

Relative-error inertial-relaxed inexact versions of Douglas-Rachford and ADMM splitting algorithms

M. Marques Alves ^{*} Jonathan Eckstein [†] Marina Geremia [‡]
Jefferson G. Melo [§]

Abstract

This paper derives new inexact variants of the Douglas-Rachford splitting method for maximal monotone operators and the alternating direction method of multipliers (ADMM) for convex optimization. The analysis is based on a new inexact version of the proximal point algorithm that includes both an inertial step and overrelaxation. We apply our new inexact ADMM method to LASSO and logistic regression problems and obtain somewhat better computational performance than earlier inexact ADMM methods.

2000 Mathematics Subject Classification: 90C25, 90C30, 47H05.

Key words: Inertial, proximal point algorithm, operator splitting, ADMM, relative error criterion, relaxation.

1 Introduction

This paper develops a sequence of three algorithms, each building on the previous one. The first algorithm is a new variant of the proximal point algorithm [28] for the general, abstract problem $0 \in T(z)$, where T is a set-valued maximal monotone operator on \mathbb{R}^n for which $T^{-1}(0) \neq \emptyset$. Our proposed method is a new inertial variant of the relaxed hybrid proximal

^{*}Departamento de Matemática, Universidade Federal de Santa Catarina, Florianópolis, Brazil, 88040-900 (maicon.alves@ufsc.br). The work of this author was partially supported by CNPq grants no. 405214/2016-2 and 304692/2017-4.

[†]Department of Management Science and Information Systems and RUTCOR, Rutgers Business School Newark and New Brunswick, Piscataway, NJ 08854, USA (jeckstei@business.rutgers.edu). The work of this author was partially supported by National Science Foundation grant CCF-161761 and Air Force Office of Scientific Research grant FA9550-15-1-0251.

[‡]Departamento de Matemática, Universidade Federal de Santa Catarina, Florianópolis, Brazil, 88040-900 (marinageremia@yahoo.com.br).

[§]IME, Universidade Federal de Goiás, Goiânia, Brazil, 74001-970 (jefferson.ufg@gmail.com).

projection (HPP) method introduced in [31]; see also [30]. It lacks the full generality of [31], but introduces a new “inertial” step modification.

Using this first algorithm, we then develop a new inexact variant of the Douglas-Rachford (DR) splitting method for monotone inclusion problems of the form $0 \in A(x) + B(x)$, where $A, B : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ are set-valued maximal monotone operators.

Finally, based on this latter method, we derive a new inexact variant of the alternating direction method of multipliers (ADMM) algorithm for solving convex optimization problems of the form $\min_{x \in \mathbb{R}^n} \{f(x) + g(x)\}$, where $f, g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ are closed proper convex functions. Using the well known LASSO and logistic regression problems as examples, we perform some computational tests on this last algorithm in Section 5 below, finding somewhat better practical performance than earlier proposed inexact ADMM methods from [17, 18].

This path for developing approximate DR and ADMM methods was pioneered in [16], and is also taken in the more recent paper by Eckstein and Yao [18]: in each case, one takes an approximate form of proximal point algorithm (PPA) [28] and uses it to obtain an approximate form of DR splitting, which can then be used to obtain a new “primDR” variant of the ADMM; the iteration complexity of the “primDR” ADMM was later studied in [3]. The main difference between this paper and the development of “primDR” in [18] is in the underlying variant of the PPA. The “primDR” analysis used the hybrid proximal extragradient (HPE) method [29] due to Solodov and Svaiter, whereas here we instead use the new inexact HPP developed in Section 2.

Our general approach resembles that of [18] in that it uses a primal derivation and the “coupling matrix” between f and g in the optimization formulation must be the identity, whereas [16], drawing on early work in [22], uses a dual derivation and allows for more general coupling matrices. Our analysis is also much closer to [18] than that of [17], which uses a primal-dual “Lagrangian splitting” analysis patterned after [19].

Inertial algorithms for convex optimization and monotone inclusions [2] have been a subject of intense research in recent years. They appear in connection with continuous dynamics — see, *e.g.* [2, 7, 8] — accelerated first- and second-order algorithms, and operator splitting methods — see *e.g.* [5, 6, 12, 13, 25] — with good theoretical and practical performance improvements over prior methods. The inertial methods we propose here have the novel property of simultaneously combining inexact iterations, inertia, and relaxation, with the maximum inertial step α and maximum relaxation factor $\bar{\rho}$ being subject to a mutual constraint; see (20) and (21) below. However, the inertial and relaxation parameters may be chosen independently of the relative-error tolerances.

The remainder of this paper is organized as follows: Section 2 presents our inertial-relaxed HPP method (Algorithm 1) and its convergence analysis (Theorems 2.4 and 2.5). Section 3 then uses the HPP method to develop an inexact inertial-relaxed DR method (Algorithm 2), for which convergence is established in Theorem 3.3. Section 4 then uses inertial-relaxed DR method to derive a partially inexact relative-error ADMM method (Algorithm 3). The main result of this section is Theorem 4.4. Section 5 presents numerical experiments on LASSO and logistic regression problems.

2 An inertial-relaxed hybrid proximal projection (HPP) method

We begin by developing a new method for the problem

$$0 \in T(z), \tag{1}$$

where $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is a maximal monotone operator; we assume that this problem has a solution. Our new proposed procedure for this problem, related to the method of [31] but having a new “inertial” step feature, is given below as Algorithm 1.

Algorithm 1. A relative-error inertial-relaxed HPP method for solving (1)

Initialization: Choose $z^0 = z^{-1} \in \mathbb{R}^n$ and $0 \leq \alpha, \sigma < 1$ and $0 < \underline{\rho} < \bar{\rho} < 2$

for $k = 0, 1, \dots$ **do**

Choose $\alpha_k \in [0, \alpha]$ and define

$$w^k = z^k + \alpha_k(z^k - z^{k-1}) \tag{2}$$

Find $(\tilde{z}^k, v^k) \in \mathbb{R}^n \times \mathbb{R}^n$ and $\lambda_k > 0$ such that

$$v^k \in T(\tilde{z}^k), \quad \|\lambda_k v^k + \tilde{z}^k - w^k\|^2 \leq \sigma^2 (\|\tilde{z}^k - w^k\|^2 + \|\lambda_k v^k\|^2) \tag{3}$$

If $v^k = 0$, then **stop**. Otherwise, choose $\rho_k \in [\underline{\rho}, \bar{\rho}]$ and set

$$z^{k+1} = w^k - \rho_k \frac{\langle w^k - \tilde{z}^k, v^k \rangle}{\|v^k\|^2} v^k \tag{4}$$

end for

We make the following remarks concerning this algorithm:

- (i) The extrapolation step in (2) introduces inertial effects — see *e.g.* [1, 2] — controlled by the parameter α_k . The effect of the overrelaxation parameter ρ_k in (4) is similar but not identical, as shown in Figure 1 below. Conditions on $\{\alpha_k\}$, $\alpha \in [0, 1)$ and $\bar{\rho} \in (0, 2)$ that guarantee the convergence of Algorithm 1 are given in Theorem 2.5 — see (20) and (21) and Figure 2 below.
- (ii) If $\alpha = 0$, in which case $\alpha_k \equiv 0$, Algorithm 1 reduces to a special case of the HPP method of [31]; see also [30]. Algorithm 1 is also closely related to the inertial version of the HPP method presented in [1], although that method uses a different relative error criterion.

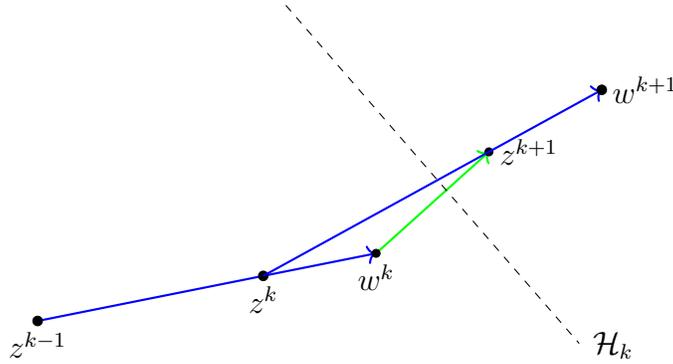


Figure 1: Geometric interpretation of steps (2) and (4) in Algorithm 1. The overrelaxed projection step (4) is orthogonal to the separating hyperplane \mathcal{H}_k , which can differ from the direction between z^{k-1} , z^k , and w^k when $\alpha_k > 0$.

- (iii) At each iteration k , condition (3) is a relative error criterion for the inexact solution of the proximal subproblem $\tilde{z}^k = (I + \lambda_k T)^{-1}(w^k) := J_{\lambda_k T}(w^k)$. If $\sigma = 0$, then this equation must be solved exactly and the pair (\tilde{z}^k, v^k) may be written $(\tilde{z}^k, v^k) = (J_{\lambda_k T}(w^k), \lambda_k^{-1}(w^k - \tilde{z}^k))$. Here, we are primarily concerned with situations in which the calculation of $J_{\lambda_k T}(w^k)$ is relatively difficult and must be approached with an iterative algorithm. In such cases, we use the condition (3) as an acceptance criterion to truncate such an iterative calculation, possibly saving computational effort. We do not specify the exact form of the iterative algorithm used to produce a pair (\tilde{z}^k, v^k) satisfying (3), as it depends on the class of problems to which the algorithm is being applied (and thus the structure of the operator T). See [30, 31] for a related discussion; an abstract formalism of the class of algorithm needed to find a solution to (3) is the “ \mathcal{B} -procedure” described in [18] and also used in Section 3 below.
- (iv) The point z^{k+1} in (4) may be viewed as $z^{k+1} = w^k + \rho_k(P_{\mathcal{H}_k}(w^k) - w^k)$, where $P_{\mathcal{H}_k}$ denotes orthogonal projection onto the hyperplane

$$\mathcal{H}_k := \{z \in \mathbb{R}^n \mid \langle z, v^k \rangle = \langle \tilde{z}^k, v^k \rangle\}, \quad (5)$$

which strictly separates w^k from the solution set $T^{-1}(0)$ of (1). This kind of projective approach to approximate proximal point algorithms was pioneered in [30].

- (v) Algorithm 1 is an inexact variant of the proximal point algorithm (PPA) [28]. In particular, each of its iterations performs an approximate resolvent calculation subject a relative error criterion, and then executes a projection operation in the manner introduced in [30]; see [29, 31] for related work. The main difference from [30] is the inertial step (2).

If $v^k = 0$ in Algorithm 1, then it follows from the inclusion in (3) that \tilde{z}^k is a solution of (1), that is, $0 \in T(\tilde{z}^k)$, so we halt immediately with the solution \tilde{z}^k . For the remainder of

this section, we assume that $v^k \neq 0$ and hence that Algorithm 1 generates an infinite sequence of iterates. The following well-known identity will be useful in the analysis of Algorithm 1:

$$\|(1 - \rho)p + \rho q\|^2 = (1 - \rho)\|p\|^2 + \rho\|q\|^2 - \rho(1 - \rho)\|p - q\|^2 \quad \forall p, q \in \mathbb{R}^n \quad \forall \rho \in \mathbb{R}. \quad (6)$$

Lemma 2.1. [31, Lemma 2] *For each $k \geq 0$, condition (3) implies that*

$$\frac{1 - \sigma^2}{1 + \sqrt{1 - (1 - \sigma^2)^2}} \|\tilde{z}^k - w^k\| \leq \|\lambda_k v^k\| \leq \frac{1 - \sigma^2}{1 - \sqrt{1 - (1 - \sigma^2)^2}} \|\tilde{z}^k - w^k\|. \quad (7)$$

An immediate implication of Lemma 2.1 is that $v^k = 0$ if and only if $\tilde{z}^k = w^k$.

The proof of the following proposition can be found, using different notation, in [31]. For the convenience of the reader, we also present it here.

Proposition 2.2. *Let $\{z^k\}$, $\{\tilde{z}^k\}$ and $\{w^k\}$ be generated by Algorithm 1 and define, for all $k \geq 1$,*

$$s_k = (2 - \bar{\rho}) \max \left\{ \bar{\rho}^{-1} \|z^k - w^{k-1}\|^2, \underline{\rho} (1 - \sigma^2)^2 \|\tilde{z}^{k-1} - w^{k-1}\|^2 \right\}. \quad (8)$$

Then, for any $z^ \in T^{-1}(0)$,*

$$\|z^{k+1} - z^*\|^2 + s_{k+1} \leq \|w^k - z^*\|^2, \quad \forall k \geq 0. \quad (9)$$

Proof. We start by defining \hat{z}^{k+1} as the orthogonal projection of w^k onto the hyperplane $\mathcal{H} := \{z \in \mathbb{R}^n \mid \langle z, v^k \rangle = \langle \tilde{z}^k, v^k \rangle\}$, i.e.,

$$\hat{z}^{k+1} := w^k - \frac{\langle w^k - \tilde{z}^k, v^k \rangle}{\|v^k\|^2} v^k. \quad (10)$$

Next we show that the hyperplane \mathcal{H} strictly separates the current point w^k from the solution set $\Omega := T^{-1}(0) \neq \emptyset$, that is,

$$\langle w^k, v^k \rangle > \langle \tilde{z}^k, v^k \rangle \geq \langle z^*, v^k \rangle \quad \forall z^* \in \Omega. \quad (11)$$

To this end, $0 \in T(z^*)$, $v^k \in T(\tilde{z}^k)$ and the monotonicity of T yield $\langle \tilde{z}^k - z^*, v^k \rangle \geq 0$, which is equivalent to the second inequality in (11). On the other hand, note that from (3) and the Young inequality $2ab \leq a^2 + b^2$ we have

$$\langle w^k - \tilde{z}^k, v^k \rangle \geq \frac{1 - \sigma^2}{2\lambda_k} (\|\tilde{z}^k - w^k\|^2 + \|\lambda_k v^k\|^2) \geq (1 - \sigma^2) \|w^k - \tilde{z}^k\| \|v^k\|,$$

which in turn yields

$$\frac{\langle w^k - \tilde{z}^k, v^k \rangle}{\|v^k\|} \geq (1 - \sigma^2) \|w^k - \tilde{z}^k\| > 0. \quad (12)$$

One consequence of (12) is the first inequality in (11), so (11) must hold.

From (10) and (11), we may infer that \widehat{z}^{k+1} is the projection w^k onto the halfspace $\{z \in \mathbb{R}^n \mid \langle z, v^k \rangle \leq \langle \widehat{z}^k, v^k \rangle\}$, which is a convex set containing z^* . The well-known firm nonexpansiveness properties of the projection operation then imply that

$$\|w^k - z^*\|^2 - \|\widehat{z}^{k+1} - z^*\|^2 \geq \|w^k - \widehat{z}^{k+1}\|^2. \quad (13)$$

Algebraic manipulation of (4) and (10) yields $z^{k+1} - z^* = (1 - \rho_k)(w^k - z^*) + \rho_k(\widehat{z}^{k+1} - z^*)$. Combining this equation with (6) with $(p, q) = (w^k - z^*, \widehat{z}^{k+1} - z^*)$ gives

$$\|z^{k+1} - z^*\|^2 = (1 - \rho_k)\|w^k - z^*\|^2 + \rho_k\|\widehat{z}^{k+1} - z^*\|^2 - \rho_k(1 - \rho_k)\|w^k - \widehat{z}^{k+1}\|^2,$$

which after some rearrangement yields

$$\|w^k - z^*\|^2 - \|z^{k+1} - z^*\|^2 = \rho_k(\|w^k - z^*\|^2 - \|\widehat{z}^{k+1} - z^*\|^2) + \rho_k(1 - \rho_k)\|w^k - \widehat{z}^{k+1}\|^2.$$

Using (13) in the first term on the right-hand side of this identity produces

$$\begin{aligned} \|w^k - z^*\|^2 - \|z^{k+1} - z^*\|^2 &\geq \rho_k\|w^k - \widehat{z}^{k+1}\|^2 + \rho_k(1 - \rho_k)\|w^k - \widehat{z}^{k+1}\|^2 \\ &= (\rho_k + \rho_k(1 - \rho_k))\|w^k - \widehat{z}^{k+1}\|^2 \\ &= \rho_k(2 - \rho_k) \left(\frac{\langle w^k - \widehat{z}^k, v^k \rangle}{\|v^k\|} \right)^2 \quad [\text{by (10)}] \quad (14) \\ &\geq \rho_k(2 - \rho_k)(1 - \sigma^2)^2 \|w^k - \widehat{z}^k\|^2. \quad [\text{by (12)}] \quad (15) \end{aligned}$$

To finish the proof, we observe that (14) and (4) yield

$$\|w^k - z^*\|^2 - \|z^{k+1} - z^*\|^2 \geq \rho_k^{-1}(2 - \rho_k)\|z^{k+1} - w^k\|^2.$$

Combining this inequality with (15), (8) and the bounds $\rho_k \in [\underline{\rho}, \bar{\rho}]$ results in (9). \square

The inequality (17) presented in the following proposition plays a role in the convergence analysis of inertial proximal algorithms — see *e.g.* [2] — similar to that played by Fejér monotonicity in the analysis of standard proximal algorithms.

Proposition 2.3. *Let $\{z^k\}$, $\{w^k\}$ and $\{\alpha_k\}$ be generated by Algorithm 1 and let $\{s_k\}$ be as in (8). Further let $z^* \in T^{-1}(0)$ and define*

$$(\forall k \geq -1) \quad \varphi_k := \|z^k - z^*\|^2 \quad \text{and} \quad (\forall k \geq 0) \quad \delta_k := \alpha_k(1 + \alpha_k)\|z^k - z^{k-1}\|^2. \quad (16)$$

Then, $\varphi_0 = \varphi_{-1}$ and

$$\varphi_{k+1} - \varphi_k + s_{k+1} \leq \alpha_k(\varphi_k - \varphi_{k-1}) + \delta_k \quad \forall k \geq 0, \quad (17)$$

that is, the sequences $\{\varphi_k\}$, $\{s_k\}$, $\{\alpha_k\}$ and $\{\delta_k\}$ satisfy the assumptions of Lemma A.5 below.

Proof. From (2) we obtain $z^k - z^* = (1 + \alpha_k)^{-1}(w^k - z^*) + \alpha_k(1 + \alpha_k)^{-1}(z^{k-1} - z^*)$, which in conjunction with (6) and some algebraic manipulation yields

$$\|w^k - z^*\|^2 = (1 + \alpha_k)\|z^k - z^*\|^2 - \alpha_k\|z^{k-1} - z^*\|^2 + \alpha_k(1 + \alpha_k)\|z^k - z^{k-1}\|^2.$$

Using the above identity and (16) we obtain, for all $k \geq 0$, that

$$\|w^k - z^*\|^2 = (1 + \alpha_k)\varphi_k - \alpha_k\varphi_{k-1} + \delta_k.$$

From (9) in Proposition 2.2 and the definition of φ_k in (16), the above inequality yields (17). Finally, $\varphi_0 = \varphi_{-1}$ follows from the initialization $z^0 = z^{-1}$ and the first definition in (16). \square

The following theorem presents our first result on the asymptotic convergence of Algorithm 1 under the summability assumption (18). Next, Theorem 2.5 gives sufficient conditions (20) and (21) on the inertial and relaxation parameters to assure that (18) is satisfied.

Theorem 2.4 (Convergence of Algorithm 1). *Let $\{z^k\}$, $\{\tilde{z}^k\}$, $\{v^k\}$, $\{\lambda_k\}$ and $\{\alpha_k\}$ be generated by Algorithm 1. If $\inf_k \lambda_k > 0$ and*

$$\sum_{k=0}^{\infty} \alpha_k \|z^k - z^{k-1}\|^2 < +\infty \quad (18)$$

then $\{z^k\}$ converges to a solution of the monotone inclusion problem (1). Moreover, $\{\tilde{z}^k\}$ converges to the same solution and $\{v^k\}$ converges to zero.

Proof. Define $\{s_k\}$ is as in (8). Using Proposition 2.3, (18), that $\alpha_k \leq \alpha < 1$ for all $k \geq 0$, and Lemma A.5, it follows that (i) $\lim_{k \rightarrow \infty} \|z^k - z^*\|$ exist for every $z^* \in \Omega := T^{-1}(0) \neq \emptyset$ and $\sum_{k=1}^{\infty} s_k < +\infty$. So, in particular, $\{z^k\}$ is bounded and (ii) $\lim_{k \rightarrow \infty} s_k = 0$. From the form of (8), that $\lim_{k \rightarrow \infty} s_k = 0$, and the assumption that $\inf \lambda_k > 0$, and Lemma 2.1, we conclude that

$$\lim_{k \rightarrow \infty} \|z^k - w^{k-1}\| = \lim_{k \rightarrow \infty} \|\tilde{z}^k - w^k\| = \lim_{k \rightarrow \infty} \|v^k\| = 0. \quad (19)$$

Now let $z^\infty \in \mathbb{R}^n$ be any cluster point of the bounded sequence $\{z^k\}$. By (19), this point is also a cluster point of $\{w^k\}$ and $\{\tilde{z}^k\}$. Let $\{k_j\}_{j=0}^{\infty}$ be an increasing sequence of indices such that $\tilde{z}^{k_j} \rightarrow z^\infty$. We then have

$$(\forall j \geq 0) \quad v^{k_j} \in T(\tilde{z}^{k_j}), \quad \lim_{j \rightarrow \infty} v^{k_j} = 0 \quad \text{and} \quad \lim_{j \rightarrow \infty} \tilde{z}^{k_j} = z^\infty,$$

which by the standard closure property of maximal monotone operators yields $z^\infty \in \Omega = T^{-1}(0)$. Hence, the desired result on $\{z^k\}$ follows from (i) and Opial's lemma (stated below as Lemma A.4). On the other hand, the convergence of $\{z^k\}$ and (19) yields the remaining results regarding $\{\tilde{z}^k\}$ and $\{v^k\}$. \square

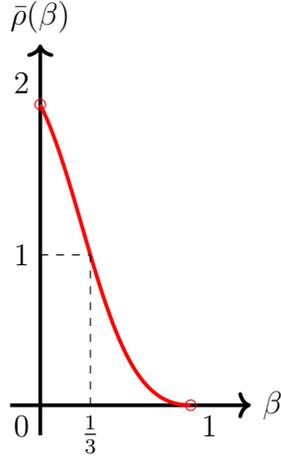


Figure 2: The relaxation parameter upper bound $\bar{\rho}(\beta)$ from (21) as a function of inertial step upper bound $\beta > 0$ of (20). Note that $\bar{\rho}(1/3) = 1$, while $\bar{\rho}(\beta) > 1$ whenever $\beta < 1/3$.

Theorem 2.5 (Convergence of Algorithm 1). *Let $\{z^k\}$, $\{\alpha_k\}$ and $\{\lambda_k\}$ be generated by Algorithm 1. Assume that $\alpha \in [0, 1)$, $\bar{\rho} \in (0, 2)$ and $\{\alpha_k\}$ satisfy the following (for some $\beta > 0$):*

$$0 \leq \alpha_k \leq \alpha_{k+1} \leq \alpha < \beta < 1 \quad \forall k \geq 0 \quad (20)$$

and

$$\bar{\rho} = \bar{\rho}(\beta) := \frac{2(\beta - 1)^2}{2(\beta - 1)^2 + 3\beta - 1}. \quad (21)$$

Then,

$$\sum_{k=1}^{\infty} \|z^k - z^{k-1}\|^2 < +\infty. \quad (22)$$

As a consequence, it follows that under the assumptions (20) and (21) the sequence $\{z^k\}$ generated by Algorithm 1 converges to a solution of the monotone inclusion problem (1) whenever $\inf \lambda_k > 0$. Moreover, under the above assumptions, $\{\tilde{z}^k\}$ converges to the same solution and $\{v^k\}$ converges to zero.

Proof. Using (2), the Cauchy-Schwarz inequality and the Young inequality $2ab \leq a^2 + b^2$ with $a := \|z^{k+1} - z^k\|$ and $b := \|z^k - z^{k-1}\|$ we find

$$\begin{aligned} \|z^{k+1} - w^k\|^2 &= \|z^{k+1} - z^k\|^2 + \alpha_k^2 \|z^k - z^{k-1}\|^2 - 2\alpha_k \langle z^{k+1} - z^k, z^k - z^{k-1} \rangle \\ &\geq \|z^{k+1} - z^k\|^2 + \alpha_k^2 \|z^k - z^{k-1}\|^2 - \alpha_k (2\|z^{k+1} - z^k\| \|z^k - z^{k-1}\|) \\ &\geq (1 - \alpha_k) \|z^{k+1} - z^k\|^2 - \alpha_k (1 - \alpha_k) \|z^k - z^{k-1}\|^2. \end{aligned} \quad (23)$$

Starting with a rearrangement of (17), we then obtain

$$\begin{aligned}
\varphi_{k+1} - \varphi_k - \alpha_k(\varphi_k - \varphi_{k-1}) &\leq \delta_k - s_{k+1} \\
&\leq \alpha_k(1 + \alpha_k)\|z^k - z^{k-1}\|^2 - (2 - \bar{\rho})\bar{\rho}^{-1}\|z^{k+1} - w^k\|^2 && \text{[by (8) and (16)]} \\
&\leq \alpha_k(1 + \alpha_k)\|z^k - z^{k-1}\|^2 \\
&\quad - (2 - \bar{\rho})\bar{\rho}^{-1} \left[(1 - \alpha_k)\|z^{k+1} - z^k\|^2 - \alpha_k(1 - \alpha_k)\|z^k - z^{k-1}\|^2 \right] && \text{[by (23)]} \\
&= -(2 - \bar{\rho})\bar{\rho}^{-1}(1 - \alpha_k)\|z^{k+1} - z^k\|^2 + [\alpha_k(1 + \alpha_k) + (2 - \bar{\rho})\bar{\rho}^{-1}\alpha_k(1 - \alpha_k)]\|z^k - z^{k-1}\|^2, \\
&= -(2 - \bar{\rho})\bar{\rho}^{-1}(1 - \alpha_k)\|z^{k+1} - z^k\|^2 + \gamma_k\|z^k - z^{k-1}\|^2, && (24)
\end{aligned}$$

where

$$\gamma_k := -2(\bar{\rho}^{-1} - 1)\alpha_k^2 + 2\bar{\rho}^{-1}\alpha_k \quad \forall k \geq 0. \quad (25)$$

Some elementary algebraic manipulations of (24) then yield

$$\varphi_{k+1} - \varphi_k - \alpha_k(\varphi_k - \varphi_{k-1}) - \gamma_k\|z^k - z^{k-1}\|^2 \leq -(2\bar{\rho}^{-1} - 1)(1 - \alpha_k)\|z^{k+1} - z^k\|^2 \quad \forall k \geq 0. \quad (26)$$

Define now the scalar function:

$$q(\nu) := 2(\bar{\rho}^{-1} - 1)\nu^2 - (4\bar{\rho}^{-1} - 1)\nu + 2\bar{\rho}^{-1} - 1, \quad (27)$$

and

$$\mu_0 := (1 - \alpha_0)\varphi_0 \geq 0, \quad \mu_k := \varphi_k - \alpha_{k-1}\varphi_{k-1} + \gamma_k\|z^k - z^{k-1}\|^2 \quad \forall k \geq 1, \quad (28)$$

where φ_k is as in (16). Using (26)-(28) and the assumption that $\{\alpha_k\}$ is nondecreasing — see (20) — we obtain, for all $k \geq 0$,

$$\begin{aligned}
\mu_{k+1} - \mu_k &\leq [\varphi_{k+1} - \varphi_k - \alpha_k(\varphi_k - \varphi_{k-1}) - \gamma_k\|z^k - z^{k-1}\|^2] + \gamma_{k+1}\|z^{k+1} - z^k\|^2 \\
&\leq [\gamma_{k+1} - (2\bar{\rho}^{-1} - 1)(1 - \alpha_{k+1})]\|z^{k+1} - z^k\|^2 \\
&= -[2(\bar{\rho}^{-1} - 1)\alpha_{k+1}^2 - (4\bar{\rho}^{-1} - 1)\alpha_{k+1} + 2\bar{\rho}^{-1} - 1]\|z^{k+1} - z^k\|^2 \\
&= -q(\alpha_{k+1})\|z^{k+1} - z^k\|^2. && (29)
\end{aligned}$$

We will now show that $q(\alpha_{k+1})$ admits a uniform positive lower bound. To this end, note first that from (21) and Lemma A.2 below that we have

$$\beta = \frac{2(2 - \bar{\rho})}{4 - \bar{\rho} + \sqrt{\bar{\rho}(16 - 7\bar{\rho})}}.$$

Using the latter identity, (27), and Lemma A.3 below with $a = 2(\bar{\rho}^{-1} - 1)$, $b = 4\bar{\rho}^{-1} - 1$, and $c = 2\bar{\rho}^{-1} - 1$, we conclude that $q(\cdot)$ is decreasing in $[0, \beta]$ and $\beta > 0$ is a root of $q(\cdot)$. Thus, in view of (20), we conclude that

$$q(\alpha_{k+1}) \geq q(\alpha) > q(\beta) = 0, \quad (30)$$

which gives the desired uniform positive lower bound on $q(\alpha_{k+1})$.

Using (29) and (30) we find

$$\|z^{k+1} - z^k\|^2 \leq \frac{1}{q(\alpha)}(\mu_k - \mu_{k+1}), \quad \forall k \geq 0, \quad (31)$$

which, in turn, combined with (20) and the definition of μ_k in (28), gives

$$\begin{aligned} \sum_{j=0}^k \|z^{j+1} - z^j\|^2 &\leq \frac{1}{q(\alpha)}(\mu_0 - \mu_{k+1}), \\ &\leq \frac{1}{q(\alpha)}(\mu_0 + \alpha\varphi_k) \quad \forall k \geq 0. \end{aligned} \quad (32)$$

Note now that (31), (20) and (28) also yield

$$\begin{aligned} \mu_0 \geq \dots \geq \mu_{k+1} &= \varphi_{k+1} - \alpha_k \varphi_k + \gamma_{k+1} \|z^{k+1} - z^k\|^2 \\ &\geq \varphi_{k+1} - \alpha \varphi_k, \quad \forall k \geq 0, \end{aligned}$$

and so,

$$\varphi_{k+1} \leq \alpha^{k+1} \varphi_0 + \frac{\mu_0}{1-\alpha} \leq \varphi_0 + \frac{\mu_0}{1-\alpha} \quad \forall k \geq -1. \quad (33)$$

Hence, (22) follows directly from (32) and (33). On the other hand, the second statement of the theorem follows from (22) and Theorem 2.4 (recall that $\alpha_k \leq \alpha < 1$ for all $k \geq 0$). \square

We close this section with a few further remarks about the analysis of Algorithm 1:

- (i) Conditions (20) and (21) on $\{\alpha_k\}$, α and $\bar{\rho}$ guarantee that the summability condition (18) is satisfied, thus guaranteeing the convergence of Algorithm 1. Similar conditions were also recently proposed and studied in [4, 6]. Since Algorithm 1 is the basis of the DR and ADMM methods developed in the next two sections, conditions (20) and (21) will also play an important role in their convergence analyses.
- (ii) If we set $\beta = 1/3$ in (20), then it follows immediately from (21) that $\bar{\rho} = 1$. On the other hand, we have $\bar{\rho} > 1$ in (21) whenever $\beta < 1/3$ (see also Figure 2). Setting $\beta = 1/3$ in (20) corresponds to the standard strategy in the literature of inertial proximal algorithms; see *e.g.* [2, 12].

3 A partially inexact inertial-relaxed Douglas-Rachford (DR) algorithm

Consider the monotone inclusion problem of finding $z \in \mathbb{R}^n$ such that

$$0 \in A(z) + B(z) \quad (34)$$

where A and B are (set-valued) maximal monotone operators on \mathbb{R}^n for which the solution set $(A + B)^{-1}(0)$ of (34) is nonempty.

A popular operator splitting algorithms for finding approximate solutions to (34) is the Douglas-Rachford (DR) algorithm [15, 24, 16]:

$$z^{k+1} = J_{\gamma A}(2J_{\gamma B}(z^k) - z^k) + z^k - J_{\gamma B}(z^k) \quad \forall k \geq 0, \quad (35)$$

where $\gamma > 0$ is a scaling parameter, z^k is the current iterate and $J_{\gamma A} = (\gamma A + I)^{-1}$ and $J_{\gamma B} = (\gamma B + I)^{-1}$ are the resolvent operators of A and B , respectively. The DR algorithm (35) is a splitting algorithm for solving the (structured) inclusion (34) in the sense that the resolvents $J_{\gamma A}$ and $J_{\gamma B}$ are employed separately, but the resolvent $J_{\gamma(A+B)}$ of $A + B$ is not. Such methods may be useful in situation in which the values of $J_{\gamma A}$ and $J_{\gamma B}$ are relatively easy to evaluate in comparison to those of $J_{\gamma(A+B)}$.

This section will develop an inexact version of the DR algorithm (35) for the situation in which the resolvent of one of the operators, say B , is relatively hard, but evaluating $J_{\gamma A}$ is a simple calculation. To this end, we consider the following equivalent formulation of (35) (see, e.g., [16]): given some $r^k, b^k \in \mathbb{R}^n$,

$$\text{Find } (s^{k+1}, b^{k+1}) \in B \text{ such that } s^{k+1} + \gamma b^{k+1} = r^k + \gamma b^k; \quad (36)$$

$$\text{Find } (r^{k+1}, a^{k+1}) \in A \text{ such that } r^{k+1} + \gamma a^{k+1} = s^{k+1} - \gamma b^{k+1}. \quad (37)$$

In this case, $z^k = r^k + \gamma b^k$. Since the resolvent $J_{\gamma A}$ of A is assumed to be easily computable, the pair (r^{k+1}, a^{k+1}) in (37) is explicitly given by

$$r^{k+1} = J_{\gamma A}(s^{k+1} - \gamma b^{k+1}) \quad a^{k+1} = \gamma^{-1}(s^{k+1} - r^{k+1}) - b^{k+1}.$$

For B , we by contrast suppose that exact computation of the pair (s^{k+1}, b^{k+1}) satisfying (36) requires a relatively time-consuming iterative process, which we model immediately below by the notion of a \mathcal{B} -procedure as introduced in [18]. We first remark that (36) can be posed in the more general framework of solving monotone inclusion problems of the form

$$0 \in s + \gamma B(s) - (r + \gamma b), \quad (38)$$

where $r, b \in \mathbb{R}^n$ and $\gamma > 0$.

Definition 3.1 (\mathcal{B} -procedure for solving (38)). *A \mathcal{B} -procedure for (approximately) solving any instance of (38) is a mapping $\mathcal{B} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{N}^* \rightarrow \mathbb{R}^n \times \mathbb{R}^n$ such that if one lets $(s^\ell, b^\ell) = \mathcal{B}(r, b, \gamma, \bar{s}, \bar{b}, \ell)$ for all $\ell \in \mathbb{N}^*$ and any given $r, b, \bar{s}, \bar{b} \in \mathbb{R}^n$ and $\gamma > 0$, then $b^\ell \in B(s^\ell)$, for all $\ell \in \mathbb{N}^*$, the sequence $\{(s^\ell, b^\ell)\}$ is convergent, and $s^\ell + \gamma b^\ell \rightarrow r + \gamma b$.*

Following [18], the intuitive meaning of $(s^\ell, b^\ell) = \mathcal{B}(r, b, \gamma, \bar{s}, \bar{b}, \ell)$ is that (s^ℓ, b^ℓ) is the ℓ^{th} trial approximation generated by some iterative procedure for solving (38), starting from some initial guess $(\bar{s}, \bar{b}) \in \mathbb{R}^n \times \mathbb{R}^n$. We refer the interested reader to [18, Section 5] for a more detailed discussion and interpretation on the \mathcal{B} -procedure concept.

We make the following standing assumption:

Assumption 1. *There exists a \mathcal{B} -procedure (according to Definition 3.1) for approximately solving any instance of (38).*

We now combine the hypothesized \mathcal{B} -procedure with an acceptance criterion for the approximate solution of (36). We will follow the general approach of [18], which is to exploit the connection between the DR algorithm (36)-(37) and the proximal point algorithm as established in [16]. Specifically, the DR algorithm (36), (37) is a special instance of the PP algorithm in the sense that,

$$r^{k+1} + \gamma b^{k+1} = (S_{\gamma,A,B} + I)^{-1}(r^k + \gamma b^k) \quad \forall k \geq 0$$

where the “splitting” operator $S_{\gamma,A,B}$ is defined as [16]

$$S_{\gamma,A,B} := \{(r + \gamma b, s - r) \in \mathbb{R}^n \times \mathbb{R}^n \mid b \in B(s), a \in A(r), \gamma a + r = s - \gamma b\}. \quad (47)$$

The operator defined in (47) is maximal monotone and

$$(A + B)^{-1}(0) = J_{\gamma B}(S_{\gamma,A,B}^{-1}(0)), \quad (48)$$

which, in particular, gives that any solution $z^* \in \mathbb{R}^n$ of the monotone inclusion problem (1) with $T := S_{\gamma,A,B}$, namely

$$0 \in S_{\gamma,A,B}(z) \quad (49)$$

yields a solution $x^* := J_{\gamma B}(z^*)$ of (34).

Here, we follow a similar derivation to [18], but use Algorithm 1 of Section 2 to (49) in place of the HPE method of [29]. The result is an inertial-relaxed inexact relative-error DR algorithm for solving (34). We should emphasize that even $\alpha_k \equiv 0$ (there is no inertial step) and $\rho_k \equiv 1$ (no overrelaxation), the resulting algorithm differs from that of [18]. This difference arises because the underlying “convergence engine” of Algorithm 1 is a form of hybrid proximal-projection (HPP) algorithm, whereas [18] used an HPE algorithm in the equivalent role, using an extragradient step instead of projection.

The proposed algorithm for solving (34) is shown as Algorithm 2. We should mention that a different inexact DR splitting algorithm in which relative errors are allowed in both (40) and (42) was recently proposed and studied in [32], but without computational testing. The following proposition shows that Algorithm 2 is indeed a special instance of Algorithm 1 for solving (1) with $T := S_{\gamma,A,B}$.

Proposition 3.2. *Consider the sequences evolved by Algorithm 2 and for each $k \geq 0$ let $\ell(k)$ denote the value of ℓ for which (43) is satisfied. For each $k \geq 0$, define, with γ as in Algorithm 2,*

$$\begin{aligned} z^k &:= r^k + \gamma b^k & w^k &:= \hat{r}^k + \gamma \hat{b}^k \\ \tilde{z}^k &:= r^{k,\ell(k)} + \gamma b^{k,\ell(k)} & v^k &:= s^{k,\ell(k)} - r^{k,\ell(k)}. \end{aligned} \quad (50)$$

Then these latter sequences satisfy the conditions (2)-(4) of Algorithm 1 with $\lambda_k \equiv 1$ and $T = S_{\gamma,A,B}$.

Algorithm 2. A partially inexact inertial–relaxed Douglas–Rachford splitting algorithm for solving (34)

Choose $\gamma > 0$, $0 \leq \alpha, \sigma < 1$ and $0 < \underline{\rho} < \bar{\rho} < 2$. Initialize $(s^0, b^0, r^0) = (s^{-1}, b^{-1}, r^{-1}) \in (\mathbb{R}^n)^3$.

for $k = 0, 1, 2, \dots$ **do**

Choose $\alpha_k \in [0, \alpha]$ and define

$$(\hat{s}^k, \hat{b}^k, \hat{r}^k) = (s^k, b^k, r^k) + \alpha_k [(s^k, b^k, r^k) - (s^{k-1}, b^{k-1}, r^{k-1})] \quad (39)$$

repeat {**for** $\ell = 1, 2, \dots$ }

Improve the solution to

$$b^{k,\ell} \in B(s^{k,\ell}), \quad s^{k,\ell} + \gamma b^{k,\ell} \approx \hat{r}^k + \gamma \hat{b}^k \quad (40)$$

by setting

$$(s^{k,\ell}, b^{k,\ell}) = \mathcal{B}(\hat{r}^k, \hat{b}^k, \gamma, \hat{s}^k, \hat{b}^k, \ell) \quad (41)$$

(thus incrementally executing a step of the \mathcal{B} –procedure)

Exactly find $(r^{k,\ell}, a^{k,\ell}) \in \mathbb{R}^n \times \mathbb{R}^n$ such that

$$a^{k,\ell} \in A(r^{k,\ell}), \quad r^{k,\ell} + \gamma a^{k,\ell} = s^{k,\ell} - \gamma b^{k,\ell} \quad (42)$$

until

$$\|s^{k,\ell} + \gamma b^{k,\ell} - (\hat{r}^k + \gamma \hat{b}^k)\|^2 \leq \sigma^2 \left(\|r^{k,\ell} + \gamma b^{k,\ell} - (\hat{r}^k + \gamma \hat{b}^k)\|^2 + \|s^{k,\ell} - r^{k,\ell}\|^2 \right) \quad (43)$$

if $s^{k,\ell} = r^{k,\ell}$, then **stop**

otherwise, choose $\rho_k \in [\underline{\rho}, \bar{\rho}]$ and set

$$s^{k+1} = s^{k,\ell}, \quad r^{k+1} = r^{k,\ell} \quad (44)$$

$$\theta_{k+1} = \frac{\langle (\hat{r}^k - r^{k,\ell}) + \gamma(\hat{b}^k - b^{k,\ell}), s^{k,\ell} - r^{k,\ell} \rangle}{\|s^{k,\ell} - r^{k,\ell}\|^2} \quad (45)$$

$$b^{k+1} = \hat{b}^k - \gamma^{-1} [(1 - \rho_k \theta_{k+1})r^{k+1} + \rho_k \theta_{k+1}s^{k+1} - \hat{r}^k] \quad (46)$$

end for

Proof. Fix any $k \geq 0$. From (39) and the definitions of z^k and w^k in (50) we have

$$\begin{aligned} w^k &= \hat{r}^k + \gamma \hat{b}^k = r^k + \gamma b^k + \alpha_k [r^k + \gamma b^k - (r^{k-1} + \gamma b^{k-1})] \\ &= z^k + \alpha_k (z^k - z^{k-1}), \end{aligned}$$

which is exactly (2). Now note that the inclusion in (3) follows from the fact that $T := S_{\gamma, A, B}$, (47), (42), $b^{k, \ell(k)} \in B(s^{k, \ell(k)})$ from (40), and the definitions of v^k and \tilde{z}^k in (50).

Further, (50) and (43) yield

$$\begin{aligned} \|v^k + \tilde{z}^k - w^k\|^2 &= \|s^{k, \ell(k)} + \gamma b^{k, \ell(k)} - (\hat{r}^k + \gamma \hat{b}^k)\|^2 \\ &\leq \sigma^2 \left(\|r^{k, \ell(k)} + \gamma b^{k, \ell(k)} - (\hat{r}^k + \gamma \hat{b}^k)\|^2 + \|s^{k, \ell} - r^{k, \ell}\|^2 \right) \\ &= \sigma^2 (\|\tilde{z}^k - w^k\|^2 + \|v^k\|^2), \end{aligned}$$

which is exactly the inequality in (3) with $\lambda_k = 1$. Finally,

$$\begin{aligned} z^{k+1} &= r^{k+1} + \gamma b^{k+1} && \text{[by (50)]} \\ &= r^{k, \ell(k)} + \gamma \left(\hat{b}^k - \frac{1}{\gamma} [(1 - \rho_k \theta_{k+1}) r^{k, \ell(k)} + \rho_k \theta_{k+1} s^{k, \ell(k)} - \hat{r}^k] \right) && \text{[by (46)]} \\ &= \hat{r}^k + \gamma \hat{b}^k + \rho_k \theta_{k+1} (r^{k, \ell(k)} - s^{k, \ell(k)}) \\ &= w^k + \rho_k \theta_{k+1} v^k && \text{[by (50)]} \\ &= w^k - \rho_k \frac{\langle w^k - \tilde{z}^k, v^k \rangle}{\|v^k\|^2} v^k, && \text{[by (45) and (50)]} \end{aligned}$$

which establishes (4) and thus completes the proof of the proposition. \square

The following theorem states the asymptotic convergence properties of Algorithm 2, which are essentially direct consequences of Proposition 3.2 and Theorem 2.5.

Theorem 3.3 (Convergence of Algorithm 2). *Consider the sequences evolved by Algorithm 2 with the parameters $\alpha \in [0, 1)$, $\bar{\rho} \in (0, 2)$ and $\{\alpha_k\}$ satisfying the conditions (20) and (21) of Theorem 2.5. Then*

- (a) *If the outer loop (over k) executes an infinite number of times, with each inner loop (over ℓ) terminating in a finite number of iterations $\ell = \ell(k)$, then $\{s^k\}$ and $\{r^k\}$ both converge to some solution $x^* \in \mathbb{R}^n$ of (34), and $\{b^{k, \ell(k)}\}$ and $\{b^k\}$ both converge to some $b^* \in B(x^*)$, with $\{a^{k, \ell(k)}\}$ converging to $-b^* \in A(x^*)$.*
- (b) *If the outer loop executes only a finite number of times, ending with $k = \bar{k}$, with the last invocation of the inner loop executing an infinite number of times, then $\{s^{\bar{k}, \ell}\}_{\ell=1}^{\infty}$ and $\{r^{\bar{k}, \ell}\}_{\ell=1}^{\infty}$ both converge to some solution $x^* \in \mathbb{R}^n$ of (34), and $\{b^{\bar{k}, \ell}\}_{\ell=1}^{\infty}$ converges to some $b^* \in B(x^*)$, with $\{a^{\bar{k}, \ell}\}_{\ell=1}^{\infty}$ converging to $-b^* \in A(x^*)$.*
- (c) *If Algorithm 2 stops with $s^{k, \ell} = r^{k, \ell}$, then $z^* := s^{k, \ell} = r^{k, \ell}$ is a solution of (34).*

Proof. (a) For each $k \geq 0$, again let $\ell = \ell(k)$ be the index of inner iteration that first meets the inner-loop termination condition. Using Proposition 3.2, (44), the descriptions of algorithms 1 and 2, and Theorem 2.5, we conclude that there exists $z^* \in \mathbb{R}^n$ such that $0 \in S_{\gamma,A,B}(z^*)$ and

$$z^k = r^k + \gamma b^k \rightarrow z^* \quad \tilde{z}^{k-1} = r^k + \gamma b^{(k-1),\ell(k-1)} \rightarrow z^* \quad v^{k-1} = s^k - r^k \rightarrow 0. \quad (51)$$

From $0 \in S_{\gamma,A,B}(z^*)$ and (48) we obtain that $x^* := J_{\gamma B}(z^*)$ is a solution of (34). Moreover, it follows from (51), the inclusion in (40), (44), and the continuity of $J_{\gamma B}$ that

$$s^k + \gamma b^{(k-1),\ell} = v^{k-1} + \tilde{z}^{(k-1)} \rightarrow 0 + z^* = z^* \quad s^k = J_{\gamma B}(s^k + \gamma b^{(k-1),\ell}) \rightarrow J_{\gamma B}(z^*) = x^*. \quad (52)$$

We also have $r^k \rightarrow x^*$ since, from (51), $s^k - r^k \rightarrow 0$. Altogether, we have that x^* is a solution of (34) and $\{s^k\}$ and $\{r^k\}$ both converge to x^* . From (52) we now have

$$b^{k,\ell(k)} = \gamma^{-1}(s^{k+1} + \gamma b^{k,\ell(k)} - s^{k+1}) \rightarrow \gamma^{-1}(z^* - x^*) := b^*. \quad (53)$$

From $x^* = J_{\gamma B}(z^*)$ we then obtain $b^* \in B(x^*)$. On the other hand, using the equation in (42), (44), (51) and (53) we find

$$a^{k,\ell(k)} = \gamma^{-1}(s^{k+1} - r^{k+1}) - b^{k,\ell(k)} \rightarrow 0 - b^* = -b^*.$$

Using the above convergence result, that $r^{k,\ell(k)} = r^{k+1} \rightarrow x^*$, the inclusion in (42), and Lemma A.1, we obtain that $-b^* \in A(x^*)$. Finally, $b^k = \gamma^{-1}(z^k - r^k) \rightarrow \gamma^{-1}(z^* - r^*) = b^*$.

(b) First note that using (41) we obtain $(s^{\bar{k},\ell}, b^{\bar{k},\ell}) = \mathcal{B}(\hat{r}^{\bar{k}}, \hat{b}^{\bar{k}}, \gamma, \hat{s}^{\bar{k}}, \hat{b}^{\bar{k}}, \ell)$, which in view of Definition 3.1 yields $(s^{\bar{k},\ell}, b^{\bar{k},\ell}) \in B$, for all $\ell \geq 1$, $s^{\bar{k},\ell} + \gamma b^{\bar{k},\ell} \rightarrow \hat{r}^{\bar{k}} + \gamma \hat{b}^{\bar{k}}$, $s^{\bar{k},\ell} \rightarrow x^*$, and $b^{\bar{k},\ell} \rightarrow b^*$, for some $x^*, b^* \in \mathbb{R}^n$. Combining limits, we obtain that $\hat{r}^{\bar{k}} + \gamma \hat{b}^{\bar{k}} = x^* + \gamma b^*$. From Lemma A.1, we also have $b^* \in B(x^*)$. Now combining the limits with (42) and the continuity of $J_{\gamma A}$, we also find

$$r^{\bar{k},\ell} = J_{\gamma A}(s^{\bar{k},\ell} - \gamma b^{\bar{k},\ell}) \rightarrow J_{\gamma A}(x^* - \gamma b^*) =: r^*$$

and so

$$a^{\bar{k},\ell} = \gamma^{-1} \left(s^{\bar{k},\ell} - r^{\bar{k},\ell} \right) - b^{\bar{k},\ell} \rightarrow \gamma^{-1}(x^* - r^*) - b^* =: a^*. \quad (54)$$

From the inclusion in (42) and (again) Lemma A.1 we obtain that $a^* \in A(r^*)$. On the other hand, using (43) and the hypothesis that the inner loop executes an infinite number of times at iteration $k = \bar{k}$, we obtain, for all $\ell \geq 1$, that

$$\|s^{\bar{k},\ell} + \gamma b^{\bar{k},\ell} - (\hat{r}^{\bar{k}} + \gamma \hat{b}^{\bar{k}})\|^2 > \sigma^2 \left(\|r^{\bar{k},\ell} + \gamma b^{\bar{k},\ell} - (\hat{r}^{\bar{k}} + \gamma \hat{b}^{\bar{k}})\|^2 + \|s^{\bar{k},\ell} - r^{\bar{k},\ell}\|^2 \right). \quad (55)$$

Since the left-hand side of the above inequality converges to zero and the right-hand side is nonnegative, the right-hand side also converges to zero and in particular $s^{\bar{k},\ell} - r^{\bar{k},\ell} \rightarrow 0$. Since $s^{\bar{k},\ell} \rightarrow x^*$ and $r^{\bar{k},\ell} \rightarrow r^*$, we conclude that $x^* = r^*$ and, hence, from (54), that $a^* = -b^*$.

(c) If $s^{k,\ell} = r^{k,\ell} =: z^*$, then it follows from the inclusion in (40) and (42) that $0 = \gamma^{-1}(s^{k,\ell} - r^{k,\ell}) = a^{k,\ell} + b^{k,\ell} \in A(r^{k,\ell}) + B(s^{k,\ell}) = (A + B)(z^*)$. \square

4 A partially inexact relative-error inertial-relaxed ADMM

We now consider the convex optimization problem

$$\min_{z \in \mathbb{R}^n} \{f(x) + g(x)\} \quad (56)$$

where $f, g : \mathbb{R}^n \rightarrow (-\infty, \infty]$ are proper, convex and lower semicontinuous functions for which $(\partial f + \partial g)^{-1}(0) \neq \emptyset$.

The alternating direction method of multipliers (ADMM) [21, 23] is a first-order algorithm for solving (56) which has become popular over the last decade largely due to its wide range of applications in data science (see, *e.g.*, [11]). As applied to (56), one iteration of the ADMM may be described as:

$$x^{k+1} \in \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ f(x) + \langle p^k, x \rangle + \frac{c}{2} \|x - z^k\|^2 \right\}, \quad (57)$$

$$z^{k+1} \in \operatorname{argmin}_{z \in \mathbb{R}^n} \left\{ g(z) - \langle p^k, z \rangle + \frac{c}{2} \|x^{k+1} - z\|^2 \right\}, \quad (58)$$

$$p^{k+1} = p^k + c(x^{k+1} - z^{k+1}). \quad (59)$$

In many applications, the function g is such that (58) has a closed-form or otherwise straightforward solution (*e.g.*, $g(\cdot) = \|\cdot\|_1$). We consider situations in which this is the case, but solving (58) is more difficult and requires some form of iterative process. Eckstein and Yao [18, Section 6] proposed and studied the asymptotic convergence of an inexact version of the ADMM tailored to such situations: at each iteration, (57) may be approximately solved within a relative-error tolerance. This method is a special version of their inexact relative-error Douglas-Rachford (DR) algorithm mentioned in Section 3, as applied to the monotone inclusion problem

$$0 \in \partial f(x) + \partial g(x) \quad (60)$$

which is, in particular, a special case of (34) with $A = \partial f$ and $B = \partial g$. Problem (60) is, under standard qualification conditions, equivalent to (56). Recall that we are assuming $(\partial f + \partial g)^{-1}(0) \neq \emptyset$, *i.e.*, that (60) admits at least one solution.

In this section, we propose and study the asymptotic behaviour of a (partially) inexact relative-error *inertial-relaxed* ADMM algorithm for solving (56). The proposed method, namely Algorithm 3, is a special version of Algorithm 2 when applied to solving (60) and may be viewed as an alternative to the Eckstein-Yao approximate ADMM [18] that incorporates inertial and relaxation effects to accelerate convergence.

To formalize the inexact solution process for the subproblems (57), we introduce the notion of an \mathcal{F} -procedure [18]. First, we note that any instance of (57) can be posed slightly more abstractly as

$$\min_{x \in \mathbb{R}^n} \left\{ f(x) + \langle p, x \rangle + \frac{c}{2} \|x - z\|^2 \right\} \quad (61)$$

where $p, z \in \mathbb{R}^n$ and $c > 0$.

Definition 4.1 (\mathcal{F} -procedure for solving (61)). *A \mathcal{F} -procedure for (approximately) solving any instance of (61) is a mapping $\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2) : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{N}^* \rightarrow \mathbb{R}^n \times \mathbb{R}^n$ such that if one lets $(x^\ell, y^\ell) = \mathcal{F}(p, z, c, \bar{x}, \ell)$ for all $\ell \in \mathbb{N}$ and any given $p, z, \bar{x} \in \mathbb{R}^n$ and $c > 0$, then*

$$\lim_{\ell \rightarrow \infty} y^\ell = 0 \quad \text{and} \quad (\forall \ell \in \mathbb{N}) \quad y^\ell \in \partial_x \left[f(x) + \langle p, x \rangle + \frac{c}{2} \|x - z\|^2 \right]_{x=x^\ell}. \quad (62)$$

Quoting [18, Assumption 2], “the idea behind this definition is that $\mathcal{F}(p, z, c, \bar{x}, \ell)$ is the ℓ^{th} iterate produced by the x -subproblem solution procedure with penalty parameter c , the Lagrange multiplier estimate p^k equal to p , and $z^k = z$, starting from the solution estimate \bar{x} ”. For the remainder of this section, we assume the following.

Assumption 2. *There exists a \mathcal{F} -procedure (according to Definition 4.1) for approximately solving any instance of (61).*

The next lemma shows that the \mathcal{F} -procedure is essentially a form of \mathcal{B} -procedure (see Definition 3.1). Although the proof essentially duplicates analysis in [17, 18], it is not presented as a separate result there. Therefore we include the proof in the interest of rigor and completeness.

Lemma 4.2. *Let $\mathcal{F}(\cdot) = (\mathcal{F}_1(\cdot), \mathcal{F}_2(\cdot))$ be a \mathcal{F} -procedure for solving (61), where $\mathcal{F}_i : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{N}^* \rightarrow \mathbb{R}^n$, for $i = 1, 2$, and define $\mathcal{B} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{N}^* \rightarrow \mathbb{R}^n \times \mathbb{R}^n$ by*

$$\mathcal{B}(r, b, \gamma, \bar{s}, \bar{b}, \ell) = \mathcal{F}(-b, r, \gamma^{-1}, \bar{s}, \ell) + (0, b - \gamma^{-1}(\mathcal{F}_1(-b, r, \gamma^{-1}, \bar{s}, \ell) - r)). \quad (63)$$

Then, \mathcal{B} is a \mathcal{B} -procedure (see Definition 3.1) for approximately solving (38) in which $s := x$, $B := \partial f$, $\gamma = c^{-1}$, $r := z$ and $b := -p$.

Proof. Assume that $(s^\ell, b^\ell) = \mathcal{B}(r, b, \gamma, \bar{s}, \bar{b}, \ell)$ for some $r, b, \bar{s}, \bar{b} \in \mathbb{R}^n$, $\gamma > 0$ and all $\ell \in \mathbb{N}^*$. In view of (63) and the fact that $\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2)$ we have

$$(s^\ell, b^\ell) = (\mathcal{F}_1(-b, r, \gamma^{-1}, \bar{s}, \ell), \mathcal{F}_2(-b, r, \gamma^{-1}, \bar{s}, \ell)) + (0, b - \gamma^{-1}(\mathcal{F}_1(-b, r, \gamma^{-1}, \bar{s}, \ell) - r))$$

and so, for all $\ell \in \mathbb{N}^*$,

$$(s^\ell, b^\ell - b + \gamma^{-1}(s^\ell - r)) = (\mathcal{F}_1(-b, r, \gamma^{-1}, \bar{s}, \ell), \mathcal{F}_2(-b, r, \gamma^{-1}, \bar{s}, \ell)) = \mathcal{F}(-b, r, \gamma^{-1}, \bar{s}, \ell).$$

Using the latter identity and the fact that $\mathcal{F}(\cdot)$ is a \mathcal{F} -procedure (see Definition 4.1) we obtain

$$\lim_{\ell \rightarrow \infty} \underbrace{(b^\ell - b + \gamma^{-1}(s^\ell - r))}_{=: y^\ell} = 0 \quad \text{and} \quad (\forall \ell \in \mathbb{N}) \quad y^\ell \in \partial_x \left[f(x) - \langle b, x \rangle + \frac{1}{2\gamma} \|x - r\|^2 \right]_{x=s^\ell}$$

which, in particular, after some computations, yields $(s^\ell, b^\ell) \in G(\partial f)$, i.e., $b^\ell \in \partial f(s^\ell)$ for all $\ell \in \mathbb{N}^*$. Using this fact and the definition of y^ℓ we find $s^\ell = (\gamma \partial f + I)^{-1}(r + \gamma(y^\ell + b))$, which

in turn combined with the fact that $\lim_{\ell \rightarrow \infty} y^\ell = 0$ and the continuity of $J_{\gamma \partial f} := (\gamma \partial f + I)^{-1}$ implies that $s^\ell \rightarrow J_{\gamma \partial f}(r + \gamma b)$. On the other hand, using the definition of y^ℓ (again) we also obtain $\gamma b^\ell + s^\ell = \gamma(y^\ell + b) + r$, which gives that $\{b^\ell\}$ is convergent and $\gamma b^\ell + s^\ell \rightarrow r + \gamma b$. Altogether, we proved that $(s^\ell, b^\ell) \in \partial f$, for all $\ell \in \mathbb{N}^*$, that the sequence $\{(s^\ell, b^\ell)\}$ is convergent and $s^\ell + \gamma b^\ell \rightarrow r + \gamma b$, which finishes the proof. \square

Our inertial-relaxed inexact ADMM for solving (56) is presented as Algorithm 3. Before establishing its convergence, we make the following remarks regarding this algorithm:

- (i) Similarly to Algorithm 2, Algorithm 3 benefits from inertial and relaxation effects — see (64) and (72) — as well as from the relative error criterion (69) allowing inexact solution of the f -subproblem (65).
- (ii) Algorithm 3 can be viewed as an inertial-relaxed version of Algorithm 4 in [18], but we emphasize that even without inertia or relaxation (that is, when $\alpha = 0$ and $\rho_k \equiv 1$) it differs from the latter algorithm since Algorithm 4 is based on an approximate proximal point algorithm using an extragradient “corrector” step, while Algorithm 3 is instead based indirectly on Algorithm 1, an approximate proximal point method using projective corrector steps. In developing Algorithm 3, we also experimented with using extragradient correction, but obtained better numerical performance from projective correction.
- (iii) The derivation of Algorithm 3 mirrors that in [18], except that the underlying convergence “engine” from [30] is replaced by Algorithm 1. It should be noted that [17] provides a different way of deriving approximate ADMM algorithms. This approach results in different approximate forms of the ADMM, allowing for both relative and absolute error criteria, both of a practically verifiable form. It is also possible that the work in [32] could lead to still more approximate forms of the ADMM.

Proposition 4.3. *For any given execution of Algorithm 3, define*

$$(s^k, b^k, r^k) := (x^k, -p^k, z^k) \qquad (\hat{s}^k, \hat{b}^k, \hat{r}^k) := (\hat{x}^k, -\hat{p}^k, \hat{z}^k) \qquad (73)$$

$$(s^{k,\ell}, b^{k,\ell}, r^{k,\ell}) := (x^{k,\ell}, -p^{k,\ell}, z^{k,\ell}) \qquad a^{k,\ell} := c(s^{k,\ell} - r^{k,\ell}) - b^{k,\ell} \qquad (74)$$

for all applicable k and l . Then these sequences conform to the recursions (39)-(46) in Algorithm 2 with $\gamma = 1/c$, the \mathcal{B} -procedure (63), and the maximal monotone operators $A = \partial g$ and $B = \partial f$.

Proof. In view of (73) and (64) we have

$$\begin{aligned} (\hat{s}^k, \hat{b}^k, \hat{r}^k) &= (\hat{x}^k, -\hat{p}^k, \hat{z}^k) = (x^k + \alpha_k(x^k - x^{k-1}), -p^k - \alpha_k(p^k - p^{k-1}), z^k + \alpha_k(z^k - z^{k-1})) \\ &= (s^k, b^k, r^k) + \alpha_k [(s^k, b^k, r^k) - (s^{k-1}, b^{k-1}, r^{k-1})], \end{aligned}$$

which is identical to (39) in Algorithm 2. Fix $\gamma = 1/c$. Then (66), Definition 4.1, (73) lead to

$$x^{k,\ell} = \mathcal{F}_1(\hat{p}^k, \hat{z}^k, c, \hat{x}^k, \ell) = \mathcal{F}_1(-\hat{b}^k, \hat{r}^k, \gamma^{-1}, \hat{s}^k, \ell) - \hat{r}^k. \qquad (75)$$

Algorithm 3. Partially inexact relative-error inertial-relaxed ADMM for (56)

Choose $c > 0$, $0 \leq \alpha, \sigma < 1$ and $0 < \rho < \bar{\rho} < 2$.

Initialize $(x^0, z^0, p^0) = (x^{-1}, z^{-1}, p^{-1}) \in (\mathbb{R}^n)^3$.

for $k = 0, 1, 2, \dots$ **do**

Choose $\alpha_k \in [0, \alpha]$ and define

$$(\hat{x}^k, \hat{z}^k, \hat{p}^k) = (x^k, z^k, p^k) + \alpha_k [(x^k, z^k, p^k) - (x^{k-1}, z^{k-1}, p^{k-1})] \quad (64)$$

repeat {**for** $\ell = 1, 2, \dots$ }

Improve the solution

$$x^{k+1} \approx \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ f(x) + \langle \hat{p}^k, x \rangle + \frac{c}{2} \|x - \hat{z}^k\|^2 \right\} \quad (65)$$

by setting

$$(x^{k,\ell}, y^{k,\ell}) = \mathcal{F}(\hat{p}^k, \hat{z}^k, c, \hat{x}^k, \ell) \quad (66)$$

(thus incrementally executing a step of the \mathcal{F} -procedure)

Define

$$p^{k,\ell} = \hat{p}^k + c(x^{k,\ell} - \hat{z}^k) - y^{k,\ell} \quad (67)$$

Exactly find

$$z^{k,\ell} = \operatorname{argmin}_{z \in \mathbb{R}^n} \left\{ g(z) - \langle p^{k,\ell}, z \rangle + \frac{c}{2} \|x^{k,\ell} - z\|^2 \right\} \quad (68)$$

until

$$\|y^{k,\ell}\|^2 \leq \sigma^2 (\|p^{k,\ell} - \hat{p}^k - c(z^{k,\ell} - \hat{z}^k)\|^2 + c^2 \|x^{k,\ell} - z^{k,\ell}\|^2) \quad (69)$$

if $x^{k,\ell} = z^{k,\ell}$ **then stop**

otherwise, choose $\rho_k \in [\rho, \bar{\rho}]$ and set

$$x^{k+1} = x^{k,\ell}, \quad z^{k+1} = z^{k,\ell} \quad (70)$$

$$\theta_{k+1} = \frac{\langle c(\hat{z}^k - z^{k,\ell}) - (\hat{p}^k - p^{k,\ell}), x^{k,\ell} - z^{k,\ell} \rangle}{c \|x^{k,\ell} - z^{k,\ell}\|^2} \quad (71)$$

$$p^{k+1} = \hat{p}^k + c [(1 - \rho_k \theta_{k+1}) z^{k+1} + \rho_k \theta_{k+1} x^{k+1} - \hat{z}^k] \quad (72)$$

end for

Combining (74), (67), (66), (75), (73), and (63), we deduce that

$$\begin{aligned}
(s^{k,\ell}, b^{k,\ell}) &= (x^{k,\ell}, -p^{k,\ell}) \\
&= (x^{k,\ell}, y^{k,\ell}) + (0, -\hat{p}^k - \gamma^{-1}(x^{k,\ell} - \hat{z}^k)) \\
&= \mathcal{F}(-\hat{b}^k, \hat{r}^k, \gamma^{-1}, \hat{s}^k, \ell) + (0, \hat{b}^k - \gamma^{-1}(\mathcal{F}_1(-\hat{b}^k, \hat{r}^k, \gamma^{-1}, \hat{s}^k, \ell) - \hat{r}^k)) \\
&= \mathcal{B}(\hat{r}^k, \hat{b}^k, \gamma, \hat{s}^k, \hat{b}^k, \ell),
\end{aligned}$$

which yields (40) and (41). Note now that (68) is equivalent to the condition $0 \in \partial g(z^{k,\ell}) - p^{k,\ell} + c(z^{k,\ell} - x^{k,\ell})$, which, in view of (74), is clearly equivalent to (42) with $A = \partial g$. To prove (43), note that from (73), (74), (67) and (69) we obtain

$$\begin{aligned}
\|s^{k,\ell} + \gamma b^{k,\ell} - (\hat{r}^k + \gamma \hat{b}^k)\|^2 &= \|\gamma y^{k,\ell}\|^2 \\
&\leq \gamma^2 \sigma^2 (\|p^{k,\ell} - \hat{p}^k - c(z^{k,\ell} - \hat{z}^k)\|^2 + c^2 \|x^{k,\ell} - z^{k,\ell}\|^2)
\end{aligned}$$

which in view of (73) and (74) is equivalent to (43). Finally, similar reasoning establishes that (44)-(46) are equivalent to (70)-(72). \square

Theorem 4.4 (Convergence of Algorithm 3). *Consider any execution of Algorithm 3 for which $\alpha \in [0, 1)$, $\bar{\rho} \in (0, 2)$, and $\{\alpha_k\}$ satisfy conditions (20) and (21) of Theorem 2.5. Then:*

- (a) *If for each $k \geq 0$ the outer loop (over k) executes an infinite number of times, with each inner loop (over ℓ) terminating in a finite number of iterations $\ell = \ell(k)$, then $\{x^k\}$ and $\{z^k\}$ both converge to some $x^* \in \mathbb{R}^n$ solution of (60), and $\{p^k\}$ converges to some $p^* \in \partial g(x^*)$ such that $-p^* \in \partial f(x^*)$.*
- (b) *If the outer loop executes only a finite number of times, ending with $k = \bar{k}$, with the last invocation of the inner loop executing an infinite number of times, then $\{x^{\bar{k},\ell}\}_\ell$ and $\{z^{\bar{k},\ell}\}_\ell$ both converge to some $x^* \in \mathbb{R}^n$ solution of (60), and $\{p^{\bar{k},\ell}\}_\ell$ converges to some $p^* \in \partial g(x^*)$ such that $-p^* \in \partial f(x^*)$.*
- (c) *If Algorithm 3 stops with either $p^{k,\ell} - \hat{p}^k = c(z^{k,\ell} - \hat{z}^k)$ or $x^{k,\ell} = z^{k,\ell}$ then $x^* := x^{k,\ell} = z^{k,\ell}$ is a solution of (60).*

Proof. The result follows from immediately by combining Proposition 4.3, Theorem 3.3, and the definitions of Algorithms 2 and 3. \square

5 Numerical experiments

This section describes numerical experiments on the LASSO and logistic regression problems, which are both instances of the minimization problem (56). We tested the following algorithms: the inexact relative-error ADMM *admm_primDR* from [18]; the relative-error method *relerr* from [17]; Algorithm 3 from this paper, which we denote as *admm_primDR_relx_in*; the absolute-error approximate ADMM *absgeom* discussed in [18] and a backtracking variant

of *FISTA* [10] (also discussed in [18]). We implemented all algorithms in MATLAB, and, analogously to [18], we used the following condition to terminate the outer loop:

$$\text{dist}_\infty(0, \partial_x[f(x) + g(x)]_{x=x^k}) \leq \epsilon, \quad (76)$$

where $\text{dist}_\infty(t, S) := \inf\{\|t - s\|_\infty \mid s \in S\}$, and $\epsilon > 0$ is a tolerance parameter set to 10^{-6} .

Moreover, in our implementation of Algorithm 3 from this paper, we replaced the error condition (69) with the stronger condition

$$\|y^{k,\ell}\| \leq \sigma \max\{\|p^{k,\ell} - \hat{p}^k - c(z^{k,\ell} - \hat{z}^k)\|, c\|x^{k,\ell} - z^{k,\ell}\|\}, \quad (77)$$

which we empirically found to yield better numerical performance.

5.1 Numerical experiments on the LASSO problem

In this subsection, we report numerical experiments on the LASSO problem [33]

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|^2 + \nu \|x\|_1, \quad (78)$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $\nu > 0$, which is an instance of (56) with $f(x) := (1/2)\|Ax - b\|^2$ and $g(x) := \nu\|x\|_1$. For the data A and b , we used the same (non-artificial) datasets as in [18].

We tested three algorithms for solving (78):

- The inexact relative-error ADMM *admm_primDR* from [18]. For this algorithm, we used the same parameter values as in [18], namely $\sigma = 0.99$ and $c = 1$ (except for the PEMS problem instance, for which $c = 20$).
- The relative-error algorithm *relerr* from [17]. We also used $\sigma = 0.99$, $c = 1$ (for all problem instances except PEMS, which we used $c = 20$). For this set of LASSO problems, the experiments in [17, 18] already show *admm_primDR* to outperform the algorithms of [17], as well as *FISTA* [10].
- Algorithm 3 from this paper which we denote as *admm_primDR_relx_in*. We used the parameter settings $\alpha_k \equiv \alpha = 0.18966$, $\beta = 0.18976$ and $\rho_k \equiv \underline{\rho} = \bar{\rho} = 1.4882$ — see conditions (20) and (21) and Figure 2. We also set $\sigma = 0.99$ and $c = 1$ (except for the PEMS problem instance, for which $c = 20$).

We implemented all of the algorithms in MATLAB, using a conjugate gradient procedure to approximately solve the subproblems corresponding to $f(x) = (1/2)\|Ax - b\|^2$, exactly as in [18]. Table 1 shows number of outer iterations, Table 2 shows the total number of inner (conjugate gradient) iterations, and Table 3 shows runtimes in seconds. Figure 3 shows the same results graphically. In each table, the smallest value in each row appears in bold. In terms of runtime, the new algorithm outperforms that of [18] for all problem except the *finance1000* instance.

Table 1: LASSO outer iterations; $\alpha = 0.18966$, $\beta = 0.18976$ and $\bar{\rho} = 1.4882$

| Problem | relerr <small>(iteration1)</small> | primDR <small>(iteration2)</small> | primDR_relx_in <small>(iteration3)</small> | $\frac{\text{iteration3}}{\text{iteration1}}$ | $\frac{\text{iteration3}}{\text{iteration2}}$ |
|---------------------|---------------------------------------|---------------------------------------|---|---|---|
| Ball64_singlepixcam | 280 | 278 | 123 | 0.439 | 0.442 |
| Logo64_singlepixcam | 283 | 282 | 139 | 0.491 | 0.493 |
| Mug32_singlepixcam | 153 | 153 | 136 | 0.888 | 0.888 |
| Mug128_singlepixcam | 920 | 914 | 435 | 0.473 | 0.476 |
| finance1000 | 974 | 1709 | 1079 | 1.107 | 0.631 |
| PEMS | 3354 | 3648 | 1088 | 0.324 | 0.298 |
| Brain | 1855 | 2295 | 1219 | 0.657 | 0.531 |
| Colon | 450 | 482 | 256 | 0.568 | 0.531 |
| Leukemia | 675 | 774 | 424 | 0.628 | 0.547 |
| Lymphoma | 908 | 925 | 482 | 0.531 | 0.521 |
| Prostate | 1520 | 1739 | 998 | 0.656 | 0.574 |
| srbct | 426 | 401 | 221 | 0.519 | 0.551 |
| Geometric mean | 692.06 | 761.02 | 399.85 | 0.577 | 0.525 |

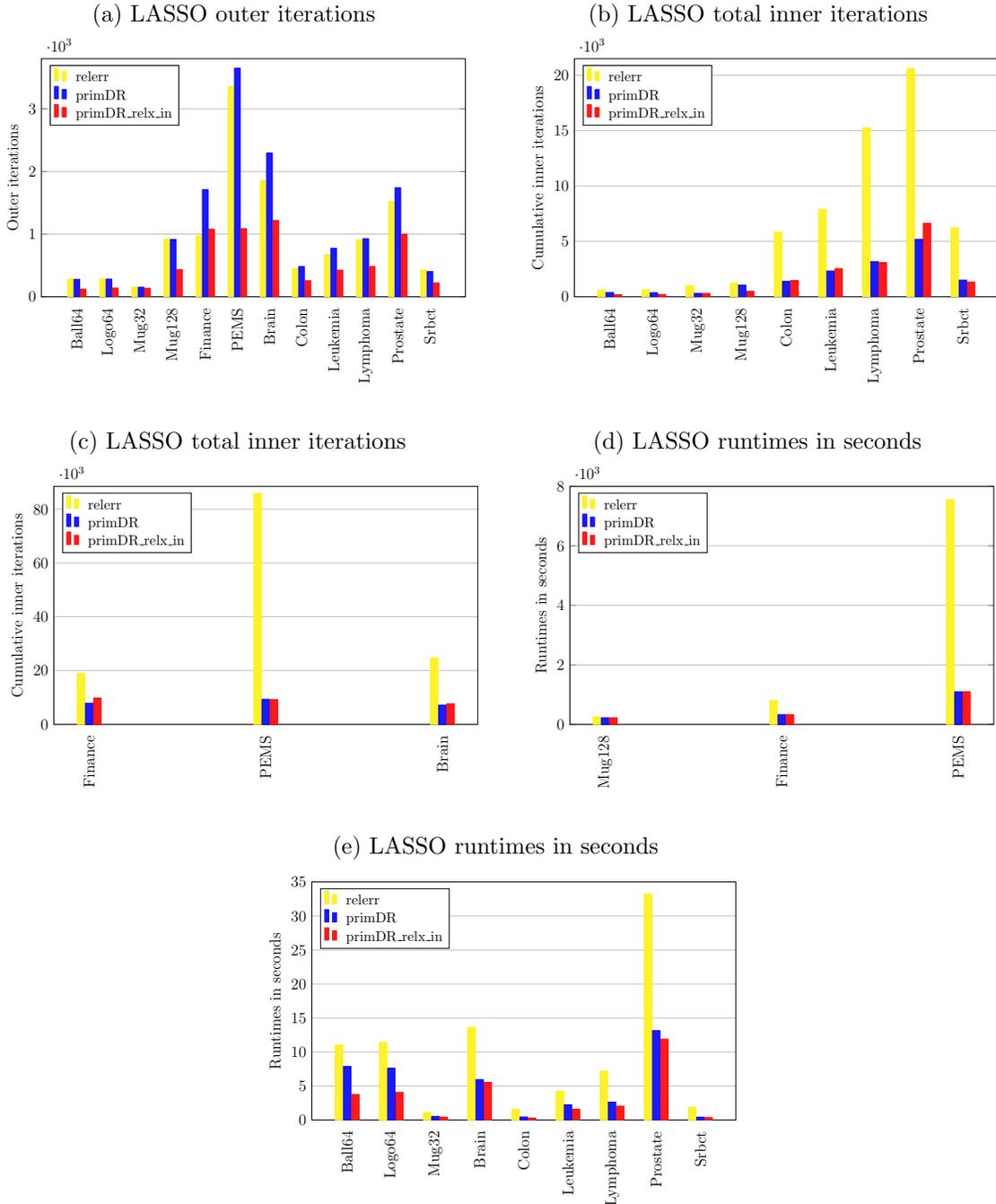
Table 2: LASSO total inner iterations; $\alpha = 0.18966$, $\beta = 0.18976$ and $\bar{\rho} = 1.4882$

| Problem | relerr <small>(iteration1)</small> | primDR <small>(iteration2)</small> | primDR_relx_in <small>(iteration3)</small> | $\frac{\text{iteration3}}{\text{iteration1}}$ | $\frac{\text{iteration3}}{\text{iteration2}}$ |
|---------------------|---------------------------------------|---------------------------------------|---|---|---|
| Ball64_singlepixcam | 603 | 382 | 191 | 0.316 | 0.500 |
| Logo64_singlepixcam | 621 | 369 | 212 | 0.341 | 0.574 |
| Mug32_singlepixcam | 998 | 307 | 302 | 0.303 | 0.984 |
| Mug128_singlepixcam | 1214 | 1046 | 488 | 0.402 | 0.466 |
| finance1000 | 18944 | 7852 | 9737 | 0.514 | 1.240 |
| PEMS | 85858 | 9318 | 9235 | 0.107 | 0.991 |
| Brain | 24612 | 7116 | 7655 | 0.311 | 1.075 |
| Colon | 5847 | 1401 | 1461 | 0.249 | 1.042 |
| Leukemia | 7888 | 2321 | 2543 | 0.322 | 1.095 |
| Lymphoma | 15266 | 3179 | 3083 | 0.202 | 0.969 |
| Prostate | 20615 | 5193 | 6629 | 0.321 | 1.276 |
| srbct | 6213 | 1505 | 1334 | 0.215 | 0.886 |
| Geometric mean | 5859.43 | 1876.32 | 1652.97 | 0.282 | 0.880 |

Table 3: LASSO runtimes in seconds; $\alpha = 0.18966$, $\beta = 0.18976$ and $\bar{\rho} = 1.4882$

| Problem | relerr <i>(time1)</i> | primDR <i>(time2)</i> | primDR_relx_in <i>(time3)</i> | $\frac{time3}{time1}$ | $\frac{time3}{time2}$ |
|---------------------|--------------------------|--------------------------|----------------------------------|-----------------------|-----------------------|
| Ball64_singlepixcam | 11.02 | 7.86 | 3.75 | 0.341 | 0.477 |
| Logo64_singlepixcam | 11.37 | 7.62 | 4.04 | 0.355 | 0.531 |
| Mug32_singlepixcam | 1.07 | 0.51 | 0.43 | 0.374 | 0.862 |
| Mug128_singlepixcam | 248.38 | 218.08 | 101.17 | 0.407 | 0.464 |
| finance1000 | 805.17 | 327.56 | 347.97 | 0.432 | 1.062 |
| PEMS | 7546.11 | 1092.16 | 988.12 | 0.131 | 0.905 |
| Brain | 13.59 | 5.94 | 5.53 | 0.407 | 0.929 |
| Colon | 1.56 | 0.45 | 0.28 | 0.179 | 0.620 |
| Leukemia | 4.24 | 2.23 | 1.59 | 0.375 | 0.717 |
| Lymphoma | 7.18 | 2.63 | 2.03 | 0.283 | 0.773 |
| Prostate | 33.21 | 13.15 | 11.88 | 0.357 | 0.904 |
| srbct | 1.83 | 0.42 | 0.35 | 0.192 | 0.847 |
| Geometric mean | 21.13 | 8.75 | 6.41 | 0.303 | 0.733 |

Figure 3: Comparison of performance in LASSO problems



5.2 Numerical experiments on logistic regression problems

This section describes numerical experiments on the ℓ_1 -regularized logistic regression problem [20, 26]

$$\min_{(w,v) \in \mathbb{R}^{n-1} \times \mathbb{R}} \sum_{i=1}^q \log(1 + \exp(-b_i(a_i^T w + v))) + \nu \|w\|_1, \quad (79)$$

using a training dataset consisting of q pairs (a_i, b_i) , where $a_i \in \mathbb{R}^{n-1}$ is a feature vector, $b_i \in \{-1, +1\}$ is the corresponding label, $w \in \mathbb{R}^{n-1}$ represents a weighting of the feature and $v \in \mathbb{R}$ represents a kind of bias. Problem (79) is clearly a special instance of (56) with $x = (v, w)$ and

$$f(v, w) := \sum_{i=1}^q \log(1 + \exp(-b_i(a_i^T w + v))) \quad \text{and} \quad g(v, w) := \nu \|w\|_1. \quad (80)$$

We considered four standard cancer DNA microarray non-artificial datasets from [14] (also used in [18, Subsection 7.2]) and tested five algorithms: *absgeom*, *relerr*, *admm_primDR*, *FISTA* and *admm_primDR_relx_in*. For *relerr* and *admm_primDR* algorithms we used the same parameter values as in Subsection 5.1; for *admm_primDR_relx_in* we used the parameter settings $\alpha_k \equiv \alpha = 0.1$, $\beta = 0.1001$ and $\rho_k \equiv \underline{\rho} = \bar{\rho} = 1.7606$ — see conditions (20) and (21) and Figure 2. We also set $\sigma = 0.99$ and $c = 1$.

Analogously to [18], we used an L-BFGS procedure to approximately solve the subproblems corresponding to $f(\cdot)$ from (80). Tables 4, 5 and 6 show outer iterations, total inner iterations and runtimes, respectively. These results are also graphically summarized in Figure 4. The new algorithm has the best aggregate performance by all measures, and the best run time for all the datasets.

A Auxiliary results

Lemma A.1 (See for example Proposition 20.33 of [9]). *If T is maximal monotone on \mathbb{R}^n , $\{(\tilde{z}^j, v^j)\}$ is such that $v^j \in T(\tilde{z}^j)$ for all $j \geq 0$, $\lim_{j \rightarrow \infty} \tilde{z}^j = z^\infty$, and $\lim_{j \rightarrow \infty} v^j = v^\infty$, then $v^\infty \in T(z^\infty)$.*

Lemma A.2. *The inverse function of the scalar map*

$$(0, 2) \ni \rho \mapsto \phi(\rho) := \frac{2(2 - \rho)}{4 - \rho + \sqrt{16\rho - 7\rho^2}} \in (0, 1)$$

is

$$(0, 1) \ni \beta \mapsto \psi(\beta) := \frac{2(\beta - 1)^2}{2(\beta - 1)^2 + 3\beta - 1} \in (0, 2).$$

Table 4: Outer iterations for logistic regression problems.

| Problem | absgeom <i>(iteration1)</i> | relerr <i>(iteration2)</i> | primDR <i>(iteration3)</i> | primDR_relx_in <i>(iteration4)</i> |
|----------------|--------------------------------|-------------------------------|-------------------------------|---------------------------------------|
| Colon | 2666 | 2145 | 1979 | 1578 |
| Leukemia | 1662 | 1116 | 922 | 788 |
| Prostate | 1936 | 1583 | 1677 | 1198 |
| Arcene | 419 | 276 | 359 | 290 |
| Geometric mean | 1376.91 | 1011.28 | 1023.76 | 810.72 |

| Problem | $\frac{\textit{iteration4}}{\textit{iteration1}}$ | $\frac{\textit{iteration4}}{\textit{iteration2}}$ | $\frac{\textit{iteration4}}{\textit{iteration3}}$ | $\frac{\textit{iteration2}}{\textit{iteration3}}$ |
|----------------|---|---|---|---|
| Colon | 0.5919 | 0.7356 | 0.7974 | 1.0839 |
| Leukemia | 0.4741 | 0.7061 | 0.8546 | 1.2104 |
| Prostate | 0.6188 | 0.7568 | 0.7144 | 0.9439 |
| Arcene | 0.6921 | 1.0507 | 0.8078 | 0.7688 |
| Geometric mean | 0.5887 | 0.8016 | 0.7924 | 0.9849 |

Table 5: Total inner iterations for logistic regression problems.

| Problem | absgeom <i>(iteration1)</i> | relerr <i>(iteration2)</i> | primDR <i>(iteration3)</i> | FISTA <i>(iteration4)</i> | primDR_relx_in <i>(iteration5)</i> |
|----------------|--------------------------------|-------------------------------|-------------------------------|------------------------------|---------------------------------------|
| Colon | 20612 | 23919 | 21697 | 26247 | 8283 |
| Leukemia | 7715 | 12086 | 11625 | 6536 | 4448 |
| Prostate | 18901 | 27505 | 24548 | 13730 | 10997 |
| Arcene | 780 | 3236 | 3589 | 4648 | 1450 |
| Geometric mean | 6958.73 | 12665.18 | 12209.43 | 10228.97 | 4923.21 |

| Problem | $\frac{\textit{iteration5}}{\textit{iteration1}}$ | $\frac{\textit{iteration5}}{\textit{iteration2}}$ | $\frac{\textit{iteration5}}{\textit{iteration3}}$ | $\frac{\textit{iteration5}}{\textit{iteration4}}$ | $\frac{\textit{iteration4}}{\textit{iteration1}}$ |
|----------------|---|---|---|---|---|
| Colon | 0.4018 | 0.3463 | 0.3817 | 0.3156 | 0.9499 |
| Leukemia | 0.5765 | 0.3681 | 0.3826 | 0.6805 | 0.6636 |
| Prostate | 0.5818 | 0.3998 | 0.4479 | 0.8009 | 0.7699 |
| Arcene | 1.8589 | 0.4481 | 0.4041 | 0.3119 | 0.2173 |
| Geometric mean | 0.7074 | 0.4032 | 0.4032 | 0.4813 | 0.5699 |

Figure 4: Comparison of performance in logistic regression problems

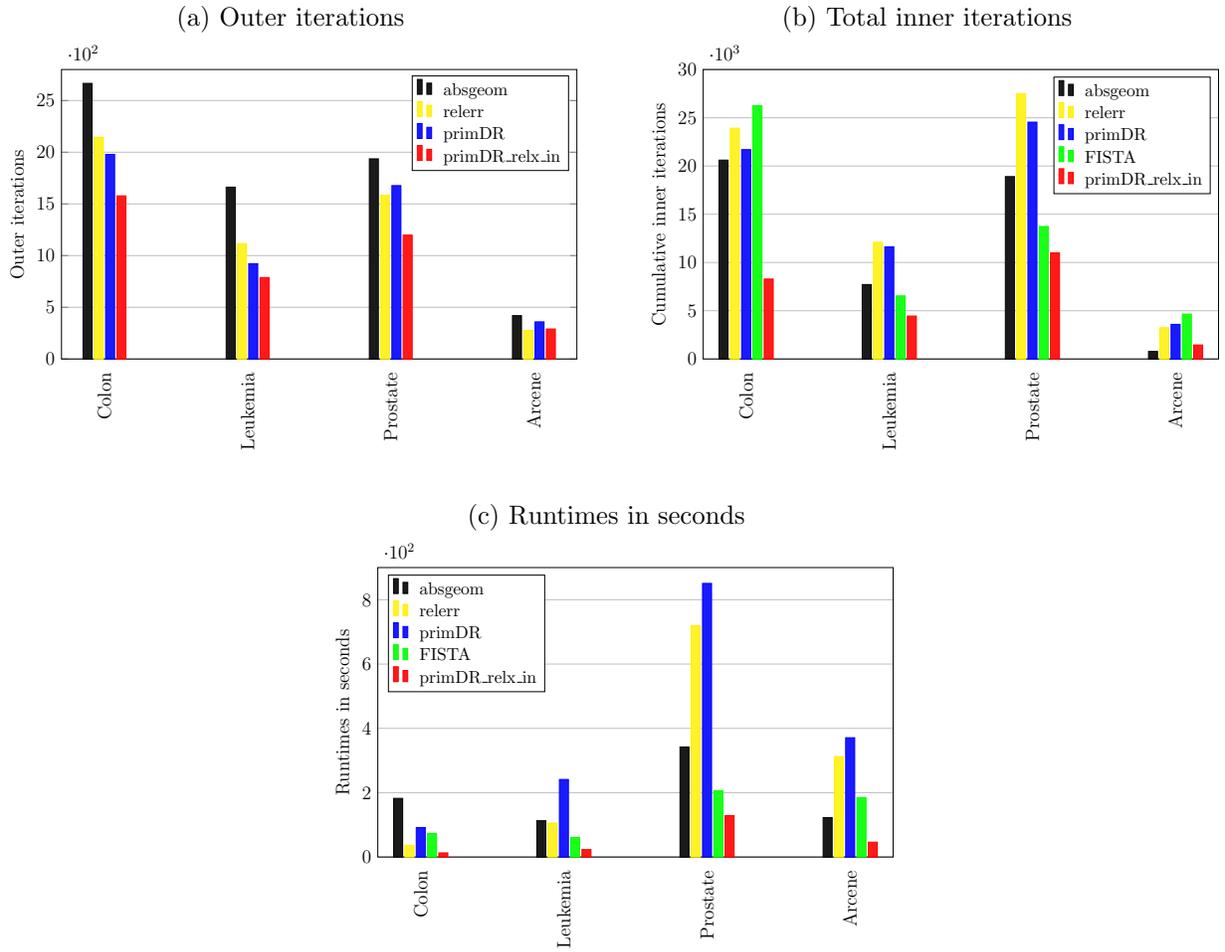


Table 6: Logistic regression runtimes in seconds.

| Problem | absgeom <small>(time1)</small> | relerr <small>(time2)</small> | primDR <small>(time3)</small> | FISTA <small>(time4)</small> | primDR_relx_in <small>(time5)</small> |
|----------------|-----------------------------------|----------------------------------|----------------------------------|---------------------------------|--|
| Colon | 182.3601 | 36.5207 | 91.5726 | 73.2987 | 12.8243 |
| Leukemia | 112.7412 | 105.4221 | 241.1378 | 60.9476 | 23.0547 |
| Prostate | 342.1609 | 719.6731 | 850.8159 | 206.3883 | 128.6972 |
| Arcene | 122.7208 | 312.1101 | 370.9415 | 184.3489 | 46.1276 |
| Geometric mean | 171.41 | 224.11 | 288.93 | 114.18 | 36.39 |

| Problem | $\frac{time5}{time1}$ | $\frac{time5}{time2}$ | $\frac{time5}{time3}$ | $\frac{time5}{time4}$ | $\frac{time4}{time3}$ |
|----------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Colon | 0.0703 | 0.1203 | 0.1401 | 0.1749 | 0.8003 |
| Leukemia | 0.2045 | 0.2186 | 0.0956 | 1.0215 | 0.2527 |
| Prostate | 0.3761 | 0.1788 | 0.1513 | 0.6236 | 0.2426 |
| Arcene | 0.3759 | 0.1478 | 0.1244 | 0.2502 | 0.4969 |
| Geometric mean | 0.2123 | 0.1623 | 0.1259 | 0.3187 | 0.3951 |

Proof. We first claim that $\psi(\beta) \in [0, 2]$ for all $\beta \in [0, 1]$ and $\psi(\beta) \in (0, 2)$ for all $\beta \in (0, 1)$. To establish this claim, we first note that by elementary calculus and some simplifications, we have

$$\frac{d}{d\beta}\psi(\beta) = \frac{6\beta^2 - 4\beta - 2}{(2(\beta - 1)^2 + 3\beta - 1)^2} = \frac{6\beta^2 - 4\beta - 2}{(2\beta^2 - \beta + 1)^2}. \quad (81)$$

The discriminant of $2\beta^2 - \beta + 1$ is negative, so it has no real roots and the denominator of (81) is always positive. The expression in the numerator is convex and applying the quadratic formula yields that its roots are $-1/3$ and 1 , so therefore it is nonpositive on $[0, 1]$ and negative on $(0, 1)$. Therefore, $\frac{d}{d\beta}\psi(\beta)$ exists for all $\beta \in [0, 1]$ and is negative for all $\beta \in (0, 1)$, implying that ψ is a decreasing function on $(0, 1)$. By direct calculation, $\psi(0) = 2$ and $\psi(1) = 0$, so therefore $\{\psi(\beta) \mid \beta \in [0, 1]\} = [0, 2]$ and $\{\psi(\beta) \mid \beta \in (0, 1)\} = (0, 2)$, establishing the initial claim. To continue the proof, we next establish that

$$\phi(\psi(\beta)) = \beta \quad \forall \beta \in (0, 1). \quad (82)$$

To this end, fix any $\beta \in (0, 1)$ and define

$$(0, 2) \ni \rho := \psi(\beta) = \frac{2(\beta - 1)^2}{2(\beta - 1)^2 + 3\beta - 1} = \frac{2\beta^2 - 4\beta + 2}{2\beta^2 - \beta + 1},$$

which implies the quadratic equation

$$2(1 - \rho)\beta^2 - (4 - \rho)\beta + (2 - \rho) = 0. \quad (83)$$

We now consider three cases in (83): $\rho = 1$, $\rho < 1$, and $\rho > 1$.

$\rho = 1$: in this case, simplification of (83) and the definition of ϕ yield that $\beta = 1/3 = \phi(1)$.

$\rho < 1$: the unique minimizer of the quadratic function in (83) is $\beta^* := (4 - \rho)/(4(1 - \rho))$, which must be greater than 1 because $\rho > 0$. Thus, we have $\beta^* > 1 > \beta > 0$, so β is the smaller root of the quadratic equation in (83). Using the quadratic formula and rationalizing the denominator,

$$\beta = \frac{4 - \rho - \sqrt{(\rho - 4)^2 - 4 \cdot 2(1 - \rho)(2 - \rho)}}{2 \cdot 2(1 - \rho)} = \frac{4 - \rho - \sqrt{16\rho - 7\rho^2}}{4(1 - \rho)} \quad (84)$$

$$\begin{aligned} &= \frac{4 - \rho - \sqrt{16\rho - 7\rho^2}}{4(1 - \rho)} \cdot \frac{4 - \rho + \sqrt{16\rho - 7\rho^2}}{4 - \rho + \sqrt{16\rho - 7\rho^2}} \\ &= \frac{16 - 24\rho + 8\rho^2}{4(1 - \rho)(4 - \rho + \sqrt{16\rho - 7\rho^2})} = \frac{8(1 - \rho)(2 - \rho)}{4(1 - \rho)(4 - \rho + \sqrt{16\rho - 7\rho^2})} \\ &= \frac{2(2 - \rho)}{4 - \rho + \sqrt{16\rho - 7\rho^2}} = \phi(\rho). \end{aligned} \quad (85)$$

$\rho > 1$: in this case, β^* as defined in the previous case is the unique maximizer of the quadratic function in (83) and $\beta^* < 0$. So $\beta^* < 0 < \beta < 1$ and β is the larger root of the quadratic in (83). Since the coefficient of the quadratic term is negative in this case, this root also takes the form (84), and consequently (85) still holds.

The proof of (82) is now complete. Finally, we now prove that

$$\psi(\phi(\rho)) = \rho \quad \forall \rho \in (0, 2). \quad (86)$$

To this end, let $0 < \rho < 2$ and define

$$(0, 1) \ni \beta := \phi(\rho) = \frac{2(2 - \rho)}{4 - \rho + \sqrt{16\rho - 7\rho^2}}.$$

Using the above definition and the quadratic formula, we conclude that β also satisfies the quadratic equation (83), which after some simple algebra gives

$$\rho = \frac{2(\beta - 1)^2}{2(\beta - 1)^2 + 3\beta - 1},$$

that is, $\rho = \psi(\beta)$, which in turn is equivalent to (86). \square

Lemma A.3. *Let $\mathbb{R} \ni \nu \mapsto q(\nu) := a\nu^2 - b\nu + c$ be a real function and assume that $b, c > 0$ and $b^2 - 4ac > 0$. Define*

$$\beta := \frac{2c}{b + \sqrt{b^2 - 4ac}} > 0. \quad (87)$$

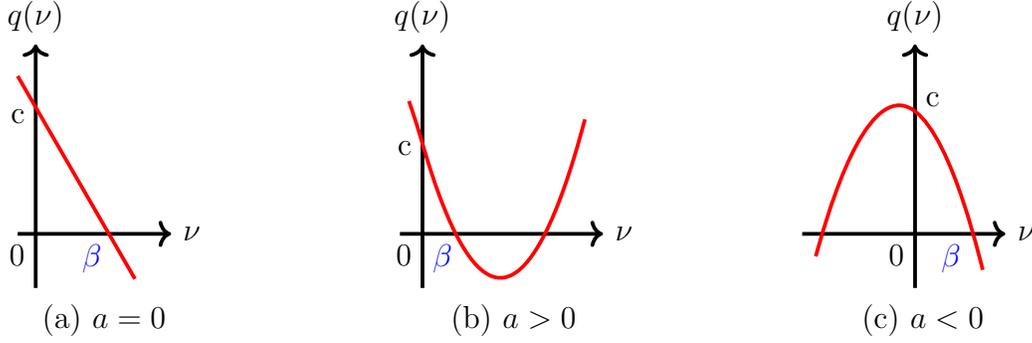


Figure 5: Possible cases for the quadratic function $q(\cdot)$ in Lemma A.3.

- (i) If $a = 0$, then $q(\cdot)$ is a decreasing affine function and $\beta > 0$ as in (87) is its unique root (see Figure 5(a)).
- (ii) If $a > 0$ (resp. $a < 0$), then $q(\cdot)$ is a convex (resp. concave) quadratic function and $\beta > 0$ as in (87) is its smallest (resp. largest) root (see Figure 5(b) and Figure 5(c), resp.).

In both cases (i) and (ii), $\beta > 0$ as in (87) is a root of $q(\cdot)$, and $q(\cdot)$ is decreasing in the interval $[0, \beta]$ (see Figure 5).

Proof. The proof of (i) is straightforward. To prove (ii), note that rationalizing the denominator of (87) results in $\beta = (b - \sqrt{b^2 - 4ac})/2a$, which in turn implies that (ii) follows from the quadratic formula and the assumption that $b, c > 0$. The last statement of the lemma is a direct consequence of (i), (ii) and the assumption that $b, c > 0$. \square

Lemma A.4 (Opial [27]). *Let $\emptyset \neq \Omega \subset \mathbb{R}^n$ and $\{z^k\}$ be a sequence in \mathbb{R}^n such that every cluster point of $\{z^k\}$ belongs to Ω and $\lim_{k \rightarrow \infty} \|z^k - z^*\|$ exists for every $z^* \in \Omega$. Then $\{z^k\}$ converges to a point in Ω .*

The following lemma was essentially proved by Alvarez and Attouch in [2, Theorem 2.1].

Lemma A.5. *Let the sequences $\{\varphi_k\}$, $\{s_k\}$, $\{\alpha_k\}$ and $\{\delta_k\}$ in $[0, +\infty[$ and $\alpha \in \mathbb{R}$ be such that $\varphi_0 = \varphi_{-1}$, $0 \leq \alpha_k \leq \alpha < 1$ and*

$$\varphi_{k+1} - \varphi_k + s_{k+1} \leq \alpha_k(\varphi_k - \varphi_{k-1}) + \delta_k \quad \forall k \geq 0. \quad (88)$$

The following hold:

- (a) For all $k \geq 1$,

$$\varphi_k + \sum_{j=1}^k s_j \leq \varphi_0 + \frac{1}{1 - \alpha} \sum_{j=0}^{k-1} \delta_j. \quad (89)$$

(b) If $\sum_{k=0}^{\infty} \delta_k < +\infty$, then $\lim_{k \rightarrow \infty} \varphi_k$ exists, i.e., the sequence $\{\varphi_k\}$ converges to some element in $[0, \infty)$.

Proof. It was proved in [2, Theorem 2.1] that $\mathcal{M} := (1 - \alpha)^{-1} \sum_{j=0}^k \delta_j \geq \sum_{j=1}^{k+1} [\varphi_j - \varphi_{j-1}]_+$, where $[\cdot]_+ = \max\{\cdot, 0\}$. Using this, the assumptions $\varphi_0 = \varphi_{-1}$, $0 \leq \alpha_k \leq \alpha < 1$ and (88), and some algebraic manipulations we find, for all $k \geq 0$,

$$\begin{aligned} \varphi_{k+1} + \sum_{j=1}^{k+1} s_j &\leq \varphi_0 + \alpha \sum_{j=1}^{k+1} [\varphi_j - \varphi_{j-1}]_+ + \sum_{j=0}^k \delta_j \\ &\leq \varphi_0 + \alpha \mathcal{M} + (1 - \alpha) \mathcal{M} = \varphi_0 + \mathcal{M}, \end{aligned}$$

which proves (a). To finish the proof, we note that (b) was established within the proof of [2, Theorem 2.1]. \square

References

- [1] F. Alvarez. Weak convergence of a relaxed and inertial hybrid projection-proximal point algorithm for maximal monotone operators in Hilbert space. *SIAM J. Optim.*, 14(3):773–782, 2003.
- [2] F. Alvarez and H. Attouch. An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping. *Set-Valued Anal.*, 9(1-2):3–11, 2001.
- [3] M. Marques Alves and M. Geremia. Iteration complexity of an inexact Douglas-Rachford method and of a Douglas-Rachford-Tseng’s F-B four-operator splitting method for solving monotone inclusions. *Numerical Algorithms*, to appear, 2019.
- [4] M. Marques Alves and R.T. Marcavillaca. On inexact relative-error hybrid proximal extragradient, forward-backward and Tseng’s modified forward-backward methods with inertial effects,. *Set-Valued and Variational Analysis*, to appear, 2019.
- [5] H. Attouch and A. Cabot. Convergence of a relaxed inertial forward-backward algorithm for structured monotone inclusions. Preprint hal-01708216, HAL Open Archive, 2018.
- [6] H. Attouch and A. Cabot. Convergence of a relaxed inertial proximal algorithm for maximally monotone operators. Preprint hal-01708905, HAL Open Archive, 2018.
- [7] H. Attouch, Z. Chbani, J. Peypouquet, and P. Redont. Fast convergence of inertial dynamics and algorithms with asymptotic vanishing viscosity. *Math. Program.*, 168(1-2, Ser. B):123–175, 2018.
- [8] H. Attouch, J. Peypouquet, and P. Redont. Fast convex optimization via inertial dynamics with Hessian driven damping. *J. Differential Equations*, 261(10):5734–5783, 2016.

- [9] H. H. Bauschke and P. L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC. Springer, New York, 2011.
- [10] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.
- [11] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, 2011.
- [12] C. Chen, R. H. Chan, S. Ma, and J. Yang. Inertial proximal ADMM for linearly constrained separable convex optimization. *SIAM J. Imaging Sci.*, 8(4):2239–2267, 2015.
- [13] C. Chen, S. Ma, and J. Yang. A general inertial proximal point algorithm for mixed variational inequality problem. *SIAM J. Optim.*, 25(4):2120–2142, 2015.
- [14] M. Dettling and P. Bühlmann. Finding predictive gene groups from microarray data. *J. Multivariate Anal.*, 90(1):106–131, 2004.
- [15] J. Douglas, Jr. and H. H. Rachford, Jr. On the numerical solution of heat conduction problems in two and three space variables. *Trans. Amer. Math. Soc.*, 82:421–439, 1956.
- [16] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program.*, 55(3):293–318, 1992.
- [17] J. Eckstein and W. Yao. Approximate ADMM algorithms derived from Lagrangian splitting. *Comput. Optim. Appl.*, 68(2):363–405, 2017.
- [18] J. Eckstein and W. Yao. Relative-error approximate versions of Douglas-Rachford splitting and special cases of the ADMM. *Math. Program.*, 170(2):417–444, 2018.
- [19] Michel Fortin and Roland Glowinski. On decomposition-coordination methods using an augmented Lagrangian. In M. Fortin and R. Glowinski, editors, *Augmented Lagrangian methods: Applications to the numerical solution of boundary-value problems*, volume 15 of *Studies in Mathematics and its Applications*, pages 97–146. North-Holland, Amsterdam, 1983.
- [20] J. Franklin. The elements of statistical learning: data mining, inference and prediction. *Math. Intelligencer*, 27(2):83–85, 2005.
- [21] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Comput. Math. Appl.*, 2:17–40, 1976.

- [22] Daniel Gabay. Applications of the method of multipliers to variational inequalities. In M. Fortin and R. Glowinski, editors, *Augmented Lagrangian methods: Applications to the numerical solution of boundary-value problems*, volume 15 of *Studies in Mathematics and its Applications*, pages 299–331. North-Holland, Amsterdam, 1983.
- [23] R. Glowinski and A. Marroco. Sur l’approximation, par éléments finis d’ordre 1, et la résolution, par pénalisation-dualité, d’une classe de problèmes de Dirichlet non linéaires. *C. R. Acad. Sci. Paris Sér. A*, 278:1649–1652, 1974.
- [24] P.-L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.*, 16(6):964–979, 1979.
- [25] D. A. Lorenz and T. Pock. An inertial forward-backward algorithm for monotone inclusions. *J. Math. Imaging Vision*, 51(2):311–325, 2015.
- [26] A. Y. Ng. Feature selection, L_1 vs. L_2 regularization, and rotational invariance. In *Proceedings, Twenty-First International Conference on Machine Learning, ICML*, pages 615–622, 2004.
- [27] Z. Opial. Weak convergence of the sequence of successive approximations for nonexpansive mappings. *Bull. Amer. Math. Soc.*, 73:591–597, 1967.
- [28] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM J. Control Optimization*, 14(5):877–898, 1976.
- [29] M. V. Solodov and B. F. Svaiter. A hybrid approximate extragradient-proximal point algorithm using the enlargement of a maximal monotone operator. *Set-Valued Anal.*, 7(4):323–345, 1999.
- [30] M. V. Solodov and B. F. Svaiter. A hybrid projection-proximal point algorithm. *J. Convex Anal.*, 6(1):59–70, 1999.
- [31] M. V. Solodov and B. F. Svaiter. A unified framework for some inexact proximal point algorithms. *Numer. Funct. Anal. Optim.*, 22(7-8):1013–1035, 2001.
- [32] B. F. Svaiter. A weakly convergent fully inexact Douglas-Rachford method with relative error tolerance. Preprint 1809.02312, ArXiv, 2018.
- [33] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1):267–288, 1996.