 16 GB of RAM.

6.1 Frame-based image restoration in presence of heteroscedastic noise

The first numerical scenario we consider concerns image restoration in the presence of pixel-dependent additive Gaussian noise. We emphasize that the optimization problem addressed hereafter is a generalization of a standard wavelet denoising problem [3].

We consider an image restoration scenario aiming at recovering an original image $\overline{y} \in \mathbb{R}^M$ from a degraded version of it, $z = (z_i)_{1\leq i\leq M} \in \mathbb{R}^M$, given the observation model

$$z = H\overline{y} + w, \tag{104}$$

where $H \in \mathbb{R}^{M\times M}$ is a matrix modelling a blur and $w = (w_i)_{1\leq i\leq M} \in \mathbb{R}^M$ is an additive noise. The noise is assumed to be zero-mean Gaussian i.i.d. and heteroscedastic (i.e. with non stationary variance). As discussed in [48,75], heteroscedastic noise is commonly encountered when dealing with raw digital images of an under or over-exposed scene, as it is often the case in satellite imaging. A suitable model for the noise variance that was used in [48] is given by

$$(\forall i \in \{1,\dots,M\}) \quad w_i = \sigma((H\overline{y})_i)\, v_i, \tag{105}$$

where $(\cdot)_i$ designates the i-th component of its vector in argument, $(v_i)_{1\leq i\leq M}$ are realizations of independent zero-mean, unit-variance Gaussian random variables, and

$$(\forall t \in \mathbb{R}) \quad \sigma(t) = \begin{cases} \sqrt{at+b} & \text{if } at+b \geq 0 \\ \sqrt{b} & \text{otherwise,} \end{cases} \tag{106}$$

with $(a,b) \in (\mathbb{R}^+)^2$.

6.1.1 Tight frame decomposition

Restoring $\overline{y}$ from its degraded version z is an ill-posed inverse problem which resolution requires the introduction of prior knowledge on the sought image. The prior can either be formulated in the original data space of the image, or through an appropriate linear representation (e.g., Fourier, cosine, wavelet domain) of it. In this experiment, we opt for a sparsity enhancing prior in a frame domain [58]. More precisely, we consider the decomposition of the image on a tight frame [68,20,40] i.e., an overcomplete dictionary of vectors $(e_1,\dots,e_N) \in (\mathbb{R}^M)^N$ with $N > M$ such that the operator

$$W\colon \mathbb{R}^M \mapsto \mathbb{R}^N \ : y \mapsto (\langle y, e_n\rangle)_{1\leq n\leq N} \tag{107}$$

satisfies $W^*W = \mu \mathrm{I}_M$ for some parameter $\mu > 0$, where W^* refers to the adjoint of W. Such an overcomplete decomposition leads to a more flexible linear model than traditional orthonormal representations. In particular, if we set $\mu \in \mathbb{N}^*$, one can build a tight frame with $W^*W = \mu \mathrm{I}_M$ by concatenating μ orthonormal wavelet bases, associated to the decomposition operator

$$W = [W_1^* \,|\, \dots \,|\, W_\mu^*]^* \in \mathbb{R}^{N\times M}, \tag{108}$$

where $N = \mu M$ and $(W_i)_{1\leq i\leq \mu} \in (\mathbb{R}^{M\times M})^\mu$ are orthogonal operators.

6.1.2 Optimization problem

Let $W \in \mathbb{R}^{N\times M}$ be the tight frame decomposition operator introduced above. We propose to follow the so-called synthesis approach for solving the inverse problem (104). This approach consists of estimating $\overline{x} \in \mathbb{R}^N$, the decomposition of the original image $\overline{y}$ onto the tight frame, by solving an optimization problem in the space of the frame coefficients. A solution to the initial Problem (104) is then retrieved as $\overline{y} = W^*\overline{x}$.

When the Gaussian noise is identically distributed for every pixel, the minimization problem on the tight frame coefficients usually takes the form

$$\underset{x\in\mathbb{R}^N}{\text{minimize}} \quad \|HW^*x - z\|^2 + \lambda F(x), \tag{109}$$

where $F\colon \mathbb{R}^N \to \mathbb{R}$ is a regularization function on the frame coefficients, and $\lambda > 0$ a regularization parameter. The main drawback of formulation (109) is the difficulty to adjust the regularization parameter λ. This can be alleviated by considering instead the constrained minimization problem [52,20]:

$$\begin{aligned} &\underset{x\in\mathbb{R}^N}{\text{minimize}} && F(x) \\ &\text{subject to} && \|HW^*x - z\|^2 \leq \alpha, \end{aligned} \tag{110}$$

in which the fidelity to the model is ensured by means of a constraint. By virtue of the law of large numbers, the new parameter $\alpha > 0$ can be estimated as $\alpha \simeq \sigma^2 M$, where σ is the standard deviation of the noise.

In the present case, the heteroscedastic noise follows the standard deviation rule (106) and using directly the formulation (110) might not be appropriate to account for the local properties of such a noise. We thus propose to partition the image into $L \in \mathbb{N}^*$ rectangular blocks of $B \in \mathbb{N}^*$ pixels and to impose a taylored data fidelity constraint on each of these blocks. For every $\ell \in \{1,\dots,L\}$, we denote by $[\cdot]_\ell$ the component corresponding to the ℓ-th block. The minimization problem we propose to solve then reads

$$\begin{aligned} &\underset{x\in\mathbb{R}^N}{\text{minimize}} && F(x), \\ &\text{subject to} && \|[HW^*x - z]_\ell\|^2 \leq \alpha_\ell \ \ (\forall \ell \in \{1,\dots,L\}), \\ & && W^*x \in [0,1]^M. \end{aligned} \tag{111}$$

Hereabove, F is a smooth ℓ_1-regularization promoting sparsity of the frame coefficients chosen as

$$(\forall x = (\xi_n)_{1\leq n\leq N} \in \mathbb{R}^N) \quad F(x) = \sum_{n=1}^{N} \lambda_n \sqrt{\delta + \xi_n^2}, \tag{112}$$

with, for every $n \in \{1,\dots,N\}$, $\lambda_n \in (0,+\infty)$. Moreover, $\delta \in (0,+\infty)$ and $(\alpha_\ell)_{1\leq\ell\leq L} \in (\mathbb{R}^+)^L$ are given hyper-parameters. Note that the optimization problem (111) is an instance of Problem $(\mathcal{P})$ where function F is defined as in (112) and constraint set C corresponds to

$$\Big\{x \in \mathbb{R}^N \mid (\forall \ell \in \{1,\dots,L\})\ \|[HW^*x - z]_\ell\|^2 \leq \alpha_\ell\Big\} \bigcap \big\{x \in \mathbb{R}^N \mid W^*x \in [0,1]^M\big\}. \quad (113)$$

Remark 4 Regarding the partition of the image, we opted here for a rectangular one as it was the most straightforward for the considered image. However one could also consider using a partition based on a segmentation of the degraded image when such a segmentation is available.

6.1.3 Proposed algorithm

We propose to apply the methods P-MMS and P-MMS$^{\text{loc}}$ to the minimization problem (111). To do so, we have to define the penalty functions for the data-fidelity constraints. For every $\ell \in \{1,\dots,L\}$, we set

$$(\forall x \in \mathbb{R}^N) \quad R_{1,\ell}(x) = \mathrm{d}^2_{\mathcal{B}_\ell}([HW^*x]_\ell), \quad (114)$$

where $\mathcal{B}_\ell$ is the Euclidean ball of $\mathbb{R}^B$ centered at $[z]_\ell$ with radius α_ℓ. Similarly, we define the penalty function for the pixel-range constraint by

$$(\forall x \in \mathbb{R}^N) \quad R_2(x) = \tfrac{1}{10}\,\mathrm{d}^2_{[0,1]^M}(W^*x). \quad (115)$$

This latter penalty function will be split into M penalty functions operating on each component of W^*x, which naturally provides a suitable applicative context for the local approach described in Section 4. In a nutshell, the penalty function for Problem (111) reads

$$(\forall x \in \mathbb{R}^N) \quad R(x) = \sum_{\ell=1}^{L} R_{1,\ell}(x) + R_2(x). \quad (116)$$

The penalized MMS method relies on Assumption 2, which requires building explicitly quadratic majorants of the objective F and the penalty functions $(R_{1,\ell})_{1\leq\ell\leq L}$ and R_2. It can be easily shown (see [24] for instance) that, for every $x = (\xi_n)_{1\leq n\leq N} \in \mathbb{R}^N$, the matrix

$$A_F(x) = \mathrm{Diag}\left\{\left(\frac{\lambda_n}{\sqrt{\delta+\xi_n^2}}\right)_{1\leq n\leq N}\right\} \in \mathbb{R}^{N\times N} \quad (117)$$

defines the curvature of a valid quadratic majorant of F at x. Regarding the penalty functions, according to Example 2, quadratic majorants for $(R_{1,\ell})_{1\leq\ell\leq L}$ are given by the curvature matrices

$$(\forall \ell \in \{1,\dots,L\})(\forall x \in \mathbb{R}^N) \quad A_{R_{1,\ell}}(x) = 2WH^\top T_\ell HW^*, \quad (118)$$

where $T_\ell = \mathrm{Diag}\left\{(t_i)_{1\leq i\leq M}\right\} \in \mathbb{R}^{M\times M}$ with $t_i = 1$ if the i-th pixel belongs to the ℓ-th block, $t_i = 0$ otherwise. Similarly, we define

$$(\forall x \in \mathbb{R}^N) \quad A_{R_2}(x) = \tfrac{1}{5}\,WW^*. \quad (119)$$

By assuming that set C defined by (113) is nonempty[1], for both P-MMS and P-MMS$^{\text{loc}}$, the convergence conditions introduced in Section 5.3 are met. Indeed, Assumptions 1-4 and Assumption 6 are fulfilled by functions F and R. By choosing the subspace as the memory gradient one in (35), Assumption 5 is satisfied. Finally, penalty parameters $(\gamma_j)_{j\in\mathbb{N}}$ in P-MMS or P-MMS$^{\text{loc}}$ were set to

$$\gamma_j = (3j)^{0.8} \quad (120)$$

and the precision parameters $(\varepsilon_j)_{j\in\mathbb{N}}$ were chosen as

$$\varepsilon_j = 40/(\gamma_j)^{0.25}, \quad (121)$$

so that the requirements in Assumption 7 are satisfied. Assumption 8 also holds since F and R are convex.

6.1.4 Results

For our numerical experiment, we consider the satellite `zaghouane` image, of size $M = 512 \times 512$, blurred by an 11×11 Gaussian kernel with standard deviation 1.7 using symmetric boundary conditions. The image is corrupted with a Gaussian noise following the standard deviation rule (106), with $a = 10^{-4}$ and $b = 10^{-3}$ (Fig. 2 (left)). The tight frame operator W is defined as a concatenation of $\mu = 3$ orthonormal wavelet bases, each one with 4 resolution levels: Daubechies of order 2, Coiflets of order 3, and Symlets of order 6 (respectively `db2`, `coif3`, `sym6` from the `pywt` Python package). The size of the considered optimization problem is therefore $N = 3M$. In the numerical implementation of the algorithm, the operators W and W^* are not stored as matrices; they are defined as functions returning a matrix-vector product relying on the discrete wavelet transform of Python. In (112), $\lambda_n = 1$ if ξ_n is a detail (i.e. high frequency) wavelet coefficient, whereas $\lambda_n = 10^{-4}$ if ξ_n is an approximation (i.e. low frequency) coefficient. The image is divided into a partition of $L = 256$ squares of size $B = 32 \times 32$ pixels. The hyper-parameters are set to $\delta = 10^{-4}$ and, for every $\ell \in \{1,\dots,L\}$, $\alpha_\ell = 0.9\,\widehat{\sigma_\ell}^{\,2} B$, where $\widehat{\sigma}_\ell$ is the empirical standard deviation of the noise on the ℓ-th block, estimated as

$$\widehat{\sigma_\ell}^{\,2} \simeq \frac{a}{B} \sum_{i\in\ell\text{-th block}} z_i + b. \quad (122)$$

[1] Since W^* is surjective, this means that there exists $y \in [0,1]^M$ such that, for every $\ell \in \{1,\dots,L\}$, $\|[Hy - z]_\ell\|^2 \leq \alpha_\ell$.

We compare the convergence speed of P-MMS and its local acceleration P-MMS$^{\text{loc}}$, with four state-of-the-art algorithms, namely the primal-dual Condat-Vũ algorithm (CV) [32,77], the parallel proximal algorithm (PPXA+) [31], FISTA [8] and the SQP trust region algorithm of Byrd et al. [18] (trust-constr). FISTA is implemented using the improved scheme [23] where subiterations for computing the proximity operator are performed with an accelerated dual Forward-Backward algorithm [1]. The trust region method is a solver already implemented in the Python package `scipy.optimize`. Note that PPXA+ is similar to a parallel version of ADMM [70,15].

The restored image using formulation (111) is displayed in Fig. 2 (right). We compare, in Fig. 3, the convergence speed of the algorithms in terms of the relative distance between the outer iterate (corresponding to x_j) and the limit point x_∞, computed after a large number of iterations for each method. One can observe that the proposed method P-MMS$^{\text{loc}}$ outperforms all the other competitors. The advantage of resorting to the local majoration strategy can be observed when comparing P-MMS and P-MMS$^{\text{loc}}$ curves.

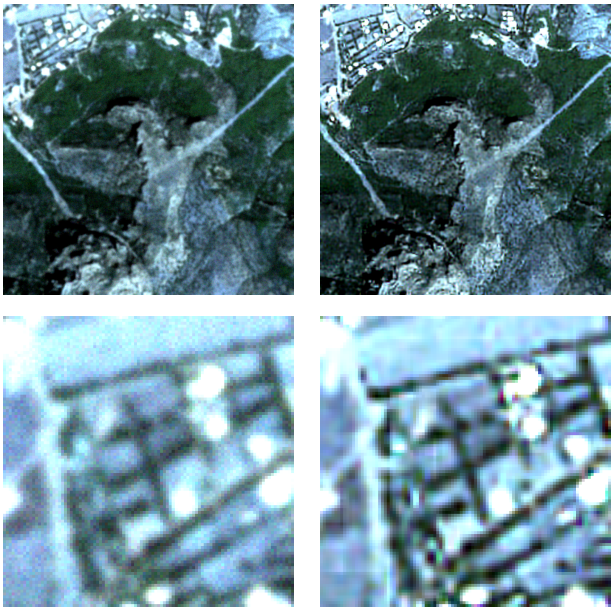


Fig. 2 (top left) Degraded image, PNSR = 29.90dB ; (top right) Restored image, PNSR = 31.58dB; (bottom left) Zoom on the degraded image ; (bottom right) Zoom on the restored image.

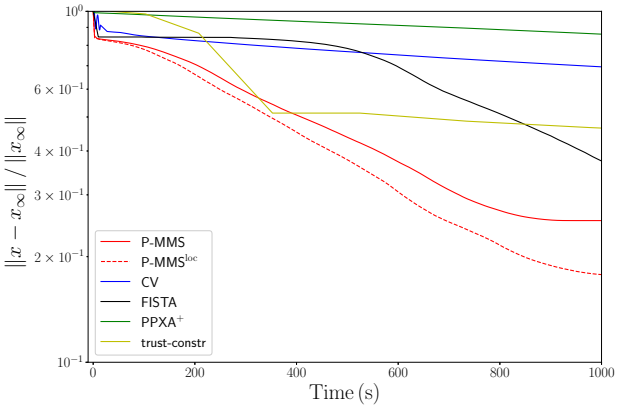


Fig. 3 Distance to the optimum versus time.

6.2 Image reconstruction in the presence of Poisson noise

Our second example focuses on a medical imaging reconstruction problem, typically encountered in Positron Emission Tomography (PET) imaging.

6.2.1 Positron Emission Tomography

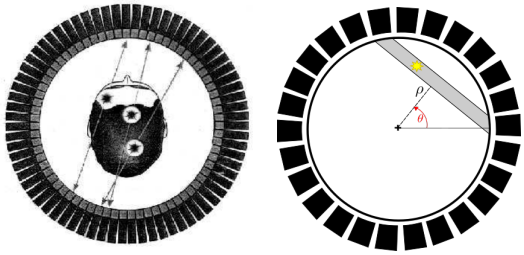
PET is a medical imaging technique used to detect the presence of specific molecules enriched in tissues (e.g. tumors) of an organ by means of an appropriate tracer injected to the patient. The tracer is marked by a radioactive element that emits a positron when it disintegrates. The positron then annihilates with an electron. This results in the emission of two photons travelling in two opposite directions (see Fig. 4(a)), making it possible to identify a "line-of-response" (LOR) in which the annihilation occurred (see Fig. 4(b)). The PET acquisition of a 2D scene, represented as a vector $\overline{x} \in \mathbb{R}^N$, is a so-called sinogram $y \in \mathbb{R}^M$ gathering the number of disintegrations detected per pair of detectors. This leads to the observation model:

$$y = \mathcal{P}(H\overline{x}), \tag{123}$$

where $H \in \mathbb{R}^{M\times N}$ is the projection matrix whose elements $H_{m,n}$ represents the intersection area between the n-th pixel and the m-th LOR, and $\mathcal{P}$ models a Poisson noise.

6.2.2 Considered optimization problem

We propose to tackle the inverse problem (123) of reconstructing $\overline{x}$ from the sinogram y and the geometry matrix H, by minimizing a data fidelity function reflecting the properties of Poisson noise, under a Total Variation (TV) regularization constraint and a pixel range constraint. A typical choice to account for Poisson noise



(a) (b)

Fig. 4 (a) Two photons are emitted in opposite directions and are detected almost simultaneously by two different detectors. (b) The annihilation is assumed to occur within an LOR delimited by the two detectors [67].

[39,52] consists of making use of the Anscombe variance stabilizing transform [5],

$$T : \quad [0,+\infty)^M \longrightarrow [0,+\infty)^M \qquad (124)$$
$$u = (u_i)_{1\leq i\leq M} \longmapsto \left(2\sqrt{u_i+\tfrac{3}{8}}\right)_{1\leq i\leq M},$$

which approximately transforms the Poisson noise into an i.i.d. Gaussian noise with zero-mean and unit variance. The associated data-fidelity function is then defined as

$$(\forall x\in\mathcal{D}) \quad F(x) = \|T(Hx)-T(y)\|^2, \qquad (125)$$

with the domain

$$\mathcal{D} = \{x\in\mathbb{R}^N \mid Hx\in[0,+\infty)^M\}. \qquad (126)$$

One can rewrite (125), using (124):

$$(\forall x\in\mathcal{D}) \quad F(x) = 4\sum_{i=1}^{M}\left((Hx)_i+\frac{3}{8}\right) + 4\sum_{i=1}^{M}(T(y))_i\ \rho((Hx)_i) + \|T(y)\|^2, \qquad (127)$$

where $(\cdot)_i$ denotes the i-th component and

$$(\forall \upsilon\in[0,+\infty)) \quad \rho(\upsilon) = -\sqrt{\upsilon+3/8}. \qquad (128)$$

In order to extend the domain of F to $\mathbb{R}^N$, we introduce the above differentiable extension of ρ on $\mathbb{R}$:

$$(\forall \upsilon\in\mathbb{R}) \quad \overline{\rho}(\upsilon) = \begin{cases} \rho(\upsilon) & \text{if } \upsilon\geq 0,\\ -\sqrt{\frac{2}{3}}\upsilon - \sqrt{3/8} & \text{if } \upsilon<0, \end{cases} \qquad (129)$$

which yields the following extension of F on the whole space:

$$(\forall x\in\mathbb{R}^N) \quad F(x) = 4\sum_{i=1}^{M}\left((Hx)_i+\frac{3}{8}\right) + 4\sum_{i=1}^{M}(T(y))_i\ \overline{\rho}((Hx)_i) + \|T(y)\|^2. \qquad (130)$$

The optimization problem to tackle is then

$$\begin{array}{ll} \underset{x\in\mathbb{R}^N}{\text{minimize}} & F(x),\\ \text{subject to} & \mathrm{TV}(x)\leq\alpha \text{ and } x\in[0,x_{\max}]^N, \end{array} \qquad (131)$$

where $\alpha\in(0,+\infty)$ is the TV bound acting as a regularization parameter, and $x_{\max}\in(0,+\infty)$ is the maximal pixel intensity of the sought image $\overline{x}$. Moreover, TV stands for the total variation semi-norm [69]:

$$(\forall x\in\mathbb{R}^N) \quad \mathrm{TV}(x) = \sum_{n=1}^{N}\|(Gx)_n\|_2, \qquad (132)$$

where $G\in\mathbb{R}^{2N\times N}$ is the 2D discrete gradient operator. A natural choice for the regularization parameter would be $\alpha=\mathrm{TV}(\overline{x})$. As discussed in [30], opting for a slightly smaller value of α (and thus over-regularizing) can improve the reconstruction performance.

The optimization problem (131) is an instance of Problem $(\mathcal{P})$ with F given by (130) and

$$C = \{x\in\mathbb{R}^N \mid \mathrm{TV}(x)\leq\alpha \text{ and } x\in[0,x_{\max}]^N\}. \qquad (133)$$

As we will see, our penalized MMS approach, although designed for differentiable objective functions, is able to handle nonsmooth regularization constraints such as the TV constraint involved here.

6.2.3 Proposed algorithm

We propose to apply the penalized MMS method described in Sections 4 and 5 to the resolution of (131). To do so, we introduce the penalty function for the TV constraint:

$$(\forall x\in\mathbb{R}^N) \quad R_1(x) = \mathrm{d}^2_{\mathcal{B}}(Gx), \qquad (134)$$

where $\mathcal{B}$ is the $\ell_{1,2}$-ball with radius α, i.e.

$$\mathcal{B} = \{g=(g_n)_{1\leq n\leq N}\in(\mathbb{R}^2)^N \mid \sum_{n=1}^{N}\|g_n\|_2\leq\alpha\}. \qquad (135)$$

Moreover, we define the penalty function for the pixel-range constraint as

$$(\forall x\in\mathbb{R}^N) \quad R_2(x) = 2\,\mathrm{d}^2_{[0,x_{\max}]^N}(x). \qquad (136)$$

Finally, the penalty function for problem (131) is

$$(\forall x\in\mathbb{R}^N) \quad R(x) = R_1(x)+R_2(x). \qquad (137)$$

Our MMS approach requires the design of quadratic tangent majorants for the functions F, R_1, and R_2, respectively defined in (130), (134) and (136).

To determine a majorant function for F, we rely on the following proposition extending the result from [43].

Proposition 3 *Let $\rho\colon [0,+\infty) \longrightarrow \mathbb{R}$ be a convex, twice continuously differentiable function with concave derivative ρ'. We denote by $\rho'(0)$ (resp. $\rho''(0)$) the right-hand derivative (resp. second-order derivative) of ρ at 0. Let $\overline{\rho}$ the continuously differentiable extended version of ρ on $\mathbb{R}$, given by*

$$(\forall \upsilon \in \mathbb{R}) \quad \overline{\rho}(\upsilon) = \begin{cases} \rho(\upsilon) & \text{if } \upsilon \geq 0, \\ \rho'(0)\upsilon + \rho(0) & \text{if } \upsilon < 0. \end{cases} \tag{138}$$

Then, $\overline{\rho}$ satisfies the following majoration property at every $\tau \in \mathbb{R}$:

$$(\forall \upsilon \in \mathbb{R}) \quad \overline{\rho}(\upsilon) \leq \overline{\rho}(\tau) + \overline{\rho}'(\tau)(\upsilon - \tau) + \frac{1}{2}\omega(\tau)(\upsilon - \tau)^2, \tag{139}$$

with $\overline{\rho}'$ the derivative of $\overline{\rho}$ and

$$(\forall \tau \in \mathbb{R}) \quad \omega(\tau) = \begin{cases} \frac{\overline{\rho}'(|\tau|) - \overline{\rho}'(0)}{|\tau|} & \text{if } \tau \neq 0, \\ \rho''(0) & \text{if } \tau = 0. \end{cases} \tag{140}$$

A proof of Proposition 3 is given in the appendix. In particular, Proposition 3 holds for ρ in (128) and $\overline{\rho}$ defined in (129). The majorant obtained in Proposition 3 is more accurate than the one based on the descent lemma ([7, Lemma 2.64]) as illustrated in the example of Fig. 5. This behaviour should be beneficial for the practical convergence speed of our method.

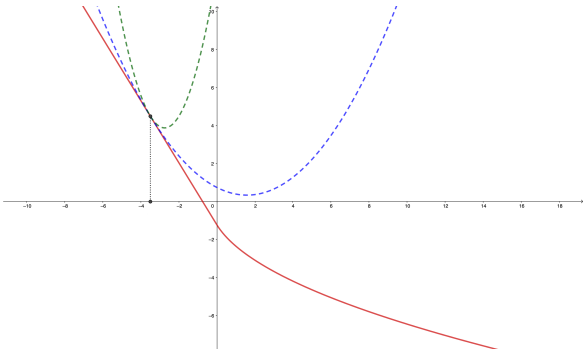


Fig. 5 Comparison of two strategies for majorizing function $\overline{\rho}$ (red) around $\upsilon = -1$. The green curve is the majorant resulting from the descent lemma (i.e., exploiting the Lipschitz constant of $\overline{\rho}'$). The blue curve is the majorant obtained with our Proposition 3.

Using now Propositions 2 and 3, we build a quadratic majorant of F at $x \in \mathbb{R}^N$ characterized by the curvature matrix:

$$(\forall x \in \mathbb{R}^N) \quad A_F(x) = 4H^\top \mathrm{Diag}\left\{ \left((T(y))_i \, \omega((Hx)_i) \right)_{1 \leq i \leq M} \right\} H + \epsilon \mathrm{I}_N, \tag{141}$$

with ϵ a small positive parameter which ensures that the curvature matrix is positive. Quadratic majorants of R_1 and R_2 can be obtained following our Example 2, which yields the respective curvatures:

$$(\forall x \in \mathbb{R}^N) \quad A_{R_1}(x) = 2G^\top G \quad \text{and} \quad A_{R_2}(x) = 4 \cdot \mathrm{I}_N. \tag{142}$$

We illustrate the performance of P-MMS and P-MMS$^{\mathrm{loc}}$ methods. For both algorithms, since the constraint set C defined in (133) is nonempty (it contains the zero image), the convergence conditions derived in Section 5.3 are met. Indeed, Assumptions 1-4 and 6 are fulfilled by functions F and R defined in (137). In addition, as the memory gradient subspace (35) was chosen, Assumption 5 is satisfied. Finally, penalty parameters $(\gamma_j)_{j\in\mathbb{N}}$ in P-MMS and P-MMS$^{\mathrm{loc}}$ were set to

$$(\forall j \in \mathbb{N}) \quad \gamma_j = (0.002j)^{0.6}, \tag{143}$$

and the precision parameters $(\varepsilon_j)_{j\in\mathbb{N}}$ were chosen as

$$(\forall j \in \mathbb{N}) \quad \varepsilon_j = 0.01/(\gamma_j)^{0.15}. \tag{144}$$

These sequences satisfy the requirements of Assumption 7. Assumption 8 also holds since F and R are convex.

6.2.4 Results

We simulate a realistic example using the brain phantom image $\overline{x}$ from the dataset [9] of size $N = 128 \times 128$ pixels and intensity range $[0, 200]$ displayed in Fig. 6 (left)). Measurements are generated using Model (123) where H models a simplified 2D PET geometry with 144 detectors regularly distributed on a circle with radius equals to $64\sqrt{2} + 16$ pixel units, leading to $M = 10296$ pairs of detectors and approximately $4 \cdot 10^5$ Poisson counts. The hyper-parameters were set to $x_{\max} = 200$, $\alpha = 0.9\,\mathrm{TV}(\overline{x})$, and $\epsilon = 10^{-3}$.

We again compare the convergence speed of P-MMS and its local acceleration P-MMS$^{\mathrm{loc}}$ with the primal-dual CV algorithm [32,77], PPXA+ [31], and the improved FISTA [8] scheme from [23] using the method [1] for the inner proximity steps. All the aforementioned algorithms require projection steps onto the $\ell_{1,2}$-ball, so as to handle the TV term. This can be expressed as a function of the projection on the ℓ_1-ball, that we implemented using Condat's algorithm in [33]. Note that a comparison to a second-order algorithm was not possible as the objective function F is not twice differentiable and the TV function is not differentiable.

The reconstructed image is displayed in Fig. 6 (right). One can assess the good quality of reconstruction obtained with the proposed formulation (131), compared

to the use of a simpler least-square approach coupled with a quadratic regularization (Fig. 6 (bottom)). We display in Fig. 7 the evolution of the relative distance between the outer iterate and the limit point x_∞, for each method. Hereagain, one can notice that the proposed method P-MMS$^{\text{loc}}$ outperforms all the other competitors.

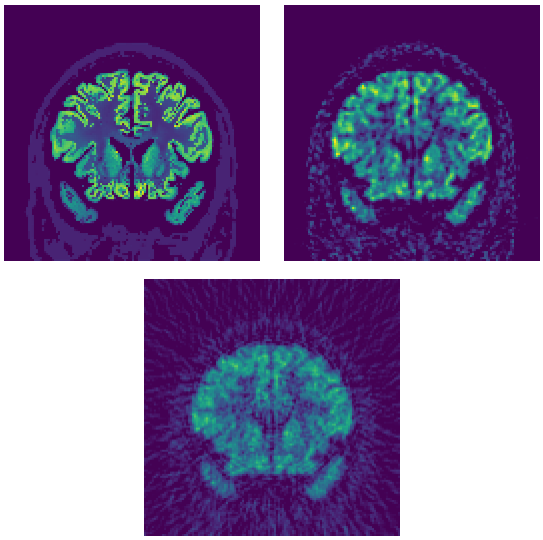


Fig. 6 (top left) Original image ; (top right) Reconstructed image with model (131) ; (bottom) Reconstructed image with a Tikhonov least squares approach.

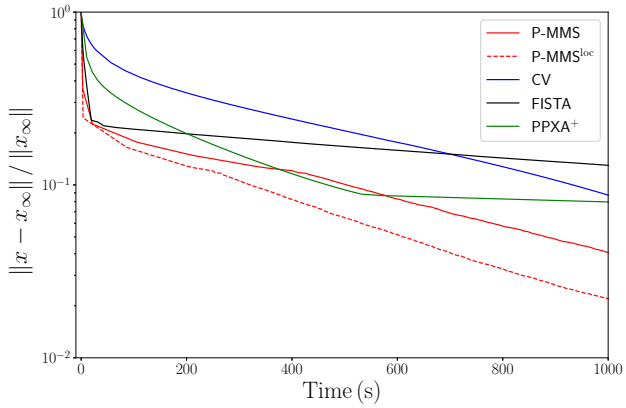


Fig. 7 Comparison of convergence speed on the PET reconstruction example. Evolution of the distance to the optimum versus time.

7 Conclusion

In this paper, we have proposed a new combined subspace-exterior penalty algorithm for solving constrained, differentiable optimization problems. An interesting feature of our approach is the trust-region technique used to accelerate the algorithm. The proposed method can handle a large variety of constraints and was shown to compare favorably with state-of-the-art algorithms in terms of convergence speed on two large-scale image recovery problems. In future work, it could be interesting to relax the convexity assumption (Assumption 8) made in the theoretical analysis of the penalty method, possibly by better relying on the KL property. Another development would consist in establishing a local convergence rate for our algorithm.

Appendix: proof of Proposition 3

Proof. Let $\tau \in \mathbb{R}$. Define, for every $\upsilon \in \mathbb{R}$,

$$\phi(\upsilon) = \overline{\rho}(\upsilon) - \overline{\rho}(\tau) - \overline{\rho}'(\tau)(\upsilon - \tau) - \frac{\omega(\tau)}{2}(\upsilon - \tau)^2. \quad (145)$$

We shall prove that ϕ is nonpositive on $\mathbb{R}$. First, note that our assumptions on ρ imply that ρ' is nondecreasing on $[0, +\infty)$ and ρ'' is positive, nonincreasing on $[0, +\infty)$. We distinguish three cases namely $\tau > 0$, $\tau = 0$, and $\tau < 0$.

(i) Suppose that $\tau > 0$. Differentiating ϕ twice yields the equality

$$(\forall \upsilon \in (0, +\infty)) \quad \phi''(\upsilon) = \rho''(\upsilon) - \frac{\rho'(\tau) - \rho'(0)}{\upsilon} \quad (146)$$

and $\phi''(0^+) = \rho''(0) - (\rho'(\tau) - \rho'(0))/\tau$. Since ρ' is concave on $[0, +\infty)$ and $\tau > 0$, it follows from the tangent inequality that

$$\phi''(0^+) \geq 0 \quad \text{and} \quad \phi''(\tau) \leq 0. \quad (147)$$

In addition, ϕ'' is nonincreasing and continuous on $[0, +\infty)$, thus there exists $\zeta \in [0, \tau]$ such that $\phi''(\zeta) = 0$. Hence, ϕ' is nondecreasing on $[0, \zeta)$ and nonincreasing on $[\zeta, +\infty)$. Since $\phi'(0) = \phi'(\tau) = 0$, it follows that

$$(\forall \upsilon \in [0, \tau)) \quad \phi'(\upsilon) \geq 0, \quad (148)$$

$$(\forall \upsilon \in [\tau, +\infty)) \quad \phi'(\upsilon) \leq 0. \quad (149)$$

Finally, ϕ is nondecreasing on $[0, \tau]$, nonincreasing on $[\upsilon, +\infty)$, and since $\phi(\tau) = 0$,

$$(\forall \upsilon \in [0, +\infty)) \quad \phi(\upsilon) \leq 0. \quad (150)$$

Let us prove the same inequality on $]-\infty, 0[$. We have

$$\forall \upsilon \in (-\infty, 0[, \ \phi''(\upsilon) = -\frac{\rho'(\tau) - \rho'(0)}{\upsilon}. \quad (151)$$

Since ρ' is nondecreasing on $[0,+\infty)$ and $\tau > 0$,

$$\forall \upsilon \in (-\infty, 0),\ \phi''(\upsilon) \leq 0. \tag{152}$$

We deduce that ϕ' is nonincreasing on $(-\infty, 0)$. Moreover, since $\phi'(0) = 0$, we obtain by continuity that $\phi'(\upsilon) \geq 0$ for all $\upsilon \in (-\infty, 0)$. Hence ϕ is nondecreasing on $(-\infty, 0)$ and, according to (150), $\phi(0) \leq 0$. Thus

$$(\forall \upsilon \in (-\infty, 0)) \quad \phi(\upsilon) \leq 0, \tag{153}$$

which concludes the study in the case when $\tau > 0$.

(ii) Suppose that $\tau = 0$. The derivative of ϕ reads

$$(\forall \upsilon \in \mathbb{R}) \quad \phi'(\upsilon) = \overline{\rho}'(\upsilon) - \overline{\rho}'(0) - \rho''(0)\upsilon, \tag{154}$$

thus by concavity of ρ' on $[0,+\infty)$, for every $\upsilon \geq 0$, $\phi'(\upsilon) \leq 0$. Moreover, for every $\upsilon < 0$,

$$\phi'(\upsilon) = -\rho''(0)\upsilon \geq 0. \tag{155}$$

Since $\phi(0) = 0$, we easily conclude that

$$(\forall \upsilon \in \mathbb{R}) \quad \phi(\upsilon) \leq 0. \tag{156}$$

(iii) Suppose that $\tau < 0$. For every $\upsilon \in [0,+\infty)$,

$$\phi''(\upsilon) = \rho''(\upsilon) - \frac{\rho'(-\tau) - \rho'(0)}{-\tau}. \tag{157}$$

Since ρ' is concave on $[0,+\infty)$,

$$\phi''(0^+) \geq 0 \quad \text{and} \quad \phi''(-\tau) \leq 0. \tag{158}$$

As for the case $\tau > 0$, there exists $\zeta \in [0, -\tau]$ such that $\phi''(\zeta) = 0$. Hence ϕ' is nondecreasing on $[0, \zeta[$ and nonincreasing on $[\zeta, +\infty)$. Let us show that $\phi'(\zeta) \leq 0$. Using the fact that $\overline{\rho}'(\tau) = \rho'(0)$ (because of (138) when $\tau < 0$), an immediate calculation yields

$$\begin{aligned}\phi'(\zeta) &= \rho'(\zeta) - \rho'(0) \\ &\quad - (\rho'(-\tau) - \rho'(0)) - \frac{\rho'(-\tau) - \rho'(0)}{-\tau}\zeta.\end{aligned} \tag{159}$$

Since ρ' is nondecreasing and $\rho''(\zeta) = \frac{\rho'(-\tau)-\rho'(0)}{-\tau}$ by definition of ζ, we have

$$\begin{aligned}\phi'(\zeta) &\leq \rho'(\zeta) - \rho'(0) - (\rho'(\zeta) - \rho'(0)) - \rho''(\zeta)\zeta \\ &= -\rho''(\zeta)\zeta \\ &\leq 0.\end{aligned} \tag{160}$$

Therefore

$$(\forall \upsilon \in [0,+\infty)) \quad \phi'(\upsilon) \leq 0, \tag{161}$$

and ϕ in nonincreasing $[0,+\infty)$.

Now, it suffices to prove that $\phi(0) \leq 0$ to deduce the nonpositivity of ϕ on $[0,+\infty)$. We have

$$\begin{aligned}\phi(0) &= \overline{\rho}(0) - \overline{\rho}(\tau) - \overline{\rho}'(\tau)(-\tau) - \frac{\rho'(-\tau) - \rho'(0)}{-2\tau}\tau^2 \\ &= \rho(0) - (\rho'(0)\tau + \rho(0)) + \rho'(0)\tau \\ &\quad + \frac{1}{2}(\rho'(-\tau) - \rho'(0))\,\tau \\ &= \frac{1}{2}(\rho'(-\tau) - \rho'(0))\,\tau \\ &\leq 0,\end{aligned} \tag{162}$$

where the last inequality follows from the fact that ρ' is nondecreasing on $]-\infty, 0]$. Hence

$$(\forall \upsilon \in [0,+\infty)) \quad \phi(\upsilon) \leq \phi(0) \leq 0. \tag{163}$$

To prove a similar inequality on $(-\infty, 0)$, we observe that

$$(\forall \upsilon \in (-\infty, 0)) \quad \phi''(\upsilon) = -\omega(\tau) \leq 0. \tag{164}$$

Thus ϕ' is nonincreasing on $(-\infty, 0)$. Since $\phi'(\tau) = 0$, we deduce that ϕ is nondecreasing on $(-\infty, \tau]$ and nonincreasing on $[\tau, 0)$. It follows that

$$(\forall \upsilon \in (-\infty, 0)) \quad \phi(\upsilon) \leq \phi(\tau) = 0. \tag{165}$$

Finally, gathering inequalities (163) and (165) yields

$$(\forall \upsilon \in \mathbb{R}) \quad \phi(\upsilon) \leq 0, \tag{166}$$

which concludes the proof.

□

References

1. Abboud, F., Chouzenoux, E., Pesquet, J.C., Chenot, J.H., Laborelli, L.: Dual block-coordinate forward-backward algorithm with application to deconvolution and deinterlacing of video sequences. Journal of Mathematical Imaging and Vision **59**(3), 415–431 (2017)
2. Afonso, M.V., Bioucas-Dias, J.M., Figueiredo, M.A.T.: An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. IEEE Transactions on Image Processing **20**(3), 681–695 (2010)
3. Afonso, M.V., Bioucas-Dias, J.M., Figueiredo, M.A.T.: A fast algorithm for the constrained formulation of compressive image reconstruction and other linear inverse problems. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 4034–4037. IEEE (2010)
4. Allain, M., Idier, J., Goussard, Y.: On global and local convergence of half-quadratic algorithms. IEEE Transactions on Image Processing **15**(5), 1130–1142 (2006)
5. Anscombe, F.J.: The transformation of Poisson, binomial and negative-binomial data. Biometrika **35**(3/4), 246–254 (1948)

6. Bahmani, S., Raj, B., Boufounos, P.T.: Greedy sparsity-constrained optimization. Journal of Machine Learning Research **14**(3), 807–841 (2013)
7. Bauschke, H.H., Combettes, P.L., et al.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces, vol. 408. Springer (2011)
8. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences **2**(1), 183–202 (2009)
9. Belzunce, M.A.: High-resolution teterogeneous digital PET [18F]FDG brain phantom based on the BigBrain Atlas (2018). URL `https://doi.org/10.5281/zenodo.1190598`
10. Bertsekas, D.P.: Constrained Optimization and Lagrange Multiplier Methods. Academic press (2014)
11. Bolte, J., Daniilidis, A., Lewis, A.: A nonsmooth Morse–Sard theorem for subanalytic functions. Journal of Mathematical Analysis and Applications **321**(2), 729–740 (2006)
12. Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Mathematical Programming **146**(1-2), 459–494 (2014)
13. Bonettini, S., Zanella, R., Zanni, L.: A scaled gradient projection method for constrained image deblurring. Inverse Problems. An International Journal on the Theory and Practice of Inverse Problems, Inverse Methods and Computerized Inversion of Data **25**(1), 015002 (2008)
14. Bonnans, J.F., Gilbert, J.C., Lemaréchal, C., Sagastizábal, C.A.: Numerical Optimization: Theoretical and Practical Aspects. Springer Science & Business Media (2006)
15. Boyd, S., Parikh, N., Chu, E.: Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. Now Publishers Inc (2011)
16. Branch, M.A., Coleman, T.F., Li, Y.: A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. SIAM Journal on Scientific Computing **21**(1), 1–23 (1999)
17. Byrd, R.H., Gilbert, J.C., Nocedal, J.: A trust region method based on interior point techniques for nonlinear programming. Mathematical Programming **89**(1), 149–185 (2000)
18. Byrd, R.H., Hribar, M.E., Nocedal, J.: An interior point algorithm for large-scale nonlinear programming. SIAM Journal on Optimization **9**(4), 877–900 (1999)
19. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing **16**(5), 1190–1208 (1995)
20. Carrillo, R.E., McEwen, J.D., Wiaux, Y.: Sparsity Averaging Reweighted Analysis (SARA): a novel algorithm for radio-interferometric imaging. Monthly Notices of the Royal Astronomical Society **426**(2), 1223–1234 (2012)
21. Castillo, E.: Quadratic penalty method for intensity-based deformable image registration and 4DCT lung motion recovery. Medical Physics **46**(5), 2194–2203 (2019)
22. Celis, M.R., Dennis, J.E., Tapia, R.A.: A trust region strategy for equality constrained optimization. Tech. rep., Rice University Houston TX Department of Mathematical Sciences (1984)
23. Chambolle, A., Dossal, C.: On the convergence of the iterates of the "Fast Iterative Shrinkage/Thresholding Algorithm". Journal of Optimization Theory and Applications **166**(3), 968–982 (2015)
24. Chan, T.F., Mulet, P.: On the convergence of the lagged diffusivity fixed point method in total variation image restoration. SIAM Journal on Numerical Analysis **36**(2), 354–367 (1999)
25. Cherni, A., Chouzenoux, E., Duval, L., Pesquet, J.C.: SPOQ ℓ_p-over-ℓ_q regularization for sparse signal recovery applied to mass spectrometry. IEEE Transactions on Signal Processing **68**, 6070–6084 (2020)
26. Chouzenoux, E., Idier, J., Moussaoui, S.: A Majorize–Minimize strategy for subspace optimization applied to image restoration. IEEE Transactions on Image Processing **20**(6), 1517–1528 (2010)
27. Chouzenoux, E., Jezierska, A., Pesquet, J.C., Talbot, H.: A Majorize-Minimize subspace approach for $\ell_2 - \ell_0$ image regularization. SIAM Journal on Imaging Sciences **6**, 563–591 (2013)
28. Chouzenoux, E., Pesquet, J.C.: Convergence rate analysis of the Majorize–Minimize subspace algorithm. IEEE Signal Processing Letters **23**(9), 1284–1288 (2016)
29. Coleman, T.F., Li, Y.: An interior trust region approach for nonlinear minimization subject to bounds. SIAM Journal on Optimization **6**(2), 418–445 (1996)
30. Combettes, P.L., Pesquet, J.C.: Image restoration subject to a total variation constraint. IEEE Transactions on Image Processing **13**(9), 1213–1222 (2004)
31. Combettes, P.L., Pesquet, J.C.: Proximal splitting methods in signal processing. In: Fixed-point Algorithms for Inverse Problems in Science and Engineering, pp. 185–212. Springer (2011)
32. Condat, L.: A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. Journal of Optimization Theory and Applications **158**(2), 460–479 (2012)
33. Condat, L.: Fast projection onto the simplex and the ℓ_1-ball. Mathematical Programming **158**(1-2), 575–585 (2015)
34. Conn, A.R., Gould, N.I.M., Toint, P.: A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. SIAM Journal on Numerical Analysis **28**(2), 545–572 (1991)
35. Conn, A.R., Gould, N.I.M., Toint, P.L.: Trust Region Methods. Society for Industrial and Applied Mathematics (2000)
36. Cragg, E.E., Levy, A.V.: Study on a supermemory gradient method for the minimization of functions. Journal of Optimization Theory and Applications **4**(3), 191–205 (1969)
37. Curtis, F.E., Gould, N.I.M., Robinson, D.P., Toint, P.L.: An interior-point trust-funnel algorithm for nonlinear optimization. Mathematical Programming **161**(1-2), 73–134 (2017)
38. Di Pillo, G., Grippo, L.: Exact penalty functions in constrained optimization. SIAM Journal on control and optimization **27**(6), 1333–1360 (1989)
39. Dupe, F.X., Fadili, J.M., Starck, J.L.: A proximal iteration for deconvolving Poisson noisy images using sparse representations. IEEE Transactions on Image Processing **18**(2), 310–321 (2009)
40. Durand, S., Fadili, J., Nikolova, M.: Multiplicative noise removal using ℓ_1 fidelity on frame coefficients. Journal of Mathematical Imaging and Vision **36**(3), 201–226 (2009)
41. Dussault, J.P.: Numerical stability and efficiency of penalty algorithms. SIAM Journal on Numerical Analysis **32**(1), 296–317 (1995)
42. Elad, M., Matalon, B., Zibulevsky, M.: Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization. Applied and Computational Harmonic Analysis **23**(3), 346–367 (2007)
43. Erdogan, H., Fessler, J.A.: Monotonic algorithms for transmission tomography. In: 2002 5th IEEE EMBS International Summer School on Biomedical Imaging, pp. 14–pp. IEEE (2002)

44. Fiacco, A.V., McCormick, G.P.: Nonlinear Programming: Sequential Unconstrained Minimization Techniques. SIAM (1990)
45. Fletcher, R.: A penalty method for nonlinear constraints. In: Numerical Optimization 1984, pp. 26–40. SIAM Publications (1985)
46. Fletcher, R., Reeves, C.M.: Function minimization by conjugate gradients. The Computer Journal **7**(2), 149–154 (1964)
47. Florescu, A., Chouzenoux, E., Pesquet, J.C., Ciuciu, P., Ciochina, S.: A Majorize-Minimize memory gradient method for complex-valued inverse problems. Signal Processing **103**, 285–295 (2014)
48. Foi, A.: Clipped noisy images: heteroskedastic modeling and practical denoising. Signal Processing **89**(12), 2609–2629 (2009)
49. Gould, N.I.M.: On the convergence of a sequential penalty function method for constrained minimization. SIAM Journal on Numerical Analysis **26**(1), 107–128 (1989)
50. Gould, N.I.M., Toint, P.L.: Nonlinear programming without a penalty function or a filter. Mathematical Programming **122**(1), 155–196 (2010)
51. Grapiglia, G.N., Yuan, J., Yuan, Y.X.: A subspace version of the Powell–Yuan trust-region algorithm for equality constrained optimization. Journal of the Operations Research Society of China **1**(4), 425–451 (2013)
52. Harizanov, S., Pesquet, J.C., Steidl, G.: Epigraphical projection for solving least squares Anscombe transformed constrained optimization problems. In: International Conference on Scale Space and Variational Methods in Computer Vision, pp. 125–136. Springer, Berlin, Heidelberg (2013)
53. Ivanov, V.K., Vasin, V.V., Tanana, V.P.: Theory of Linear Ill-Posed Problems and its Applications. De Gruyter (2013)
54. Konnov, I.V.: An approximate penalty method with descent for convex optimization problems. Russian Mathematics **63**(7), 41–55 (2019)
55. Lee, J.H., Jung, Y.M., Yuan, Y.x., Yun, S.: A subspace SQP method for equality constrained optimization. Computational Optimization and Applications. An International Journal **74**(1), 177–194 (2019)
56. Liu, D.C., Nocedal, J.: On the limited memory BFGS method for large scale optimization. Mathematical Programming **45**(1-3), 503–528 (1989)
57. Luenberger, D.G.: Optimization by Vector Space Methods. John Wiley & Sons (1997)
58. Mallat, S.: A Wavelet Tour of Signal Processing. Elsevier (1999)
59. Miele, A., Cantrell, J.W.: Study on a memory gradient method for the minimization of functions. Journal of Optimization Theory and Applications **3**(6), 459–470 (1969)
60. Murtagh, B.A., Saunders, M.A.: Large-scale linearly constrained optimization. Mathematical Programming **14**(1), 41–72 (1978)
61. Musse, O., Heitz, F., Armspach, J.P.: Topology preserving deformable image matching using constrained hierarchical parametric models. IEEE Transactions on Image Processing **10**(7), 1081–1093 (2001)
62. Narkiss, G., Zibulevsky, M.: Sequential subspace optimization method for large-scale unconstrained optimization. Tech. rep., Technion, Israel Institute of Technology (2005)
63. Nikolova, M.: Weakly constrained minimization: application to the estimation of images and signals involving constant regions. Journal of Mathematical Imaging and Vision **21**(2), 155–175 (2004)
64. Nikolova, M., Ng, M.K.: Analysis of half-quadratic minimization methods for signal and image recovery. SIAM Journal on Scientific Computing **27**(3), 937–966 (2005)
65. Nocedal, J., Wright, S.: Numerical Optimization. Springer Science & Business Media (2006)
66. Powell, M.J.D.: Variable metric methods for constrained optimization. In: Mathematical Programming the State of the Art, pp. 288–311. Springer (1983)
67. Pustelnik, N.: Méthodes proximales pour la résolution de problèmes inverses: application à la tomographie par émission de positrons. Ph.D. thesis, Université Paris-Est (2010). URL `https://pastel.archives-ouvertes.fr/tel-00559126`
68. Pustelnik, N., Benazza-Benhayia, A., Zheng, Y., Pesquet, J.C.: Wavelet-based image deconvolution and reconstruction. In: Wiley Encyclopedia of Electrical and Electronics Engineering, pp. 1–34. American Cancer Society (2016)
69. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena **60**(1-4), 259–268 (1992)
70. Setzer, S., Steidl, G., Teuber, T.: Deblurring Poissonian images by split Bregman techniques. J. Vis. Commun. Image Represent **21**(3), 193–199 (2010)
71. Sghaier, M., Chouzenoux, E., Pesquet, J.C., Muller, S.: A novel task-based reconstruction approach for digital breast tomosynthesis. Medical Image Analysis **77**, 102341 (2022)
72. Shanno, D.F., Marsten, R.E.: Conjugate gradient methods for linearly constrained nonlinear programming. In: Mathematical Programming Studies, pp. 149–161. Springer Berlin Heidelberg (1982)
73. Shultz, G.A., Schnabel, R.B., Byrd, R.H.: A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties. SIAM Journal on Numerical Analysis **22**(1), 47–67 (1985)
74. Sun, Q.Y., Wang, C.Y., Shi, Z.J.: Global convergence of a modified gradient projection method for convex constrained problems. Acta Mathematicae Applicatae Sinica. English Series **22**(2), 227–242 (2006)
75. Thai, T.H., Cogranne, R., Retraint, F.: Camera model identification based on the heteroscedastic noise model. IEEE Transactions on Image Processing **23**(1), 250–263 (2013)
76. Toint, P.L.: Global convergence of a class of trust-region methods for nonconvex minimization in Hilbert space. IMA Journal of Numerical Analysis **8**(2), 231–252 (1988)
77. Vũ, B.C.: A splitting algorithm for dual monotone inclusions involving cocoercive operators. Advances in Computational Mathematics **38**(3), 667–681 (2013)
78. Wald, A., Schuster, T.: Sequential subspace optimization for nonlinear inverse problems. Journal of Inverse and Ill-Posed Problems **25**(1), 99–117 (2017)
79. Wang, Y., Ma, S.: A fast subspace method for image deblurring. Applied Mathematics and Computation **215**(6), 2359–2377 (2009)
80. Wang, Z.H., Wen, Z.W., Yuan, Y.X.: A subspace trust region method for large scale unconstrained optimization. In: Numerical Linear Algebra and Optimization, pp. 264–274. Science Press (2004)
81. Wang, Z.H., Yuan, Y.X.: A subspace implementation of quasi-Newton trust region methods for unconstrained optimization. Numerische Mathematik **104**(2), 241–269 (2006)
82. Yu, C., Zhao, J., Wang, Y., Wang, C., Geng, W.: Separation and imaging diffractions by a sparsity-promoting model and subspace trust-region algorithm. Geophysical Journal International **208**(3), 1756–1763 (2017)

83. Yuan, Y.X.: Subspace methods for large scale nonlinear equations and nonlinear least squares. Optimization and Engineering. International Multidisciplinary Journal to Promote Optimization Theory & Applications in Engineering Sciences **10**(2), 207–218 (2008)
84. Yuan, Y.X.: Recent advances in trust region algorithms. Mathematical Programming **151**(1), 249–281 (2015)
85. Zhang, B., Zhu, Z., Li, S.: A modified spectral conjugate gradient projection algorithm for total variation image restoration. Applied Mathematics Letters. An International Journal of Rapid Publication **27**, 26–35 (2014)
86. Zibulevsky, M.: SESOP-TN: Combining sequential subspace optimization with truncated Newton method. Tech. rep., Computer Science Department, Technion (2008)
87. Zibulevsky, M., Elad, M.: $\ell_1 - \ell_2$ optimization in signal and image processing. IEEE Signal Processing Magazine **27**(3), 76–88 (2010)