Non-stationary Douglas-Rachford and alternating direction method of multipliers: adaptive stepsizes and convergence*

Dirk A. Lorenz † Quoc Tran-Dinh ‡

January 14, 2018

Abstract

We revisit the classical Douglas-Rachford (DR) method for finding a zero of the sum of two maximal monotone operators. Since the practical performance of the DR method crucially depends on the stepsizes, we aim at developing an adaptive stepsize rule. To that end, we take a closer look at a linear case of the problem and use our findings to develop a stepsize strategy that eliminates the need for stepsize tuning. We analyze a general non-stationary DR scheme and prove its convergence for a convergent sequence of stepsizes with summable increments. This, in turn, proves the convergence of the method with the new adaptive stepsize rule. We also derive the related non-stationary alternating direction method of multipliers (ADMM) from such a non-stationary DR method. We illustrate the efficiency of the proposed methods on several numerical examples.

Keywords: Douglas-Rachford method, alternating direction methods of multipliers, maximal monotone inclusions, adaptive stepsize, non-stationary iteration

AMS: 90C25, 65K05, 65J15, 47H05

*This material was based upon work partially supported by the National Science Foundation under Grant DMS-1127914 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. The work of Q. Tran-Dinh was partially supported by the NSF grant, no. DMS-161984.

†Institute of Analysis and Algebra, TU Braunschweig, 38092 Braunschweig, Germany, (d.lorenz@tu-braunschweig.de).

‡Department of Statistics and Operations Research, University of North Carolina at Chapel Hill, Chapel Hill, NC (quoctd@email.unc.edu).

1 Introduction

In this paper we consider the problem of finding a zero of the sum of two maximal monotone operators, i.e., solving:

$$0 \in (A+B)x, \tag{1}$$

where $A, B : \mathcal{H} \rightrightarrows \mathcal{H}$ are two (possibly multivalued) maximal monotone operators from a Hilbert space $\mathcal{H}$ into itself [32].

The Douglas-Rachford (DR) method originated from [16] and was initially proposed to solve the discretization of stationary and non-stationary heat equations where the involved monotone operators are linear (namely, the discretization of second derivatives in different spatial directions, for example $A \approx -\partial_x^2$ and $B \approx -\partial_y^2$). The iteration uses resolvents $J_A = (\mathbb{I} + A)^{-1}$ ($\mathbb{I}$ is the identity map) and $J_B = (\mathbb{I} + B)^{-1}$, and reads as

$$u^{n+1} := J_{tB}\left(J_{tA}((\mathbb{I} - tB)u^n) + tBu^n\right), \tag{2}$$

where $t > 0$ is a given stepsize. This iteration scheme also makes sense for general maximal monotone operators as soon as B is single-valued. It has been observed in [28] that the iteration can be rewritten for arbitrary maximally monotone operators by substituting $u := J_{tB}y$ and using the identity

$$tBJ_{tB}y = tB(\mathbb{I} + tB)^{-1}y = y - (\mathbb{I} + tB)^{-1}y = y - J_{tB}y \tag{3}$$

to get

$$y^{n+1} := y^n + J_{tA}\left(2J_{tB}y^n - y^n\right) - J_{tB}y^n. \tag{4}$$

Comparing (2) and (4), we see that (4) does not require to evaluate Bu, which avoids assuming that B is single-valued as in (2). Otherwise, u^{n+1} is not uniquely defined. Note that the iterations (2) and (4) are equivalent in the stationary case, but they are not equivalent in the non-stationary case, i.e., when the stepsize t varies along the iterations; we will shed more light on this later in Section 2.2.

From a practical point of view, the performance of a DR scheme mainly depends on the following two aspects:

- *Good stepsize t:* It seems to be generally acknowledged that the choice of the stepsize is crucial for the algorithmic performance of the method but a general rule to choose the stepsize seems to be missing [2, 17]. So far, convergence theory of DR methods provides some theoretical guidance to select the parameter t in a given range in order to guarantee convergence of the method. Such a choice is often globally fixed for all iterations, and does not take into account local structures of the underlying operators A and B. Moreover, the global convergence rate of the DR method is known to be $\mathcal{O}(1/n)$ under only monotonicity assumption [14, 15, 23], where n is the iteration counter. Several experiments have shown that DR methods have better practical rate than its theoretical bound [30].

- *Proper metric:* Since the DR method is not invariant as the Newton method, the choice of metric and preconditioning seems to be crucial to accelerate its performance. Some researchers have been studying this aspect from different views, see, e.g., [11, 9, 20, 21, 22, 31]. Clearly, the choice of a metric and a preconditioner also affects the choice of the stepsize.

Note that a metric choice often depends on the variant of methods, while the choice of stepsize depends on the problem structures such as the strongly monotonicity parameters and the Lipschitz constants. In general cases, where A and B are only monotone, we only have a general rule to select the parameter t to obtain its sublinear convergence rate [15, 17, 23]. This stepsize only depends on the mentioned global parameters and does not adequately capture the local structure of A and B to adaptively update t. For instance, a linesearch procedure to evaluate a local Lipschitz constant for computing stepsize in first-order methods can beat the optimal stepsize using global Lipschitz constant [6], or a Barzilai-Borwein stepsize in gradient descent methods essentially exploits local curvature of the objective function to obtain a good performance.

Our contribution: While we will not give a general stepsize rule for t in this paper, we will propose a way of adapting the stepsize t dynamically and prove that the method will still converge. Our stepsize is not completely a "trial-and-error" quantity. It is based on an observation from analyzing the spectral radius of the DR operator in the linear case (a similar study with different focus can be found in [3]). Then, we generalize this rule to the nonlinear case when B is single-valued. When B is not single-valued, we also provide an update rule derived from the same observation. Especially, we are able to analyze the convergence of a new variant of the non-stationary DR method that differs from the non-stationary method that has been analyzed in [26]. Since DR is equivalent to ADMM in the dual setting, we also derive a similar step-size for ADMM. The adaptive stepsize for ADMM derived from the non-stationary DR method seems to be new.

Paper organization: To this end, we will analyze the case of linear monotone operators more detailed in section 2, analyze the convergence of the non-stationary form of the iteration (2), i.e. the form where $t = t_n$ varies with n, in section 3, and then propose adaptive stepsize rules in section 4. Section 5 extends the analysis to non-stationary ADMM. Finally, section 6 provides several numerical experiments.

1.1 State of the art

There are several results on the convergence of the iteration (4). The seminal paper [28] showed that, for any positive stepsize t, the iteration map in (4) is firmly nonexpansive, that the sequence $\{y^n\}$ weakly converges to a fixed point of the iteration map [28, Prop. 2] and that, $\{u^n = J_{tB}y^n\}$ weakly converges to a solution of the inclusion (1) as soon as A, B and $A + B$ are maximal monotone [28, Theorem 1]. In the case where B is coercive and Lipschitz continuous, linear convergence was also shown in [28, Proposition 4]. These results have been extended in various ways. Let us attempt to summarize some key contributions on the DR method. Eckstein and Bertsekas in [17] showed that the DR scheme can be cast into a special case of the proximal point method [32]. This allows the authors to exploit inexact computation from the proximal point method [32]. They also presented a connection between the DR method and the alternating direction method of multipliers (ADMM). In [35] Svaiter proved a weak convergence of the DR method in Hilbert space without the assumption that $A+B$ is maximal monotone. Combettes [12] cast the DR method as special case of the averaging operator from a fixed-point framework. Applications of the DR method have been studied in [13]. The convergence rate of the DR method was first studied in [28] for the strongly monotone case, while the sublinear rate was first proved in [23]. A more intensive research on convergence rates of the DR methods can be found in [14, 15]. The DR method has been extended to accelerated variant in [30] but specifying for a special setting. In [26] the authors analyzed a non-stationary DR method derived from (4) in the framework of perturbations of non-expansive iterations and showed convergence for convergent stepsize sequences with summable errors.

The DR method together with its dual variant, ADMM, become extremely popular in recent years due to a wide range of applications in image processing, and machine learning [25, 7], which are unnecessary to recall them all here.

In terms of stepsize selection for DR schemes as well as for ADMM methods, it seems that there is very little work available from the literature. Some general rules for fixed stepsizes based on further properties of the operators such as strong monotonicity, Lipschitz continuity, and coercivity are given in [19], and it is shown that the resulting linear rates are tight. Heuristic rules for fixed stepsizes motivated by quadratic problems are derived in [21]. A self-adaptive stepsize for ADMM proposed in [24] seems to be one of the first work in this direction. A recent work in [37] also proposed an adaptive update rule for stepsize in ADMM based on a spectral estimation. Some other papers rely on theoretical analysis to choose optimal stepsize such as [18], but it only works in the quadratic case. In [27], the authors proposed a nonincreasing adaptive rule for the penalty parameter in ADMM. Another update rule for ADMM can be found in [34]. While ADMM is a dual variant of the DR scheme, we unfortunately have not seen any work that converts such an adaptive step-size from ADMM to the DR scheme. In addition, the adaptive step-size for the DR scheme by itself seems to not exist in the literature.

1.2 A motivating linear example

While the Douglas-Rachford iteration (weakly) converges for all positive stepsizes $t > 0$, it seems to be folk wisdom, that there is a "sweet spot" for the stepsize which leads to fast convergence. We illustrate this effect with a simple linear example. We consider a linear equation

$$0 = Ax + Bx, \tag{5}$$

where $A, B \in \mathbb{R}^{m\times m}$ are two matrices of the size $m\times m$ with $m = 200$. We choose symmetric positive definite matrices with $\operatorname{rank}(A) = \frac{m}{2} + 10$ and $\operatorname{rank} B = \frac{m}{2}$ such that $A + B$ has full rank, and thus the equation $0 = Ax + Bx$ has zero as its unique solution.[1] Since B is single-valued, we directly use the iteration (2).

Remark 1.1. Note that the shift $\tilde{B}x = Bx - y$ would allow to treat inhomogeneous equation $(A + B)x = y$. If x^* is a solution of this equation, then one sees that iteration (2) applied to $A + \tilde{B}$ is equivalent to applying the iteration to $A + B$ but for the residual $x - x^*$.

We ran the DR scheme (2) for a given range of different values of $t > 0$, and show the residuals $\|(A + B)u^n\|$ in semi-log-scale on the left of Figure 1. One observes the following typical behavior for this example:

- A not too small stepsize ($t = 0.5$ in this case) leads to good progress in the beginning, but slows down considerably in the end.
- Large stepsizes (larger than 2 in this case) are slower in the beginning and tend to produce non-monotone decrease of the error.
- Somewhere in the middle, there is a stepsize which performs much better than the small and large stepsizes.

In this particular example the stepsize $t = 1.5$ greatly outperforms the other stepsizes. On the right of Figure 1 we show the norm of the residual after a fixed number of iterations for varying stepsizes. One can see that there is indeed a sweet spot for the stepsizes around $t = 1.5$. Note that the value of $t = 1.5$ is by no means universal and this sweet spot of 1.5 varies with the problem size, with the ranks of A and B, and even for each particular instance of this linear example.

2 Analysis of the linear monotone inclusion

In order to develop an adaptive stepsize update rule for our non-stationary DR method in the next sections, we first consider the linear problem instance of (1). We consider the original DR scheme (2) instead of (4) since (2) generates the sequence $\{u^n\}$ which converges to a solution of (1), while the sequence $\{y^n\}$ computed by (4) does not converge to a solution and its limit does depend on the stepsize in general.

[1] The exact construction of A and B is $A = C^T C$ and $B = D^T D$, where $C \in \mathbb{R}^{(0.5m+10)\times m}$ and $D \in \mathbb{R}^{0.5m\times m}$ are drawn from the standard Gaussian distribution in Matlab.

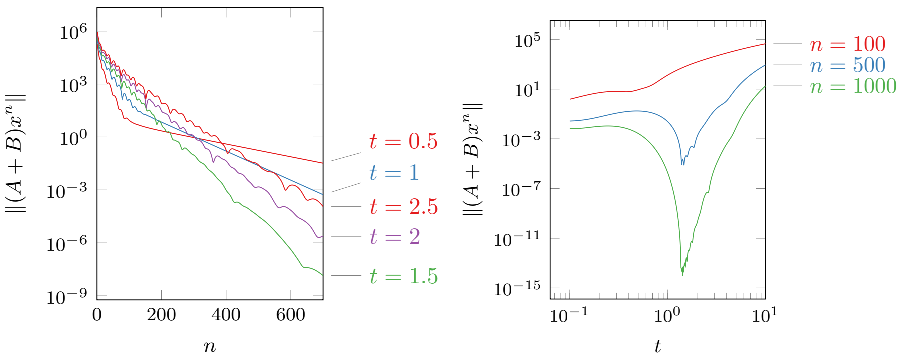

Figure 1: The residual of the Douglas-Rachford scheme (2) for the linear example. Left: The dependence of the residual on the iterations with different stepsizes. Right: The dependence of the residual on the the stepsize with different numbers of iterations.

2.1 The construction of adaptive stepsize for single-valued operator B

When both A and B are linear, the DR scheme (2) can be expressed as a fixed-point iteration scheme of the following mapping:

$$\begin{aligned} H_t &:= J_{tB}\left(J_{tA}(\mathbb{I} - tB) + tB\right) \\ &= (\mathbb{I} + tB)^{-1}(\mathbb{I} + tA)^{-1}(\mathbb{I} + t^2 AB) \\ &= (\mathbb{I} + tA + tB + t^2 AB)^{-1}(\mathbb{I} + t^2 AB). \end{aligned} \tag{6}$$

Recall that, by Remark 1.1, all of this section also applies not only to problem $(A+B)x = 0$ but also problem $(A+B)x = y$. The notion of a monotone operator has a natural equivalence for matrices, which is, however, not widely used. Hence, we recall the following definition.

Definition 2.1. A matrix $A \in \mathbb{R}^{m\times m}$ is called *monotone*, if, for all $x \in \mathbb{R}^m$, it holds that $\langle x, Ax\rangle \geq 0$.

Note that any symmetric positive semidefinite (spd) matrix is monotone, but a monotone matrix is not necessarily spd. Examples of a monotone matrices that are not spd are

$$A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad \text{and} \quad A = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \quad \text{with } |t| \leq 2.$$

The first matrix is skew symmetric, i.e., $A^T = -A$ and any such matrix is monotone. Note that even if A and B are spd (as in our example in Section 1.2), the iteration map H_t in (6) is not even symmetric. Consequently, the convergence speed of the iteration scheme (2) is not governed by the norm of H_t but by its *spectral radius* $\rho(H_t)$, which is the largest magnitude of an eigenvalue of H_t. Moreover, the eigenvalues and eigenvectors of H_t are complex in general.

First, it is clear from the derivation of H_t that the eigenspace of H_t for the eigenvalue $\lambda = 1$ exactly consists of the solutions of $(A+B)x = 0$.

In the following, for any $z \in \mathbb{C}$ (the set of complex numbers) and $r > 0$, we denote by $\mathbb{B}_r(z)$ the ball of radius r centered at z. Now, we estimate other eigenvalues of H_t that are different from 1. The following lemma provides us some useful estimates.

Lemma 2.1. *Let A and B be linear and monotone, and H_t be defined by* (6). *Let $\lambda \in \mathbb{C}$ be an eigenvalue of H_t with corresponding eigenvector z. Assume that $\lambda \neq 1$ and define c by*

$$c := \frac{\mathrm{Re}(\langle Bz, z\rangle)}{t^{-1}\|z\|^2 + t\|Bz\|^2} \geq 0. \tag{7}$$

Then, we have

$$\left|\lambda - \frac{1}{2}\right| \leq \sqrt{\frac{1}{4} - \frac{c}{1+2c}} \leq \frac{1}{2},$$

i.e. $\lambda \in \mathbb{B}_{\frac{1}{2}}\left(\frac{1}{2}\right)$.

Proof. Note that for a real, linear, and monotone map M, and a complex vector $a = b + ic$, it holds that $\langle Ma, a\rangle = \langle Mb, b\rangle + \langle Mc, c\rangle + i\left\langle (M^T - M)b, c\right\rangle$ and thus, $\mathrm{Re}(\langle Ma, a\rangle) \geq 0$. This shows that $c \geq 0$.

We can see from (6) that any pair (λ, z) of eigenvalue and eigenvector of H_t fulfills

$$z + t^2 ABz = \lambda(z + tAz + tBz + t^2ABz).$$

Now, if we denote $u := Bz$, then this expression becomes

$$z + t^2Au = \lambda z + \lambda tAz + \lambda tu + \lambda t^2 Au,$$

which, by rearranging, leads to

$$-(\lambda - 1)z - \lambda tu = tA(\lambda z + (\lambda - 1)tu).$$

Hence, by monotonicity of tA, we can derive from the above relation that

$$\begin{aligned} 0 &\leq \mathrm{Re}(\langle \lambda z + (\lambda-1)tu, -(\lambda-1)z - \lambda tu\rangle) \\ &= -\mathrm{Re}(\lambda(\bar{\lambda} - 1))\|z\|^2 - (|\lambda|^2 + |\lambda - 1|^2)t\,\mathrm{Re}(\langle u, z\rangle) - \mathrm{Re}((\lambda - 1)\bar{\lambda})t^2\|u\|^2. \end{aligned}$$

This leads to

$$(|\lambda|^2 + |\lambda - 1|^2)\,\mathrm{Re}(\langle u, z\rangle) \leq \frac{\mathrm{Re}(\lambda - |\lambda|^2)}{t}\|z\|^2 + \mathrm{Re}((\bar{\lambda} - |\lambda|^2))t\|u\|^2.$$

Denoting $\lambda := x + iy \in \mathbb{C}$, the last expression reads as

$$(x^2 + (x-1)^2 + 2y^2)\,\mathrm{Re}(\langle u, z\rangle) \leq (x - x^2 - y^2)\Big(\frac{\|z\|^2}{t} + t\|u\|^2\Big).$$

Recalling the definition of c in (7), we get

$$(x^2 + (x-1)^2 + 2y^2)c \leq x - x^2 - y^2.$$

This is equivalent to

$$0 \leq x - x^2 - y^2 - cx^2 - c(x-1)^2 - 2cy^2 = (1+2c)(x - x^2 - y^2) - c,$$

which, in turn, is equivalent to $x^2 - x + y^2 \leq -\frac{c}{1+2c}$. Adding $\frac{1}{4}$ to both sides, it leads to $(x - \frac{1}{2})^2 + y^2 \leq \frac{1}{4} - \frac{c}{1+2c}$, which shows the desired estimate. □

In general, the eigenvalues of H_t depend on t in a complicated way. For $t = 0$, we have $H_0 = \mathbb{I}$ and hence, all eigenvalues are equal to one. For growing $t > 0$, some eigenvalues move into the interior of the circle $\mathbb{B}_{1/2}(1/2)$ and for $t \to \infty$, it seems that all eigenvalues tend to converge to the boundary of such a circle, see Figure 2 for an illustration of eigenvalue distribution.

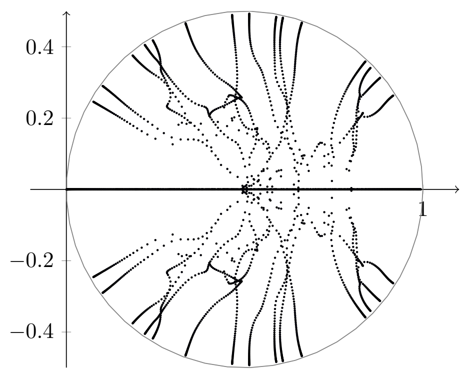

Figure 2: Eigenvalues of H_t for different values of t for a linear example similar to the example (5) in Section 1.2 (but with $m = 50$).

Remark 2.1. It appears that Lemma 2.1 is related to Proposition 4.10 of [4] and also to the fact that the iteration mapping H_t is (in the general nonlinear case) known to be not only non-expansive, but *firmly non-expansive* (cf. [17, Lemma 1] and [17, Figure 1]). In general, firmly non-expansiveness allows over-relaxation of the method, and indeed, one can also easily see this in the linear case as well: If λ is an eigenvalue of H_t, then it lies in $\mathbb{B}_{1/2}(1/2)$ (when it is not equal to one) and the corresponding eigenvalue λ_ρ of the relaxed iteration map

$$H_t^\rho = (1-\rho)\mathbb{I} + \rho H_t$$

is $\lambda_\rho = 1 - \rho + \rho\lambda$ and lies in $\mathbb{B}_{\rho/2}(1 - \frac{\rho}{2})$. Therefore, for $0 \leq \rho \leq 2$ all eigenvalues different from one of the relaxed iteration

$$u^{n+1} = (1-\rho)u^n + \rho H_t u^n$$

lie in a circle of radius $\rho/2$ centered at $1 - \rho/2$, and hence, the iteration is still non-expansive.

Lemma 2.1 tells us a little more than that all eigenvalues of the iteration map H_t lie in a circle centered at $\frac{1}{2}$ of radius $\frac{1}{2}$. Especially, all eigenvalues except for $\lambda = 1$ have magnitude strictly smaller than one and this implies that the iteration map H_t is indeed a contraction outside the set of solutions $\{x^* \in \mathcal{H} \mid (A+B)x^* = 0\}$ of (1). This proves that the stationary iteration $u^{n+1} = H_t u^n$ converges to a zero point of the map $A + B$ at a linear rate. Note that this does not imply the convergence in the non-stationary case.

To optimize the convergence speed, we aim at minimizing the contraction factor of H_t, which is the magnitude of the largest eigenvalue of H_t and there seems to be little hope to explicitly minimize the contraction factor expression.

Here is a heuristic argument based on Lemma 2.1, which we will use to derive an adaptive stepsize rule: Note that $c \mapsto \frac{c}{1+2c}$ is increasing and hence, to minimize the upper bound on λ (more precisely: the distance of λ to $\frac{1}{2}$) we want to make c from (7) as large as possible. This is achieved by minimizing the denominator of c over t which happens for

$$t = \frac{\|z\|}{\|Bz\|}.$$

This gives $c = \mathrm{Re}(\langle Bz, z\rangle)/(2\|z\|\|Bz\|)$ and note that $0 \le c \le 2$ (which implies $0 \le \frac{c}{1+2c} \le \frac{1}{4}$). This motivates an adaptive choice for the stepsize t_n as

$$t_n := \frac{\|u^n\|}{\|Bu^n\|}, \tag{8}$$

in the Douglas-Rachford iteration scheme (2).

Remark 2.2. One can use the above derivation to deduce that $t = 1/\|B\|$ is a good constant step-size. In fact, this is also the stepsize that minimizes the contraction factor for the linear rate derived in [28, Proposition 4], which is minimized when $t = 1/M$ where M is the Lipschitz constant of B. However, this choice does not perform well in practice in our experiments.

Since very limited result is known about the non-stationary Douglas-Rachford iteration in general (besides the result from [26] on convergent stepsizes with summable errors), we turn to an investigation of this method in Section 3. Before we do so, we give some more results on even simpler examples where we can derive the optimal stepsize and convergence rate explicitly in the following subsection.

Let us consider the following representative example.

Example 2.1. We consider the special case where $A = \mathbb{I}$ and B is a symmetric positive semidefinite matrix. In this case, using eigen-decomposition, we can write $B = U\Sigma U^T$ with an orthonormal matrix U and a real diagonal matrix Σ. The iteration matrix map H_t for any stepsize $t > 0$ becomes

$$H_t = (1+t)^{-1}(\mathbb{I} + tB)^{-1}(\mathbb{I} + t^2 B) = \tfrac{1}{1+t} U(\mathbb{I} + t\Sigma)^{-1}(\mathbb{I} + t^2\Sigma)U^T = USU^T,$$

where $S := \mathrm{diag}(s_i)$ is a diagonal matrix with diagonal entries

$$s_i = \frac{1 + t^2\sigma_i}{(1+t)(1+t\sigma_i)}.$$

Hence, we can compute the spectral radius of H_t as

$$\rho(H_t) = \max_{1\le i\le m} \{s_i\} = \max_{1\le i\le m} \left\{ \frac{1+t^2\sigma_i}{(1+t)(1+t\sigma_i)} \right\}.$$

Our goal is to minimize $\rho(H_t)$ over $t > 0$, i.e. to compute

$$\min_{t>0} \rho(H_t) = \min_{t>0} \max_{1\le i\le m} \left\{ \frac{1+t^2\sigma_i}{(1+t)(1+t\sigma_i)} \right\} = \min_{t>0} \left\{ \frac{1}{1+t} \right\} \left\{ \max_{1\le i\le m} \frac{1+t^2\sigma_i}{1+t\sigma_i} \right\}.$$

The function $f(\sigma) := \frac{1+t^2\sigma}{1+t\sigma}$ is strictly decreasing if $0 < t < 1$; is constant 1 for $t = 1$; and strictly increasing for $t > 1$. Consequently, the maximum is attained at the smallest eigenvalue if $0 < t \le 1$ and at the largest eigenvalue if $t > 1$. Hence, the minimization problem above reduces to

$$\min_{t\ge 0} \left\{ \frac{f(t)}{1+t} \right\}, \quad \text{with} \quad f(t) := \begin{cases} \frac{1+t^2\sigma_1}{1+t\sigma_1} & 0 < t \le 1 \\ \frac{1+t^2\sigma_m}{1+t\sigma_m} & t > 1 \end{cases}.$$

Now, we distinguish three cases:

$\sigma_1 < 1 < \sigma_m$: Careful inspection shows that, in this case, the minimum is attained at the optimal stepsize $t^* := 1$ and the optimal contraction factor is the value

$$\rho(H_{t^*}) = \frac{1}{2}.$$

$\sigma_m \le 1$: The minimum is attained in the interval $t > 1$ at the minimum of the function $f(t) = \frac{1+t^2\sigma_m}{(1+t)(1+t\sigma_m)}$. The minimum occurs at the optimal stepsize $t^* := \frac{1}{\sqrt{\sigma_m}}$ and the optimal contraction factor is

$$\rho(H_{t^*}) = \frac{2\sqrt{\sigma_m}}{(1+\sqrt{\sigma_m})^2} \le \frac{1}{2}.$$

$\sigma_1 \ge 1$: The minimum is attained in the interval $0 < t < 1$ at the minimum of the function $f(t) = \frac{1+t^2\sigma_1}{(1+t)(1+t\sigma_1)}$. The minimum occurs at the optimal stepsize $t^* := \frac{1}{\sqrt{\sigma_1}}$ and the optimal contraction factor is

$$\rho(H_{t^*}) = \frac{2\sqrt{\sigma_1}}{(1+\sqrt{\sigma_1})^2} \le \frac{1}{2}.$$

Note that even for this simple case, computing explicitly the spectral radius of H_t is still complicated. This discourages us to analyze other more complex cases along this line.

Remark 2.3. The eigenvalues of the matrix $A + B$ above are $1 + \sigma_i$, and the condition number of $A + B$ is $\kappa = \frac{1+\sigma_m}{1+\sigma_1}$. Note that the optimal contraction factor $\rho(H_t)$ of the method is always less than or equal to $1/2$, independently of the condition number. Hence, the method may converge with a linear rate smaller than $1/2$ even for a large condition number.

2.2 The construction of adaptive stepsize for non-single-valued B

In the case of multi-valued B, one needs to apply the iteration (4) instead of (2). To motivate an adaptive choice for the stepsize, we again consider the linear case, where eigenvalues can be used to describe contractions, and use this as a heuristic for the non-linear (and multivalued) case as well.

In the linear case, the iteration (4) is given by the iteration matrix

$$F_t = J_{tA}(2J_{tB} - \mathbb{I}) - J_{tB} + \mathbb{I}.$$

Comparing this with the iteration map H_t from (6) (corresponding to (2)) one notes that

$$F_t = (\mathbb{I} + tB)H_t(\mathbb{I} + tB)^{-1},$$

i.e., the matrices F_t and H_t are similar and hence, have the same eigenvalues. Moreover, if z is an eigenvector of H_t with the eigenvalue λ, then $(\mathbb{I} + tB)z$ is an eigenvector of F_t for the same eigenvalue λ.

However, in the case of the iteration (4) we do not assume that B is single-valued, and thus, the adaptive stepsize using the quotient $\|u\|/\|Bu\|$ cannot be used. However, again due to (3), we can rewrite this quotient without applying B and get, with $J_{tB}y = u$, that

$$\frac{\|u\|}{\|Bu\|} = \frac{\|J_{tB}y\|}{\|\frac{1}{t}(y - J_{tB}y)\|} = t\frac{\|J_{tB}y\|}{\|y - J_{tB}y\|}. \tag{9}$$

Note that the two iteration schemes (2) and (4) are not equivalent in the non-stationary and non-linear case. Indeed, let us consider y^n such that $u^n := J_{t_{n-1}B}y^n$. By induction, we have $u^{n+1} = J_{t_nB}y^{n+1}$. Substituting u^{n+1} into (2), we obtain

$$y^{n+1} = J_{t_nA}\left(u^n - t_nBu^n\right) + t_nBu^n. \tag{10}$$

From (3) we have

$$Bu^n = BJ_{t_{n-1}B}y^n = \tfrac{1}{t_{n-1}}\left(y^n - J_{t_{n-1}B}y^n\right).$$

Substituting $u^n = J_{t_{n-1}B}y^n$ and Bu^n into (10), we obtain

$$\begin{aligned} y^{n+1} &= J_{t_nA}\left(J_{t_{n-1}B}y^n - \tfrac{t_n}{t_{n-1}}\left(y^n - J_{t_{n-1}B}y^n\right)\right) + \tfrac{t_n}{t_{n-1}}\left(y^n - J_{t_{n-1}B}y^n\right) \\ &= \tfrac{t_n}{t_{n-1}}y^n + J_{t_nA}\left(\left(1 + \tfrac{t_n}{t_{n-1}}\right)J_{t_{n-1}B}y^n - \tfrac{t_n}{t_{n-1}}y^n\right) - \tfrac{t_n}{t_{n-1}}J_{t_{n-1}B}y^n. \end{aligned}$$

Updating t_n by (9) would then give

$$t_n := \kappa_n t_{n-1}, \quad \text{where} \quad \kappa_n := \frac{\|J_{t_{n-1}B} y^n\|}{\|y^n - J_{t_{n-1}B} y^n\|}.$$

In summary, we can write an alternative DR scheme for solving (1) as

$$\begin{cases} u^n & := J_{t_{n-1}B} y^n, \\ \kappa_n & := \frac{\|u^n\|}{\|u^n - y^n\|}, \\ t_n & := \kappa_n t_{n-1}, \\ v^n & := J_{t_n A}\left((1+\kappa_n)u^n - \kappa_n y^n\right), \\ y^{n+1} & := v^n + \kappa_n (y^n - u^n). \end{cases} \tag{11}$$

This scheme essentially has the same per-iteration complexity as in the standard DR method since the computation of κ_n does not significantly increase the cost.

Note that the non-stationary scheme (11) is notably different from the non-stationary scheme derived directly from (4) (which has been analyzed in [26]). To the best of our knowledge, the scheme (11) is new.

3 Convergence of the non-stationary DR method

In this section, we prove weak convergence of the new non-stationary scheme (11). We follow the approach by [35] and restate the DR iteration as follows: Given (u^0, b^0) such that $b^0 \in B(u^0)$ and a sequence $\{t_n\}_{n\geq 0}$, at each iteration $n \geq 0$, we iterate

$$\begin{cases} a^n \in A(v^n), & v^n + t_n a^n = u^{n-1} - t_n b^{n-1} \\ b^n \in B(u^n), & u^n + t_n b^n = v^n + t_n b^{n-1}. \end{cases} \tag{12}$$

Note that, in the case of single-valued B, this iteration reduces to

$$u^n = J_{t_n B}(J_{t_n A}(u^{n-1} - t_n B u^{n-1}) + t_n B u^{n-1}),$$

and this scheme can, as shown in Section 2.2, be transformed into the non-stationary iteration scheme (11).

Below are some consequences which we will need in our analysis:

$$u^{n-1} - u^n = t_n(a^n + b^n). \tag{13}$$

$$t_n(b^{n-1} - b^n) = u^n - v^n. \tag{14}$$

$$u^n - v^n + t_n(a^n + b^n) = t_n(a^n + b^{n-1}) = u^{n-1} - v^n. \tag{15}$$

Before proving our convergence result, we state the following lemma.

Lemma 3.1. *Let $\{\alpha_n\}$, $\{\beta_n\}$, and $\{\omega_n\}$ be three nonnegative sequences, and $\{\tau_n\}$ be a bounded sequence such that for $n \geq 0$:*

$$0 < \underline{\tau} \leq \tau_n \leq \bar{\tau}, \quad |\tau_n - \tau_{n+1}| \leq \omega_n, \quad \text{and} \quad \sum_{n=0}^{\infty} \omega_n < \infty.$$

If $\alpha_{n-1} + \tau_n\beta_{n-1} \geq \alpha_n + \tau_n\beta_n$, then $\{\alpha_n\}$ and $\{\beta_n\}$ are bounded.

Proof. If $\tau_{n+1} \leq \tau_n$, then

$$\alpha_{n-1} + \tau_n\beta_{n-1} \geq \alpha_n + \tau_n\beta_n \geq \alpha_n + \tau_{n+1}\beta_n.$$

If $\tau_{n+1} \geq \tau_n$, then $\frac{\tau_n}{\tau_{n+1}} \leq 1$ and

$$\alpha_{n-1} + \tau_n\beta_{n-1} \geq \alpha_n + \tau_n\beta_n \geq \tfrac{\tau_n}{\tau_{n+1}}\alpha_n + \tau_n\beta_n = \tfrac{\tau_n}{\tau_{n+1}}(\alpha_n + \tau_{n+1}\beta_n).$$

By the assumption that $\frac{\tau_n}{\tau_{n+1}} \geq 1 - \frac{\omega_n}{\underline{\tau}}$ and, without loss of generality, we assume that the latter term is positive (which is fulfilled for n large enough, because $\omega_n \to 0$). Thus, in both cases, we can show that

$$\alpha_{n-1} + \tau_n\beta_{n-1} \geq \left(1 - \tfrac{\omega_n}{\underline{\tau}}\right)(\alpha_n + \tau_{n+1}\beta_n).$$

Recursively, we get

$$\alpha_0 + \tau_1\beta_0 \geq \prod_{l=1}^{n}\left(1 - \tfrac{\omega_l}{\underline{\tau}}\right)(\alpha_n + \tau_{n+1}\beta_n).$$

Under the assumption $\sum_{n=0}^{\infty}\omega_n < +\infty$, we have $\prod_{l=1}^{n}\left(1 - \frac{\omega_l}{\underline{\tau}}\right) \geq M$ for some $M > 0$ and all $n \geq 1$. Then, we have $\alpha_n + \tau_{n+1}\beta_n \leq \frac{1}{M}(\alpha_0 + \tau_1\beta_0)$. This shows that $\{\alpha_n + \tau_{n+1}\beta_n\}$ is bounded. Since $\{\alpha_n\}$, $\{\beta_n\}$, and $\{\tau_n\}$ are all nonnegative, it implies that $\{\alpha_n\}$ and $\{\beta_n\}$ are bounded. □

Theorem 3.1 (Convergence of non-stationary DR)**.** *Let A and B be maximally monotone and $\{t_n\}$ be a positive sequence such that*

$$0 < \underline{t} \leq t_n \leq \bar{t}, \quad \sum_{n=0}^{\infty}|t_n - t_{n+1}| < \infty \quad \text{and} \quad t_n \to t^*,$$

where $0 < \underline{t} \leq \bar{t} < +\infty$ are given. Then, the sequence $\{(u^n, b^n)\}$ generated by the iteration scheme (12) *weakly converges to some (u^*, b^*) in the extended solution set $S(A,B) = \{(z,w) \mid w \in B(z),\ -w \in A(z)\}$ of* (1)*, so in particular, $0 \in (A+B)(u^*)$.*

Proof. The proof of this theorem follows the proof of [35, Theorem 1]. First, we observe that, for any $(u,b) \in S(A,B)$, we have

$$\begin{aligned}
\left\langle u^{n-1} - u^n, u^n - u\right\rangle &= t_n\left\langle a^n + b^n, u^n - u\right\rangle && \text{by (13)}\\
&= t_n\big[\left\langle a^n + b, u^n - u\right\rangle + \left\langle b^n - b, u^n - u\right\rangle\big]\\
&\geq t_n\left\langle a^n + b, u^n - u\right\rangle && A \text{ is monotone}\\
&= t_n\big[\left\langle a^n + b, u^n - v^n\right\rangle + \left\langle a^n + b, v^n - u\right\rangle\big]\\
&\geq t_n\left\langle a^n + b, u^n - v^n\right\rangle. && B \text{ is monotone}
\end{aligned}$$

From this and (14) it follows that

$$\begin{aligned}\left\langle u^{n-1}-u^n, u^n-u\right\rangle + t_n^2 \left\langle b^{n-1}-b^n, b^n-b\right\rangle &\geq t_n \left\langle a^n+b, u^n-v^n\right\rangle \\ &\quad + t_n \left\langle u^n-v^n, b^n-b\right\rangle \\ &= t_n \left\langle u^n-v^n, a^n+b^n\right\rangle.\end{aligned}$$

Moreover, by (13) and (14) it holds that

$$\|u^{n-1}-u^n\|^2 + t_n^2\|b^{n-1}-b^n\|^2 = t_n^2\|a^n+b^n\|^2 + \|u^n-v^n\|^2,$$

and thus

$$\begin{aligned}\|u^{n-1}-u\|^2 + t_n^2\|b^{n-1}-b\|^2 &= \|u^{n-1}-u^n+u^n-u\|^2 + t_n^2\|b^{n-1}-b^n+b^n-b\|^2 \\ &= \|u^{n-1}-u^n\|^2 + 2\left\langle u^{n-1}-u^n, u^n-u\right\rangle + \|u^n-u\|^2 \\ &\quad + t_n^2\left[\|b^{n-1}-b^n\|^2 + 2\left\langle b^{n-1}-b^n, b^n-b\right\rangle + \|b^n-b\|^2\right] \\ &\geq t_n^2\|a^n+b^n\|^2 + \|u^n-v^n\|^2 + 2t_n\left\langle u^n-v^n, a^n+b^n\right\rangle \\ &\quad + \|u^n-u\|^2 + t_n^2\|b^n-b\|^2 \\ &= \|u^n-u\|^2 + t_n^2\|b^n-b\|^2 + \|u^n-v^n+t_n(b^n+a^n)\|^2. \qquad (16)\end{aligned}$$

We see from (16) that

$$\|u^{n-1}-u\|^2 + t_n^2\|b^{n-1}-b\|^2 \geq \|u^n-u\|^2 + t_n^2\|b^n-b\|^2,$$

and using Lemma 3.1 with $\alpha_n = \|u^n-u\|^2$, $\tau_n = t_n^2$ and $\beta_n = \|b^n-b\|^2$, we can conclude that both sequences $\{\|u^n-u\|\}$ and $\{\|b^n-b\|\}$ are bounded.

Again from (16) we can deduce using (15) that

$$\begin{aligned}\|u^{n-1}-u\|^2 + t_n^2\|b^{n-1}-b\|^2 &\geq \|u^n-u\|^2 + t_n^2\|b^n-b\|^2 + \|u^{n-1}-v^n\|^2 \\ &= \|u^n-u\|^2 + t_n^2\|b^n-b\|^2 + t_n^2\|a^n+b^{n-1}\|^2.\end{aligned} \qquad (17)$$

The first line gives

$$\begin{aligned}\|u^{n-1}-u\|^2 + t_n^2\|b^{n-1}-b\|^2 \geq \|u^n-u\|^2 &+ t_{n+1}^2\|b^n-b\|^2 + \|u^{n-1}-v^n\|^2 \\ &+ (t_n^2-t_{n+1}^2)\|b^n-b\|^2.\end{aligned}$$

Summing this inequality from $n=1$ to $n=N$, we get

$$\begin{aligned}\sum_{n=1}^N \|u^{n-1}-v^n\|^2 \leq \|u^0-u\|^2 + t_1^2\|b^0-b\|^2 - \left(\|u^N-u\|^2 + t_{N+1}^2\|b^N-b\|^2\right) \\ + \sum_{n=1}^N (t_{n+1}^2-t_n^2)\|b^n-b\|^2.\end{aligned}$$

Now, since $\|b^n-b\|^2$ is bounded and it holds that

$$\sum_{n=1}^\infty |t_n^2-t_{n+1}^2| = \sum_{n=1}^\infty |t_n-t_{n+1}||t_n+t_{n+1}| \leq 2\bar{t}\sum_{n=1}^\infty |t_n-t_{n+1}| < \infty$$

by our assumption, we can conclude that

$$\sum_{n=1}^{\infty} \|u^{n-1} - v^n\|^2 < \infty,$$

i.e., by (15), we have

$$\lim_{n\to\infty} u^{n-1} - v^n = \lim_{n\to\infty} a^n + b^{n-1} = 0.$$

This expression shows that v^n and a^n are also bounded. Due to the boundedness of $\{(u^n, b^n)\}$, we conclude the existence of weak convergence subsequences $\{u_{n_l}\}_l$ and $\{b_{n_l}\}_l$ such that

$$u^{n_l} \rightharpoonup u^*, \quad b^{n_l} \rightharpoonup b^*,$$

and by the above limits, we also have

$$v^{n_l+1} \rightharpoonup u^*, \quad a^{n_l+1} \rightharpoonup b^*.$$

From [1, Corollary 3] it follows that $(u^*, b^*) \in S(A, B)$. This shows that $\{(u^n, b^n)\}$ has a weak cluster point and that all such points are in $S(A, B)$. Now we deduce from (17) that

$$\begin{aligned} \|u^n - u^*\|^2 + (t^*)^2\|b^n - b^*\|^2 \quad &\le \|u^{n-1} - u^*\|^2 + (t^*)^2\|b^{n-1} - b^*\|^2 \\ &+|t_n^2 - (t^*)^2| \left|\|b^{n-1} - b^*\|^2 - \|b^n - b^*\|^2\right|. \end{aligned}$$

Since $\|b^n - b^*\|^2$ is bounded and $t^n \to t^*$, this shows that the sequence $\{(u^n, b^n)\}$ is quasi-Fejer convergent to the extended solution set $S(A, B)$ with respect to the distance $d((u, b), (z, w)) = \|u - z\|^2 + (t^*)^2\|b - w\|^2$. Thus, similar to the proof of [35, Theorem 1], we conclude that the whole sequence $\{(u^n, b^n)\}$ weakly converges to an element of $S(A, B)$. □

4 An adaptive step-size for DR methods

The step-size t_n suggested by (8) or by (9) is derived from our analysis of a linear case and it does not guarantee the convergence in general. In this section, we suggest modifying this step-size so that we can prove the convergence of the DR scheme. We build our adaptive step-size based on two insights:

- The estimates of the eigenvalues of the DR-iteration in the linear case from Subsection 2.1 motivated the adaptive stepsize

  $$t_n = \frac{\|u^n\|}{\|Bu^n\|}. \tag{18}$$

  This works when B is single-valued. For general cases, we construct t_n from Subsection 2.2, so that we obtain

  $$t_n = \frac{\|J_{t_{n-1}B}y^{n-1}\|}{\|y^{n-1} - J_{t_{n-1}B}y^{n-1}\|} t_{n-1}. \tag{19}$$

  Here, there is no need to evaluate Bu^n.

- Theorem 3.1 ensures the convergence of the non-stationary DR-iteration as soon as the stepsize sequence is convergent with summable increments .

However, the sequences (18) and (19) are not guaranteed to converge (and numerical experiments indicate that, indeed, divergence may occur). Here is a way to adapt the sequence (18) to produce a suitable stepsize sequence:

1. Choose $0 < t_{\min} < t_{\max} < \infty$, a summable sequence $\omega_n \in (0,1]$ with $\omega_0 = 1$ and start with $t_0 = 0$.

2. Let $\operatorname{proj}_{[\gamma,\rho]}(\cdot)$ be the projection onto a box $[\gamma,\rho]$. We construct $\{t_n\}$ as

$$t_n = (1-\omega_n)t_{n-1} + \omega_n \operatorname{proj}_{[t_{\min},t_{\max}]}\left(\frac{\|u^n\|}{\|Bu^n\|}\right). \tag{20}$$

The following lemma ensures that this will lead to a convergent sequence $\{t_n\}$.

Lemma 4.1. *Let $\{\alpha_n\}$ be a bounded sequence, i.e., $\underline{\alpha} \leq \alpha_n \leq \bar{\alpha}$, and $\{\omega_n\} \subset (0,1]$ such that $\sum_{n=0}^{\infty} \omega_n < \infty$ and $\omega_0 = 1$. Then, the sequence $\{\beta_n\}$ defined by $\beta_0 = 0$ and*

$$\beta_n = (1-\omega_n)\beta_{n-1} + \omega_n\alpha_n,$$

is in $[\underline{\alpha}, \bar{\alpha}]$ and converges to some β^* and it holds that $\sum_{n=0}^{\infty} |\beta_{n+1} - \beta_n| < \infty$.

Proof. Obviously, $\beta_0 = \alpha_0$ and since β_n is a convex combination of α_n and β_{n-1}, one can easily see that β_n obeys the same bounds as α_n, i.e. $\underline{\alpha} \leq \beta_n \leq \bar{\alpha}$. Moreover, it holds that

$$\beta_n - \beta_{n-1} = \omega_n\alpha_n + (1-\omega_n)\beta_{n-1} - \beta_{n-1} = \omega_n(\alpha_n - \beta_{n-1}),$$

thus $|\beta_n - \beta_{n-1}| \leq \omega_n(\bar{\alpha} - \underline{\alpha})$ from which the assertion follows, since ω_n is summable. □

Clearly, if we apply Lemma 4.1 to the sequence $\{t_n\}$ defined by (20), then it converges to some t^*. This is our adaptive stepsize for the case where B is single-valued. In practice, a choice $\omega_n = 2^{-\frac{(n-1)}{100}}$ works well.

If B is not single-valued, then the step-size (20) is not suitable. We use a similar trick to construct an adaptive stepsize based on the choice (19). More precisely, we construct $\{t_n\}$ as follows:

1. Choose $0 < \kappa_{\min} < \kappa_{\max} < \infty$, a summable sequence $\{\omega_n\} \subset (0,1]$, and $t_0 = 1$.

2. We construct $\{t_n\}$ as

$$\begin{aligned} \kappa_n &:= \operatorname{proj}_{[\kappa_{\min},\kappa_{\max}]}\left(\frac{\|J_{t_{n-1}B}y^{n-1}\|}{\|y^{n-1} - J_{t_{n-1}B}y^{n-1}\|}\right), \\ t_n &:= \nu_n t_{n-1}, \quad \text{where} \quad \nu_n := 1 - \omega_n + \omega_n\kappa_n. \end{aligned} \tag{21}$$

In this case we get that $t_n = \prod_{k=1}^{n} \nu_k t_0$ and since $|\nu_n - 1| = \omega_n |\kappa_n - 1|$ and κ_n is bounded, the summability of ω_n implies summability of $|\nu_n - 1|$. This implies that $\prod_{k=1}^{\infty} \nu_k$ converges to some positive value and and thus, $t_n \to t^* > 0$, too. In our implementation below, we again choose $\omega_n = 2^{-\frac{n}{100}} \in (0,1)$ and it works well.

The stepsize sequence $\{t_n\}$ constructed by either (20) or (21) fulfills the conditions of Theorem 3.1. Hence, the convergence of the nonstationary DR scheme using this adaptive stepsize follows as a direct consequence.

5 Application to ADMM

It is well-known that the alternating direction method of multipliers (ADMM) can be interpreted as the DR method on its dual problem, see, e.g. [17]. In this section, we apply our adaptive stepsize to ADMM to obtain a new variant for solving the following constrained problem:

$$\min_{u,v} \left\{ \phi(u,v) = \varphi(u) + \psi(v) \mid Du + Ev = c \right\}, \tag{22}$$

where $\varphi : \mathcal{H}_u \to \mathbb{R} \cup \{+\infty\}$, $\psi : \mathcal{H}_v \to \mathbb{R} \cup \{+\infty\}$ are two proper, closed, and convex functions, $D : \mathcal{H}_u \to \mathcal{H}$ and $E : \mathcal{H}_v \to \mathcal{H}$ are two given bounded linear operators, and $c \in \mathcal{H}$.

The dual problem associated with (22) becomes

$$\min_{x} \left\{ \varphi^*(D^T x) + \psi^*(E^T x) - c^T x \right\}, \tag{23}$$

where φ^* and ψ^* are the Fenchel conjugate of φ and ψ, respectively. The optimality condition of (23) becomes

$$0 \in \underbrace{D\partial\varphi^*(D^T x) - c}_{Ax} + \underbrace{E\partial\psi^*(E^T x)}_{Bx}, \tag{24}$$

which is of the form (1).

In the stationary case, ADMM is equivalent to the DR method applying to the dual problem (24), see, e.g., [17]. However, for the non-stationary DR method, we can derive a different parameter update rule for ADMM. Let us summarize this result into the following theorem for the non-stationary scheme (11). The proof of this theorem is given in Appendix A.

Theorem 5.1. *Given* $0 < t_{\min} < t_{\max} < +\infty$*, the ADMM scheme for solving* (22) *derived from the non-stationary DR method* (11) *applying to* (24) *becomes:*

$$\begin{cases} u^{n+1} & := \operatorname*{argmin}_u \left\{ \varphi(u) - \langle Du, w^n \rangle + \frac{t_{n-1}}{2} \|Du + Ev^n - c\|^2 \right\}, \\ v^{n+1} & := \operatorname*{argmin}_v \left\{ \psi(v) - \langle Ev, w^n \rangle + \frac{t_{n-1}}{2} \|Du^{n+1} + Ev - c\|^2 \right\}, \\ w^{n+1} & := w^n - t_{n-1}(Du^{n+1} + Ev^{n+1} - c), \\ t_n & := (1-\omega_n)t_{n-1} + \omega_n \operatorname{proj}_{[t_{\min}, t_{\max}]} \left(\frac{\|w^{n+1}\|}{\|Ev^{n+1}\|} \right), \quad \omega_n \in (0,1). \end{cases} \tag{25}$$

Consequently, the sequence $\{w^n\}$ generated by (25) *weakly converges to a solution x^* of the dual problem* (23)*.*

The ADMM variant (25) is essentially the same as the standard ADMM, but its parameter t_n is adaptively updated. This rule is different from [24, 37].

6 Numerical experiments

In this section we provide several numerical experiments to illustrate the influence of the stepsize and the adaptive choice in practical applications. Although we motivate the adaptive stepsize only for linear problems, we will apply it to problems that do not fulfill this assumption. Note that the convergence of the method is always ensured by Theorem 3.1 in both linear and nonlinear cases. We also note that the steps of the non-stationary method may be more costly than the one with constant stepsize, if the evaluation of the resolvents is costly and the constant stepsize can be leveraged to precompute something. This is the case when A and/or B is linear and the resolvents involve the solution of a linear system for which a matrix factorization can be precomputed. However, there are tricks to overcome this issue, see [8, pages 28-29], but we will not going in more detail here.

6.1 Experiments for non-stationary Douglas-Rachford

We provide four numerical examples to illustrate the new adaptive DR scheme (11) on some well-studied problems in the literature.

6.1.1 The linear toy example

We start with the linear toy example from Section 1.2 and use the adaptive stepsize (20) from Section 4 with weights $w^n = 2^{-(n-1)/100}$ and bounds $t_{\min} = 10^{-4}$, $t_{\max} = 10^4$. The residual sequence along the iterations is shown on the left of Figure 3. Additionally, we determined the stepsize t_{opt} that leads to the smallest contraction factor, i.e. to the smallest spectral radius of the iteration map $H_{t_{\text{opt}}}$ (in this case $t_{\text{opt}} = 1.367$) and also plot the corresponding residual sequence with this optimal constant stepsize in the same figure. The adaptive stepsize does indeed improve the convergence considerably both by using small steps in the beginning and automatically tuning to a stepsize t that is close to the optimal one (cf. Figure 1, right). It also outperforms the optimal constant stepsize t_{opt}.

6.1.2 LASSO problems

The LASSO problem is the minimization problem

$$\min_x \left[F(x) = \tfrac{1}{2}\|Kx - b\|_2^2 + \alpha\|x\|_1\right] \tag{26}$$

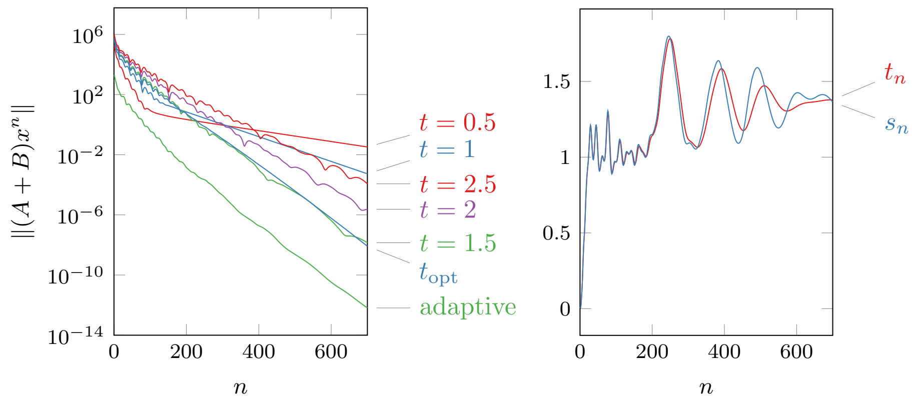

Figure 3: Results for the linear problem from Section 1.2 using fixed and the adaptive stepsizes. Left: Residual sequences. Right: Auxiliary sequence $s_n = \|u^n\|/\|Bu^n\|$ and the stepsize t_n.

and is also known as basis pursuit denoising [36]. We will treat this with the Douglas-Rachford method as follows: We set $F = f + g$ with

$$g(x) = \tfrac{1}{2}\|Kx - b\|_2^2, \qquad B = \nabla g(x) = K^T(Kx - b)$$
$$f(x) = \alpha\|x\|_1, \qquad A = \partial f(x).$$

In this particular example we take $K \in \mathbb{R}^{100\times 1000}$ with orthonormal rows, and hence, by the matrix inversion lemma, we get

$$(I + tB)^{-1}x = (I + tK^TK)^{-1}(x + tK^Tb) = (I - \tfrac{t}{t+1}K^TK)(x + tK^TB).$$

The resolvent of A is the so-called soft-thresholding operator:

$$(I + tA)^{-1}x = \max(|x| - t\alpha, 0)\,\mathrm{sign}(x).$$

Note that B is single-valued and A is a subgradient and hence, the adaptive stepsize t_n computed by (20) does apply. Figure 4 shows the result of the Douglas-Rachford iteration with constant and adaptive stepsizes, and also a comparison with the FISTA [5] method. (Note that if K would not have orthonormal rows, one would have to solve a linear system at each Douglas-Rachford step which would make the comparison with FISTA by iteration count unfair.) As shown in this plot, the adaptive stepsize again automatically tunes to a stepsize close to 10 which, experimentally, seems to be the best constant stepsize for this particular instance.

6.1.3 Convex-concave saddle-point problems

Let X and Y be two finite dimensional Hilbert spaces, $K : X \to Y$ be a bounded linear operator and $f : X \to \mathbb{R} \cup \{+\infty\}$ and $g : Y \to \mathbb{R} \cup \{+\infty\}$ be two proper,

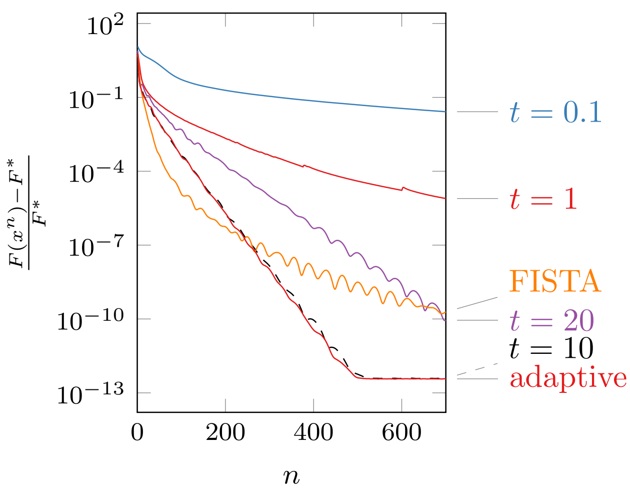

Figure 4: The convergence behavior of the Douglas-Rachford iteration and the FISTA method on a LASSO problem using fixed and the adaptive stepsizes.

convex and lower-semicontinuous functionals. The saddle point problem then reads as

$$\min_{x\in X}\max_{y\in Y}\left\{f(x) + \langle Kx, y\rangle - g(y)\right\}.$$

Saddle points (x^*, y^*) are characterized by the inclusion

$$0 \in \begin{bmatrix} \partial f & K^T \\ -K & \partial g \end{bmatrix} \begin{bmatrix} x^* \\ y^* \end{bmatrix}.$$

To apply the Douglas-Rachford method we split the optimality system as follows. We denote $z = (x, y)$ and set

$$A = \begin{bmatrix} \partial f & 0 \\ 0 & \partial g \end{bmatrix}, \qquad B = \begin{bmatrix} 0 & K^T \\ -K & 0 \end{bmatrix},$$

(cf. [29, 10]). The operator A is maximally monotone as a subgradient and B is linear and skew-symmetric, hence maximally monotone and even continuous.

One standard problem in this class in the so-called Rudin-Osher-Fatemi model for image denoising [33], also known as total variation denoising. For a given noisy image $u_0 \in \mathbb{R}^{M\times N}$ one seeks a denoising image u as the minimizer of

$$\min_u \left\{\tfrac{1}{2}\|u - u_0\|_2^2 + \lambda \||\nabla u|\|_1\right\},$$

where $\nabla u \in \mathbb{R}^{M\times N\times 2}$ denotes the discrete gradient of u and $|\nabla u|$ denotes the components-wise magnitude of this gradient. The penalty term $\||\nabla u|\|_1$ is the discretized total variation, and $\lambda > 0$ is a regularization parameter. The saddle point form of this minimization problem is

$$\min_u \max_{|\phi|\le\lambda} \left\{\tfrac{1}{2}\|u - u_0\|_2^2 + \langle \nabla u, \phi\rangle\right\}.$$

We test our DR scheme (11) using the adaptive stepsize t_n and compare with two constant stepsizes $t = 1$ and $t = 13$. The constant stepsize $t = 13$ seems to be the best among many trial stepsizes after tuning. The convergence behavior of these cases is plotted in Figure 5 for one particular image called `auge` of the size 256×256. As we can see from this figure that the adaptive stepsize has a good performance and is comparable with the best constant stepsize in this example ($t = 13$).

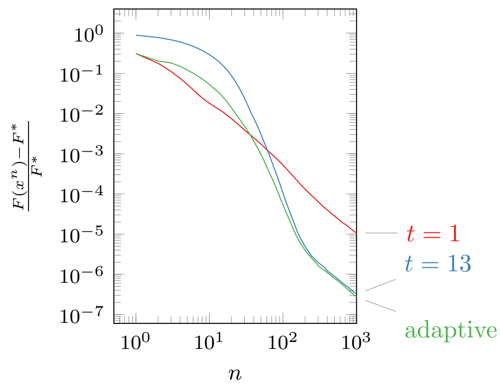

Figure 5: The decrease of the objective values of three DR variants in the total variation denoising problem.

6.1.4 Sparse inverse covariance estimation

Let us consider the well-known sparse inverse covariance estimation problem as follows:

$$\min_{X \succ 0} \left\{ F(X) = \mathrm{trace}(SX) - \log\det(X) + \alpha \|X\|_1 \right\}, \tag{27}$$

where $\alpha > 0$ is a regularization parameter, and X is a symmetric positive definite matrix variable.

Our goal is to apply the Douglas-Rachford scheme (11) to solve (27). To that end, we split the objective function into $g(U) = \mathrm{trace}(SU) - \log\det(U)$, and $h(V) = \alpha\|V\|_1$. Note that g is differentiable with its derivative

$$B(U) = \nabla g(U) = S - U^{-1},$$

while h is non-smooth. Thus, $U^+ = J_{tB}(U)$, i.e. the resolvent of B (which is the proximal operator of g) is given by the positive definite solution of

$$0 = t(S - (U^+)^{-1}) + U^+ - U,$$

which can be solved with one othogonal eigenvector decomposition of S, see, e.g., [8]. The functional h is non-smooth, but the respective proximal operator for λh with $\lambda > 0$ is the resolvent of $A = \lambda\partial h$, which is the following soft-thresholding operator:

$$\mathrm{prox}_{\lambda h}(V) = \max(|V| - \lambda, 0)\,\mathrm{sign}(V),$$

where all operations are applied component-wise.

Since B is single-valued, we can apply the Douglas-Rachford iteration in the form (2) where we use a constant and dynamic stepsize. Figure 6 shows the results for an example with an 100×100 sample covariance matrix with rank 19.

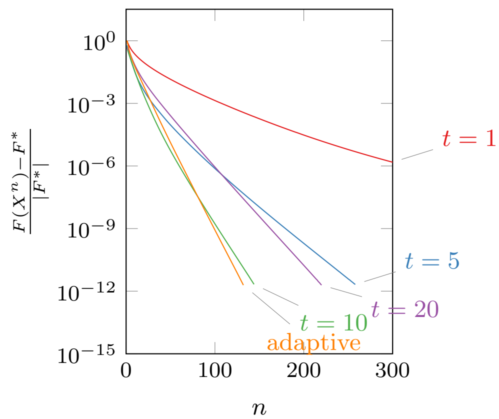

Figure 6: The relative objective residual of the sparse inverse covariance estimation obtained from the Douglas-Rachford method. It can be seen that the dynamic stepsize approaches a value close to $t = 10$, which is around the best one that could be found experimentally.

Again, the adaptive stepsize is close to the performance of an optimally chosen constant stepsize. In this particular run, this optimal constant stepsize is $t = 10$ after tuning different values of t.

6.2 Experiments for ADMM with an adaptive stepsize

In this subsection we verify the performance of the ADMM variant (25) on three existing convex optimization problems.

6.2.1 LASSO

Let us apply the ADDM scheme (25) to solve the problem LASSO problem (26) above. By duplicating the variable, we can reformulate (26) into (22) with $\varphi(u) = \alpha\|u\|_1$ and $\psi(v) = \frac{1}{2}\|Kv - b\|_2^2$ with the linear constraint $u - v = 0$. In this case, our ADMM scheme (25) becomes

$$\begin{cases} u^{n+1} &= \mathrm{prox}_{\alpha/t_{n-1}\|\cdot\|_1}(v^n + t_{n-1}^{-1}w^n) \\ v^{n+1} &= (K^TK + t_{n-1}I)^{-1}(K^Tb + t_{n-1}u^{n+1} - w^n) \\ w^{n+1} &= w^n - t_{n-1}(u^{n+1} - v^{n+1}), \\ t_n &:= (1-\omega_n)t_{n-1} + \omega_n \,\mathrm{proj}_{[t_{\min}, t_{\max}]}\left(\frac{\|w^{n+1}\|}{\|v^{n+1}\|}\right), \quad \omega_n \in (0,1). \end{cases}$$

Note that the cost to compute t_n is negligible.

We modify the code in [8] to couple with our adaptive stepsize t. We generate two problem instances of the size 1500×5000, and provide a true 300-sparse solution $x^\natural$ to generate the output $b = Kx^\natural + \mathcal{N}(0, 10^{-3})$, where $\mathcal{N}(\cdot)$ is the standard normal distribution. The first instance is a standard normal distribution with normalized columns, while the second one has 50% correlated columns.

We implement the ADMM scheme (25) using fixed t_n and adaptive t_n. In the adaptive mode, we implement also the update rule in [24] (He's t_n) using the same default setting as suggested in [8]. The convergence behavior of ADMM on these two problem instances is given in Figure 7.

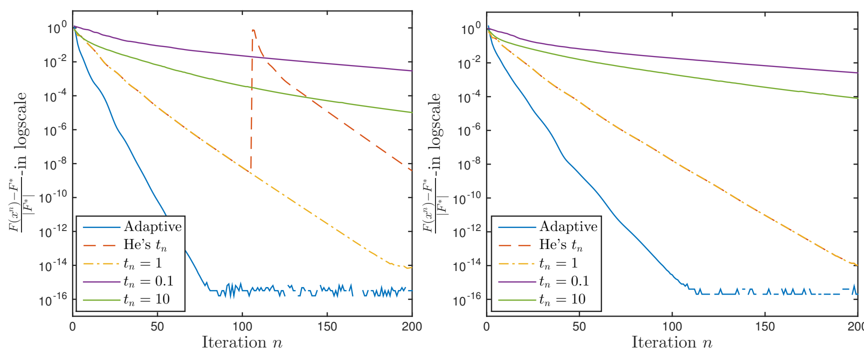

Figure 7: The relative objective residual of the LASSO using the ADMM scheme (25) with different stepsizes after 200 iterations (Left: Non-correlated data, Right: 50% correlated columns, ADMM with stepsize $t_n = 1$ and He's adaptive stepsize are identical in this case).

As we can see that for fixed stepsizes, only $t_n = 1$ produces the best result compared to $t_n = 0.1$ and $t_n = 10$. Initially, the stepsize $t_n = 1$ beats our adaptive stepsize, but it is getting slower when we run longer. He's adaptive stepsize generates a jump but then decreases the objective residual in the first instance, while it is fixed at $t = 1$ in the second instance. It works better than the $t_n = 0.1$ and $t_n = 10$ fixed stepsizes. The latter case happens due to the choice of the tuning parameters in the update formula.

6.2.2 Square-root LASSO

We test the ADMM scheme (25) on the following square-root LASSO problem:

$$F^* = \min_x \Big[F(x) = \|Kx - b\|_2 + \alpha \|x\|_1 \Big], \tag{28}$$

where $\alpha > 0$ is a regularization parameter.

We implement (25) and compare it with the standard ADMM using different fixed stepsizes. We generate the data similarly as in the LASSO problem (26), where K is a 600×2000 matrix drawn from the standard Gaussian distribution. The noise level of the observed measurement b is 10^{-3}. The regularization parameter is $\alpha = 0.1$. The convergence behavior of different ADMM variants is plotted in Figure 8 in terms of the relative objective residual $\frac{F(x^n) - F(x^*)}{|F(x^*)|}$.

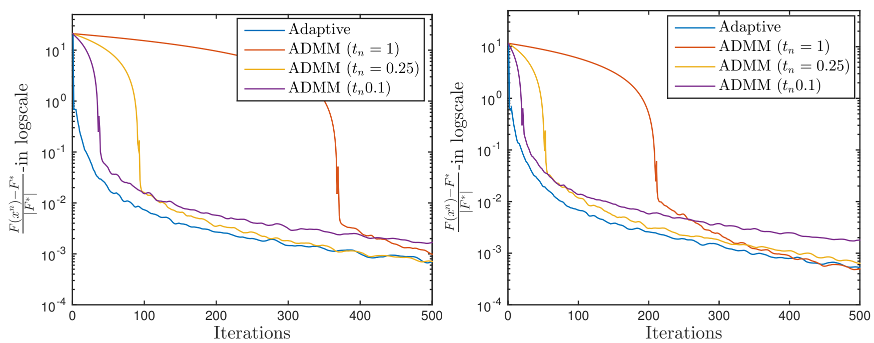

Figure 8: The objective residual of the square-root LASSO using the ADMM scheme (25) with different stepsizes after 500 iterations. Left: Non-correlated data, Right: 50% correlated columns.

Clearly, the adaptive step-size t_n leads to a better performance compared to different choices of constant step-sizes. For constant stepsizes, when t is large, we initially get a slow convergence, but then it decreases faster in the end. Note that by using some simple linear algebraic tricks, the per-iteration complexity of the ADMM scheme (25) is essentially the same as in standard ADMM methods.

6.2.3 Sparse logistic regression

Our last example is the following common ℓ_1-regularized logistic regression:

$$F^* = \min_{x:=(u,\mu)\in\mathbb{R}^{m+1}} \left\{ F(x) = \sum_{i=1}^{N} \log\left(1 + \exp(-b_i(a_i^T u + \mu)\right) + \alpha\|u\|_1 \right\}, \quad (29)$$

where $(a_i, b_i) \in \mathbb{R}^m \times \{-1, 1\}$ is the input data for $i = 1, \cdots, N$, μ is the intercept, and $\alpha > 0$ is a regularizer parameter.

By introducing an intermediate variable z, we can reformulate (29) into a constrained problem with a linear constraint $x - z = 0$. The main steps of the ADMM scheme (25) for solving (29) become

$$\begin{cases} x^{n+1} &= \operatorname*{argmin}_{x:=(u,\mu)} \left\{ \sum_{i=1}^{N} \log\left(1 + e^{-b_i(a_i^T u + \mu}\right) + \frac{t_{n-1}}{2}\|x - z^n - \rho_{n-1}^{-1} w^n\|^2 \right\} \\ z^{n+1} &= \mathcal{S}_{\alpha/t_{n-1}}(z^{n+1} - t_{n-1}^{-1} w^n) \\ w^{n+1} &= w^n - t_{n-1}(x^{n+1} - z^{n+1}) \\ t_n &= (1 - \omega_n)t_{n-1} + \omega_n \operatorname{proj}_{[t_{\min}, t_{\max}]}\left(\frac{\|w^{n+1}\|}{\|z^{n+1}\|}\right), \quad \omega_n \in (0, 1), \end{cases}$$

where $\mathcal{S}_\tau(\cdot)$ is the soft-thresholding operator (i.e., the proximal operator of the ℓ_1-norm). To solve the subproblem in x^{n+1}, we perform a Newton method to approximate its solution up to maximum 50 iterations, and a fixed tolerance 10^{-5}.

The regularization parameter α is computed following the same procedure as in [8]. We test our ADMM method (25) and compare it with the standard ADMM using fixed penalty parameter t_n, where $t_n = 1$, $t_n = 10$ and $t_n = 50$, respectively. We use two datasets: `a1a` and `w4a` downloaded from `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html`.

The convergence behavior of the two algorithmic variants on both problem instances is plotted in Figure 9 and Figure 10, respectively, for the objective value $F(x^n)$, the primal violation $\|x^n - z^n\|$, and the dual violation $t_{n-1}\|z^{n+1} - z^n\|$.

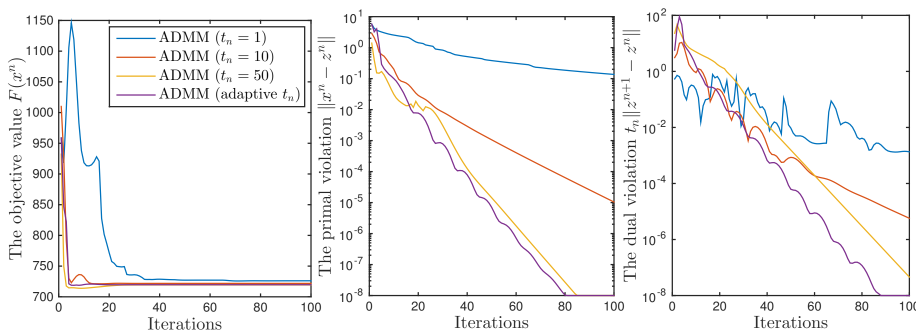

Figure 9: The convergence behavior of four ADMM variants on `a1a` after $n = 100$ iterations.

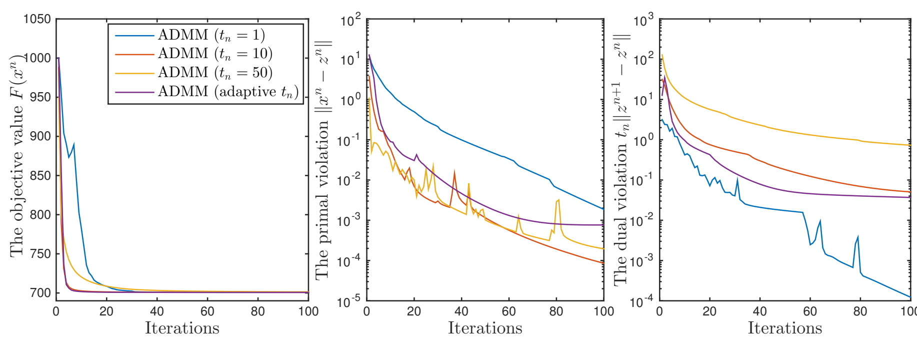

Figure 10: The convergence behavior of four ADMM variants on `w4a` after $n = 100$ iterations.

As we can see from these figures that our ADMM scheme (25) with adaptive step-size t_n performs quite well. It is competitive with the standard ADMM when we select a good penalty parameter t_n. In both cases, (25) slightly outperforms the best one over three standard ADMM variants. Note that the computational cost of t_n in (25) is negligible.

7 Conclusion

We have attempted to address one fundamental practical issue in the well-known DR method: step-size selection. This issue has been standing for a long time and has not adequately been well-understood. In this paper, we have proposed an adaptive step-size that is derived from an observation of the linear case. Our non-stationary DR method is new: it is derived from the iteration for single-valued B and differs from the standard non-stationary iteration considered previously. Our stepsize remains heuristic in the general cases, but we can guarantee a global convergence of the DR method. These two aspects, a new adaptive stepsize and convergence of a non-stationary DR scheme, are our main contribution. As a byproduct, we have also derived a new ADMM variant that uses an adaptive stepsize and has a convergence guarantee. This is practically significant since ADMM has been widely used in many areas in the last two decades. Our finding also opens some future research ideas: Although we gained some insight, the linear case is still not properly understood. Since our heuristic applies to general A and B, there is the possibility to investigate, which operators should be used as "B" to compute the adaptive stepsize. Moreover, the connection of our heuristic to other criteria such as He's stepsize or the method from [18] could be analyzed. As shown in [29], one can rescale convex-concave saddle point problems to use two different stepsizes for the Douglas-Rachford method, and one may extend our heuristic to this case. Moreover, the convergence speed of the non-stationary method under additional assumptions such as Lipschitz continuity or coercivity could be analyzed.

A The proof of Theorem 5.1

Let us assume that we apply (11) to solve the optimality condition (24) of the dual problem (23). From (11), i.e.,

$$y^{n+1} = J_{t_n A}((1+\kappa_n)J_{t_{n-1}B}y^n - \kappa_n y^n) + \kappa_n \left(y^n - J_{t_{n-1}B}y^n\right),$$

we define $w^{n+1} := J_{t_{n-1}B}y^n$ and $z^{n+1} := J_{t_n A}((1+\kappa_n)w^{n+1} - \kappa_n y^n)$ to obtain

$$\begin{cases} w^{n+1} & := J_{t_{n-1}B}y^n \\ z^{n+1} & := J_{t_n A}((1+\kappa_n)w^{n+1} - \kappa_n y^n) \\ y^{n+1} & = z^{n+1} + \kappa_n(y^n - w^{n+1}). \end{cases}$$

Shifting up this scheme by one index and changing the order, we obtain

$$\begin{cases} z^n & = J_{t_{n-1}A}((1+\kappa_{n-1})w^n - \kappa_{n-1}y^{n-1}) \\ y^n & = z^n + \kappa_{n-1}(y^{n-1} - w^n) \\ w^{n+1} & = J_{t_{n-1}B}y^n = J_{t_{n-1}B}\left(z^n + \kappa_{n-1}(y^{n-1} - w^n)\right). \end{cases}$$

Let $(1+\kappa_{n-1})w^n - \kappa_{n-1}y^{n-1} = x^n + w^n$. This gives $x^n = \kappa_{n-1}(w^n - y^{n-1})$ and hence, $z^n + \kappa_{n-1}(y^{n-1} - w^n) = z^n - x^n$ and $x^{n+1} = \kappa_n(w^{n+1} - y^n) = \kappa_n(w^{n+1} - z^n + x^n)$. Substituting these into the above expression of the DR scheme, we obtain

$$\begin{cases} z^n & = J_{t_{n-1}A}(x^n + w^n) \\ w^{n+1} & = J_{t_{n-1}B}(z^n - x^n) \\ x^{n+1} & = \kappa_n(x^n + w^{n+1} - z^n), \end{cases} \tag{30}$$

where $x^n = \kappa_{n-1}(w^n - y^{n-1})$.

From $z^n = J_{t_{n-1}A}(w^n + x^n)$, we have $z^n = (I + t_{n-1}A)^{-1}(w^n + x^n)$ or

$$0 \in z^n - w^n - x^n + t_{n-1}(D\nabla\varphi^*(D^T z^n) - c).$$

Let $u^{n+1} \in \nabla\varphi^*(D^T z^n)$, which implies $D^T z^n \in \partial\varphi(u^{n+1})$. Hence, we have $z^n - w^n - x^n + t_{n-1}(Du^{n+1} - c) = 0$, therefore $D^T z^n = D^T(w^n + x^n - t_{n-1}(Du^{n+1} - c)) \in \partial\varphi(u^{n+1})$. This condition leads to

$$0 \in D^T(t_{n-1}(Du^{n+1} - c) - x^n - w^n) + \partial\varphi(u^{n+1}).$$

This is the optimality condition of

$$u^{n+1} = \operatorname*{argmin}_u \left\{ \varphi(u) + \frac{t_{n-1}}{2}\|Du - c - t_{n-1}^{-1}(x^n + w^n)\|^2 \right\}.$$

Similarly, from $w^{n+1} = J_{t_{n-1}B}(z^n - x^n)$, if we define $v^{n+1} \in \nabla\psi^*(E^T w^{n+1})$, then we can also derive that

$$v^{n+1} = \operatorname*{argmin}_v \left\{ \psi(v) + \frac{t_{n-1}}{2}\|Ev + t_{n-1}^{-1}(x^n - z^n)\|^2 \right\}.$$

From the line $z^n - w^n - x^n + t_{n-1}(Du^{n+1} - c) = 0$ above, we can write $x^n - z^n = t_{n-1}(Du^{n+1} - c) - w^n$. Substituting this expression into the above step, we obtain

$$v^{n+1} \quad = \operatorname*{argmin}_v \left\{ \psi(v) - \langle w^n, Ev\rangle + \tfrac{t_{n-1}}{2}\|Ev + Du^{n+1} - c\|^2 \right\}.$$

This is the second line of (25).

Next, from $w^{n+1} - z^n + x^n + t_{n-1}Ev^{n+1} = 0$, we have $w^n = z^{n-1} - x^{n-1} - t_{n-2}Ev^n$. This implies $Ev^n = -t_{n-2}^{-1}(x^{n-1} + w^n - z^{n-1})$. From the last line of (30), we have $x^n = \kappa_{n-1}(x^{n-1} + w^n - z^{n-1})$. Combine these two lines, we get $Ev^n = -\frac{1}{\kappa_{n-1}t_{n-2}}x^n = -\frac{1}{t_{n-1}}x^n$ due to the update rule (9): $t_{n-1} = \kappa_{n-1}t_{n-2}$. Substituting $Ev^n = -\frac{1}{t_{n-1}}x^n$ into the u-subproblem, we obtain

$$u^{n+1} \quad = \operatorname{argmin}_u \left\{ \varphi(u) - \langle w^n, Du\rangle + \tfrac{t_{n-1}}{2}\|Du + Ev^n - c\|^2 \right\}.$$

This is the first line of (25).

Now, since $z^n = w^n - t_{n-1}(Du^{n+1} - c) + x^n$, and $w^{n+1} = z^n - x^n - t_{n-1}Ev^{n+1}$, combining these expressions, we obtain $w^{n+1} = w^n - t_{n-1}(Du^{n+1} + Ev^{n+1} - c)$. This is the last line of (25).

Finally, we derive the update rule for t_n. Indeed, note that $y^n = z^n - x^n$, and $z^n - w^n - x^n + t_{n-1}(Du^{n+1} - c) = 0$. These relations show that $y^n = w^n - t_{n-1}(Du^{n+1} - c)$. Moreover, we also have $w^{n+1} = J_{t_{n-1}B}(z^n - x^n) = J_{t_{n-1}B}(y^n)$. In this case, we have $J_{t_{n-1}B}(y^n) - y^n = w^{n+1} - w^n + t_{n-1}(Du^{n+1} - c) = -t_{n-1}(Du^{n+1} + Ev^{n+1} - c) + t_{n-1}(Du^{n+1} - c) = -t_{n-1}Ev^{n+1}$. Hence, we can compute κ_n as

$$\kappa_n := \frac{\|J_{t_{n-1}B}(y^n)\|}{\|y^n - J_{t_{n-1}B}(y^n)\|} = \frac{\|w^{n+1}\|}{t_{n-1}\|Ev^{n+1}\|}.$$

Using the fact that $t_n := \kappa_n t_{n-1}$, we show that $t_n := \frac{\|w^{n+1}\|}{\|Ev^{n+1}\|}$, which is the last line of (25) after projecting and weighting as in Section 4. Since $\{w^n\}$ is equivalent to the sequence $\{u^n\}$ in the DR scheme (2) (or equivalently, (11)) applying to the dual optimality condition (24) of the dual problem (23), the last conclusion is a direct consequence of Theorem 3.1. □

References

[1] Heinz H. Bauschke. A note on the paper by Eckstein and Svaiter on "General projective splitting methods for sums of maximal monotone operators". *SIAM Journal on Control and Optimization*, 48(4):2513–2515, 2009.

[2] Heinz H. Bauschke and Patrick L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, second edition edition, 2017.

[3] Heinz H. Bauschke, Brett Lukens, and Walaa M. Moursi. Affine nonexpansive operators, Attouch–Théra duality and the Douglas–Rachford algorithm. *Set-Valued and Variational Analysis*, 25(3):481–505, Sep 2017.

[4] Heinz H. Bauschke, Sarah M. Moffat, and Xianfu Wang. Firmly nonexpansive mappings and maximally monotone operators: Correspondence and duality. *Set-Valued and Variational Analysis*, 20(1):131–153, 2012.

[5] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[6] S. Becker, E. J. Candès, and M. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Math. Program. Compt.*, 3(3):165–218, 2011.

[7] Stephen Becker and Patrick Ll Combettes. An algorithm for splitting parallel sums of linearly composed monotone operators, with applications to signal recovery. *Arxiv preprint:1305.5828*, 2013.

[8] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[9] Kristian Bredies and Hong Peng Sun. Preconditioned Douglas–Rachford algorithms for TV-and TGV-regularized variational imaging problems. *Journal of Mathematical Imaging and Vision*, 52(3):317–344, 2015.

[10] Kristian Bredies and Hongpeng Sun. Preconditioned Douglas-Rachford splitting methods for convex-concave saddle-point problems. *SIAM Journal on Numerical Analysis*, 53(1):421–444, 2015.

[11] Kristian Bredies and Hongpeng Sun. Accelerated Douglas-Rachford methods for the solution of convex-concave saddle-point problems. *arXiv preprint arXiv:1604.06282*, 2016.

[12] Patrick L. Combettes. Solving monotone inclusions via compositions of nonexpansive averaged operators. *Optimization*, 53(5-6):475–504, 2004.

[13] Patrick L. Combettes and Jean-Christophe Pesquet. A Douglas–Rachford splitting approach to nonsmooth convex variational signal recovery. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):564–574, 2007.

[14] D. Davis. Convergence rate analysis of the forward-Douglas-Rachford splitting scheme. *SIAM Journal on Optimization*, 25(3):1760–1786, 2015.

[15] Damek Davis and Wotao Yin. Convergence rate analysis of several splitting schemes. In *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 115–163. Springer International Publishing, 2016.

[16] Jim Jr. Douglas and Henry H. Jr. Rachford. On the numerical solution of heat conduction problems in two and three space variables. *Transactions of the American mathematical Society*, 82(2):421–439, 1956.

[17] Jonathan Eckstein and Dimitri P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1):293–318, 1992.

[18] Euhanna Ghadimi, André Teixeira, Iman Shames, and Mikael Johansson. Optimal parameter selection for the alternating direction method of multipliers (ADMM): quadratic problems. *IEEE Transactions on Automatic Control*, 60(3):644–658, 2015.

[19] Pontus Giselsson. Tight global linear convergence rate bounds for Douglas–Rachford splitting. *Journal of Fixed Point Theory and Applications*, 19(4):2241–2270, Dec 2017.

[20] Pontus Giselsson and Stephen Boyd. Diagonal scaling in Douglas-Rachford splitting and ADMM. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 5033–5039. IEEE, 2014.

[21] Pontus Giselsson and Stephen Boyd. Linear convergence and metric selection for Douglas-Rachford splitting and ADMM. *IEEE Transactions on Automatic Control*, 62(2):532–544, 2017.

[22] Bingsheng He and Xiaoming Yuan. Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective. *SIAM Journal on Imaging Sciences*, 5(1):119–149, 2012.

[23] Bingsheng He and Xiaoming Yuan. On the $O(1/n)$ convergence rate of the Douglas–Rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2):700–709, 2012.

[24] BS He, Hai Yang, and SL Wang. Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and Applications*, 106(2):337–356, 2000.

[25] Xinxin Li and Xiaoming Yuan. A proximal strictly contractive Peaceman–Rachford splitting method for convex programming with applications to imaging. *SIAM Journal on Imaging Sciences*, 8(2):1332–1365, 2015.

[26] Jingwei Liang, Jalal Fadili, and Gabriel Peyré. Local convergence properties of Douglas-Rachford and alternating direction method of multipliers. *J. Optim. Theory Appl.*, 172(3):874–913, 2017.

[27] Zhouchen Lin, Risheng Liu, and Zhixun Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Advances in neural information processing systems*, pages 612–620, 2011.

[28] Pierre-Louis Lions and Bertrand Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

[29] Daniel O'Connor and Lieven Vandenberghe. Primal-dual decomposition by operator splitting and applications to image deblurring. *SIAM Journal on Imaging Sciences*, 7(3):1724–1754, 2014.

[30] Panagiotis Patrinos, Lorenzo Stella, and Alberto Bemporad. Douglas-Rachford splitting: Complexity estimates and accelerated variants. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 4234–4239. IEEE, 2014.

[31] Thomas Pock and Antonin Chambolle. Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1762–1769. IEEE, 2011.

[32] R. Tyrrell Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976.

[33] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[34] Changkyu Song, Sejong Yoon, and Vladimir Pavlovic. Fast ADMM algorithm for distributed optimization with adaptive penalty. In *AAAI*, pages 753–759, 2016.

[35] Benar Fux Svaiter. On weak convergence of the Douglas–Rachford method. *SIAM Journal on Control and Optimization*, 49(1):280–287, 2011.

[36] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[37] Zheng Xu, Mário AT Figueiredo, and Tom Goldstein. Adaptive ADMM with spectral penalty parameter selection. *arXiv preprint arXiv:1605.07246*, 2016.