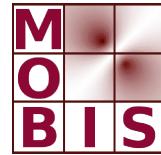
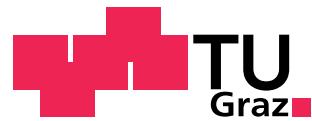




SpezialForschungsBereich F 32



Karl-Franzens Universität Graz
Technische Universität Graz
Medizinische Universität Graz



Preconditioned Douglas-Rachford algorithms for TV and TGV regularized variational imaging problems

K. Bredies H. Sun

SFB-Report No. 2014-009

June 2014

A-8010 GRAZ, HEINRICHSTRASSE 36, AUSTRIA

Supported by the
Austrian Science Fund (FWF)

FWF Der Wissenschaftsfonds.

SFB sponsors:

- Austrian Science Fund (FWF)
- University of Graz
- Graz University of Technology
- Medical University of Graz
- Government of Styria
- City of Graz



Preconditioned Douglas-Rachford algorithms for TV and TGV regularized variational imaging problems

Kristian Bredies · Hongpeng Sun

June 30, 2014

Abstract The recently introduced preconditioned Douglas-Rachford iteration (PDR) for convex-concave saddle-point problems is applied to variational imaging problems with total variation (TV) and total generalized variation (TGV) penalty. Based on PDR, we study and derive new fast iterative algorithms for TV-denoising and TV-deblurring with L^2 and L^1 discrepancy, respectively, as well as for TGV-denoising of second order, again with L^2 and L^1 discrepancy. While for denoising, symmetric (block) Red-Black Gauss-Seidel preconditioners are effective, we propose fast Fourier-transform (FFT) based preconditioners for the respective deblurring problems. For the L^2 -TGV denoising problem, we moreover derive an effective modified primal-dual gap which may be used as a stopping criterion. All algorithms are tested and compared in numerical experiments. For problems where strong convexity does not hold, it turns out that the proposed preconditioning techniques are beneficial and lead to competitive results.

Keywords Preconditioned Douglas-Rachford iteration · Primal-dual algorithms · Variational image denoising and deblurring · Total Generalized Variation · Block preconditioner

1 Introduction

In this paper, we address the problem of efficiently computing solutions of TV and TGV-regularized variational imaging problems for instances of the type

$$\min_{u \in X} F(u) + \alpha \operatorname{TV}(u), \quad \min_{u \in X} F(u) + \operatorname{TGV}_\alpha^2(u) \quad (1)$$

where X is a discrete image space, $F : X \rightarrow \mathbf{R}_\infty$ a convex fidelity functional for recovering the image u from some given data and TV and $\operatorname{TGV}_\alpha^2$ suitable discretizations of the total variation and total generalized variation semi-norm, respectively. We propose iterative algorithms which base on the recently introduced Preconditioned Douglas-Rachford (PDR) method for the solution of certain convex-concave saddle-point problems [3]. Dedicated linear preconditioners are developed which account for the differential structure of TV and TGV as well as the possible underlying structure of the image reconstruction problem.

Kristian Bredies, Hongpeng Sun

Institute for Mathematics and Scientific Computing, University of Graz, Heinrichstr. 36, A-8010 Graz, Austria
E-mail: kristian.bredies@uni-graz.at, hongpeng.sun@uni-graz.at

Even in finite dimensions, the numerical optimization of the functionals in (1) is a challenge due to the non-smooth nature of the problems. For convex optimization, first-order iterative algorithms based on the resolvent of the subgradient of a proper, convex and lower semi-continuous functional, or, equivalently, on the associated proximal mapping are an efficient and widely used choice. They are most commonly employed in conjunction with splitting methods for the underlying monotone operator inclusions. Today's state-of-the art first-order iterative solvers for (1) utilize a reformulation to such a monotone inclusion problem associated with a convex-concave saddle-point problem as well as accelerated step-size strategies, the latter where applicable. Chambolle and Pock's first order primal-dual algorithm [6] is in particular efficient for TV-regularized problems and can also be used for TGV penalty terms [1]. As it involves forward evaluations with respect to a linear operator, it is conditionally stable, i.e., step-size constraints have to be respected in order to ensure convergence. In contrast, the Douglas-Rachford splitting method is a powerful and fully implicit algorithm for the solution of monotone operator inclusions, i.e., unconditionally stable. However, it suffers from the need of solving, in each iteration step, implicit equations which can be very expensive even in an inexact manner. In [3], the authors introduced a preconditioned Douglas-Rachford splitting method which could help to deal with the linear subproblems appearing in certain convex-concave saddle-point problems. In particular, under mild assumptions one as well as finitely many inner iterations of an approximate linear solver is sufficient to establish convergence. The effectiveness of this approach for L^2 -TV denoising problems (also known as the ROF model [13]) has already been shown in [3].

In this paper, we study more TV- and TGV-regularized imaging problems, and introduce new preconditioned Douglas-Rachford splitting methods for TV-regularized problems in imaging, such as L^1 -TV denoising, L^2 -TV and L^1 -TV deblurring, which have not been discussed in [3]. The denoising problems are preconditioned by a symmetric Red-Black Gauss-Seidel method while for deblurring, fast Fourier-transform (FFT) based preconditioners are introduced. Moreover, we derive dedicated methods for TGV-regularized variational imaging problems where the underlying linear system is more complicated and difficult to solve in a traditional manner. This leads to symmetric block Red-Black Gauss-Seidel preconditioners and provides new ways to deal numerically with TGV-regularized imaging problems, in particular, L^2 -TGV and L^1 -TGV denoising. Additionally, a new primal-dual gap for L^2 -TGV denoising is derived which constitutes a provable upper bound for the functional distance to the minimum as well as an estimate for the L^2 -norm of the distance to the minimizer.

The organization of the paper is as follows. First, we briefly recall in Section 2 the abstract preconditioned Douglas-Rachford splitting methods introduced in [3]. Section 3 then deals with the application of these algorithms to certain TV-regularized variational imaging problems, mainly L^2 - and L^1 -penalized denoising and deblurring. With the exception of L^2 -TV-denoising which has already been treated in [3], this results in new iterative solution techniques. Section 4 then deals with preconditioning TGV-regularized denoising problems. After shortly reviewing TGV of second order and presenting the basic outline of the algorithms, we investigate the linear system which needs to be preconditioned during the iteration. The actual preconditioner then turns out to be a symmetric block Gauss-Seidel method on a modified equation that allows for Red-Black ordering. Discussing the associated finite-difference stencils, we give details for the implementation of the preconditioner as well as the resulting concrete methods. Stopping criteria based on the primal-dual gap are discussing in Section 5. While for TV-regularized problems, the algorithms allow for an efficient evaluation of the primal-dual gap, the situation is more difficult for TGV-regularization. We introduce a modified primal-dual gap which establishes a provable bound for the functional distance to the minimizer. In Section 6, thorough numerical tests are performed which demonstrate the efficiency of all preconditioned algorithms discussed in this paper. Finally, some discussions and conclusions are presented in the last section.

2 Abstract Preconditioned Douglas-Rachford Iterations

Let us shortly review the techniques and results from [3] for the solution of saddle-point problems of the type

$$\min_{x \in \text{dom } \mathcal{F}} \max_{y \in \text{dom } \mathcal{G}} \langle \mathcal{K}x, y \rangle + \mathcal{F}(x) - \mathcal{G}(y). \quad (2)$$

We fix X and Y as real Hilbert spaces and $\mathcal{K} : X \rightarrow Y$ as a continuous linear mapping. The functionals $\mathcal{F} : X \rightarrow \mathbf{R}_\infty$ (where $\mathbf{R}_\infty = (-\infty, +\infty]$ are the extended real numbers) and $\mathcal{G} : Y \rightarrow \mathbf{R}_\infty$ are proper, convex and lower semi-continuous. Recall that the Fenchel conjugate functionals $\mathcal{F}^* : X \rightarrow \mathbf{R}_\infty$, $\mathcal{G}^* : Y \rightarrow \mathbf{R}_\infty$, given by

$$\mathcal{F}^*(x) = \sup_{x' \in X} \langle x, x' \rangle - \mathcal{F}(x'), \quad \mathcal{G}^*(y) = \sup_{y' \in Y} \langle y, y' \rangle - \mathcal{G}(y')$$

are again proper, convex and lower semi-continuous. With these notions, (2) can be regarded as the Fenchel-Rockafellar primal-dual formulation associated with the primal and dual problem

$$\min_{x \in X} \mathcal{F}(x) + \mathcal{G}^*(\mathcal{K}x), \quad \max_{y \in Y} -\mathcal{F}^*(-\mathcal{K}^*y) - \mathcal{G}(y), \quad (3)$$

respectively [7, 11]. Under suitable conditions, primal-dual solution pairs $(x^*, y^*) \in X \times Y$ of (3) are exactly the solutions of the saddle-point problem (2).

The Preconditioned Douglas-Rachford iterations introduced in [3] come in two variants. The first, called PDR, is suitable for general \mathcal{F} and \mathcal{G} for which the resolvents can be computed while the second, called PDRQ, assumed that either the primal or dual functional is purely quadratic-linear. They both employ linear preconditioners M for the solution of the equation $Tx = b$ with T given by the problem. The preconditioner is applied by performing one step with respect to the splitting $T = M - (M - T)$, i.e.,

$$x^{\text{new}} = x^{\text{old}} + M^{-1}(b - Tx^{\text{old}}).$$

The central condition on M needed for ensuring convergence is the following.

Definition 1 Let $T, M : X \rightarrow X$ be linear, continuous symmetric and positive definite. Then, M is called a *feasible preconditioner* for T if $M - T$ is positive semi-definite.

In the PDR method, we set $T = T_{\text{PDR}} = I + \sigma^2 \mathcal{K}^* \mathcal{K}$ for some step-size $\sigma > 0$ and M a feasible preconditioner for T , resulting in the iteration outlined in Table 1. For the PDRQ method, either \mathcal{F} or \mathcal{G} is assumed to be purely quadratic-linear, here we restrict ourselves to the primal case, i.e.,

$$\mathcal{F}(x) = \frac{1}{2} \langle Qx, x \rangle - \langle f_0, x \rangle$$

for $Q : X \rightarrow X$ linear, continuous and symmetric positive semi-definite and $f_0 \in X$. In this case, M should be a feasible preconditioner for $T = T_{\text{PDRQ}} = \sigma Q + \sigma^2 \mathcal{K}^* \mathcal{K}$ for some $\sigma > 0$ (which means in particular that we assume T to be positive definite). The iteration is then performed according to Table 2 and needs less variables compared to PDR.

Convergence of both methods is summarized in the following theorem whose proof can be found in [3, Theorem 2.3].

Theorem 1 Let M be a feasible preconditioner for T_{PDR} and T_{PDRQ} , respectively. If there exists a saddle point for (2), then (PDR) and (PDRQ) converges weakly to a saddle point (x^*, y^*) of (2), respectively.

PDR	Objective:	Solve $\min_{x \in \text{dom } \mathcal{F}} \max_{y \in \text{dom } \mathcal{G}} \langle \mathcal{K}x, y \rangle + \mathcal{F}(x) - \mathcal{G}(y)$
Initialization:	$(x^0, \bar{x}^0, \bar{y}^0) \in X \times X \times Y$ initial guess, $\sigma > 0$ step-size, $T = I + \sigma^2 \mathcal{K}^* \mathcal{K}$, $M - T \geq 0$	
Iteration:		$\begin{cases} b^k = \bar{x}^k - \sigma \mathcal{K}^* \bar{y}^k, \\ x^{k+1} = x^k + M^{-1}(b^k - Tx^k), \\ y^{k+1} = \bar{y}^k + \sigma \mathcal{K}x^{k+1}, \\ \bar{x}^{k+1} = \bar{x}^k + (I + \sigma \partial \mathcal{F})^{-1}[2x^{k+1} - \bar{x}^k] - x^{k+1}, \\ \bar{y}^{k+1} = \bar{y}^k + (I + \sigma \partial \mathcal{G})^{-1}[2y^{k+1} - \bar{y}^k] - y^{k+1}, \end{cases} \quad (\text{PDR})$

Table 1: The abstract preconditioned Douglas-Rachford iteration for saddle-point problems of the type (2).

PDRQ	Solve $\min_{x \in X} \max_{y \in \text{dom } \mathcal{G}} \langle \mathcal{K}x, y \rangle + \langle \frac{1}{2}Qx - f_0, x \rangle - \mathcal{G}(y)$
Initialization:	$(x^0, \bar{y}^0) \in X \times Y$ initial guess, $\sigma > 0$ step-size, $T = \sigma Q + \sigma^2 \mathcal{K}^* \mathcal{K}$, $M - T \geq 0$
Iteration:	

$$\begin{cases} b^k = -\sigma \mathcal{K}^* \bar{y}^k + \sigma f_0 \\ x^{k+1} = x^k + M^{-1}(b^k - Tx^k) \\ y^{k+1} = \bar{y}^k + \sigma \mathcal{K}x^{k+1} \\ \bar{y}^{k+1} = \bar{y}^k + (I + \sigma \partial \mathcal{G})^{-1}[2y^{k+1} - \bar{y}^k] - y^{k+1}, \end{cases} \quad (\text{PDRQ})$$

Table 2: The abstract preconditioned Douglas-Rachford iteration for purely quadratic-linear primal functionals.

Regarding the choice of the preconditioner M from given T , the feasibility condition allows for great flexibility. The following proposition summarizes ways to construct feasible preconditioners, see [3, Propositions 2.12, 2.14 & 3.2].

Proposition 1 *Let $T : X \rightarrow X$ linear, continuous, symmetric and positive definite be given. Then, the following update schemes correspond to feasible preconditioners.*

(i) *For $M_0 : X \rightarrow X$ linear, continuous such that $M_0 - \frac{1}{2}T$ is positive definite,*

$$\begin{cases} x^{k+1/2} = x^k + M_0^{-1}(b^k - Tx^k), \\ x^{k+1} = x^{k+1/2} + M_0^{-*}(b^k - Tx^{k+1/2}). \end{cases} \quad (4)$$

(ii) *For $M : X \rightarrow X$ feasible and $n \geq 1$,*

$$\begin{cases} x^{k+(i+1)/n} = x^{k+i/n} + M^{-1}(b^k - Tx^{k+i/n}), \\ i = 0, \dots, n-1. \end{cases} \quad (5)$$

(iii) *For $T = T_1 + T_2$, with $T_1, T_2 : X \rightarrow X$ linear, continuous, symmetric, T_1 positive definite, T_2 positive semi-definite and $M : X \rightarrow X$ feasible for T_1 , $M - T_2$ boundedly invertible,*

$$\begin{cases} x^{k+1/2} = x^k + M^{-1}((b^k - T_2 x^k) - Tx^k), \\ x^{k+1} = x^k + M^{-1}((b^k - T_2 x^{k+1/2}) - Tx^k). \end{cases} \quad (6)$$

Preconditioner	T	λI	$(\lambda + 1)D$	M_{SGS}	M_{SSOR}
Conditions	—	$\lambda \geq \ T\ $	$\lambda \geq \lambda_{\max}(T - D)$	—	$\omega \in (0, 2)$
Iteration type	Douglas-Rachford	Richardson	Damped Jacobi	Symmetric Gauss-Seidel	Symmetric SOR

$D = \text{diag}(T)$, $T = D - E - E^*$, E lower triangular, $M_{SGS} = (D - E)(T + D)^{-1}(D - E^*)$,
 $M_{SSOR} = (\frac{1}{\omega}D - E)(T + \frac{2-\omega}{\omega}D)^{-1}(\frac{1}{\omega} - E^*)$

Table 3: Summary of common preconditioners and possible conditions for feasibility.

The update scheme (4) is useful if one has a non-symmetric “feasible” preconditioner M_0 for $\frac{1}{2}T$, then the concatenation with the adjoint preconditioner will be symmetric and feasible. Likewise, (5) corresponds to the n -fold application of a feasible preconditioner which is again feasible. Finally, (6) is useful if T can be split into $T_1 + T_2$ for which T_1 can be easily preconditioned by M . Then, one gets a feasible preconditioner for T only using forward evaluation of T_2 .

Several standard preconditioners can be seen to be feasible, including symmetric Gauss-Seidel and SSOR. Table 3 provides an overview; for details, we refer to [3].

3 TV-Regularized Denoising and Deblurring

We start with preconditioning strategies for discrete TV-regularized problems, i.e.,

$$\min_{u \in U} F(u) + \alpha \text{TV}(u) = \min_{u \in U} F(u) + \alpha \|\nabla u\|_1.$$

Here, U is a discrete image space and ∇ a discrete gradient operator which will be described in the following. Following essentially the presentation in [1, 5], consider the image domain $\Omega \subset \mathbf{Z}^2$ as the discretized grid

$$\Omega = \{(i, j) \mid i, j \in \mathbb{N}, 0 \leq i \leq N_x - 1, 0 \leq j \leq N_y - 1\}$$

where N_x, N_y are the image dimensions. In order not having to distinguish cases in the discretization, we assume $N_x \geq 3$ and $N_y \geq 3$. We define the image space as $U = \{u : \Omega \rightarrow \mathbf{R}\}$ with the standard L^2 scalar product. Finite differences are used to discretize the operator ∇ and its adjoint operator $\nabla^* = -\text{div}$ with homogeneous Neumann and Dirichlet boundary conditions, respectively. We define ∇ as the following operator

$$(\nabla u) = \begin{pmatrix} \partial_x^+ u \\ \partial_y^+ u \end{pmatrix},$$

where forward differences are taken according to

$$(\partial_x^+ u)_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j}, & \text{if } 0 \leq i < N_x - 1, \\ 0, & \text{if } i = N_x - 1, \end{cases} \quad (\partial_y^+ u)_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j}, & \text{if } 0 \leq j < N_y - 1, \\ 0, & \text{if } j = N_y - 1. \end{cases}$$

With $V = U^2$ with the standard product scalar product, this gives a linear operator $\nabla : U \rightarrow V$. The discrete divergence is then the negative adjoint of ∇ , i.e., the unique linear mapping $\text{div} : V \rightarrow U$ which satisfies

$$\langle \nabla u, p \rangle_V = \langle u, \nabla^* p \rangle_U = -\langle u, \text{div } p \rangle_U, \quad \forall u \in U, p \in V.$$

It can be computed to read as

$$\text{div } p = \partial_x^- p^1 + \partial_y^- p^2$$

involving the backward difference operators

$$(\partial_x^- u)_{i,j} = \begin{cases} u_{0,j}, & \text{if } i = 0, \\ u_{i,j} - u_{i-1,j}, & \text{if } 0 < i < N_x - 1, \\ -u_{N_x-1,j}, & \text{if } i = N_x - 1, \end{cases} \quad (\partial_y^- u)_{i,j} = \begin{cases} u_{i,0}, & \text{if } j = 0, \\ u_{i,j} - u_{i,j-1}, & \text{if } 0 < j < N_y - 1, \\ -u_{i,N_y-1}, & \text{if } j = N_y - 1. \end{cases}$$

In order to define the discrete version of the TV (and later the TGV) functionals, we still need the discrete versions of the L^1 , L^2 and L^∞ norms: For $u \in U$, $p = (p^1, p^2) \in V$, $1 \leq t < \infty$,

$$\begin{aligned} \|u\|_t &= \left(\sum_{(i,j) \in \Omega} |u_{i,j}|^t \right)^{1/t}, \quad \|u\|_\infty = \max_{(i,j) \in \Omega} |u_{i,j}|, \\ \|p\|_t &= \left(\sum_{(i,j) \in \Omega} ((p_{i,j}^1)^2 + (p_{i,j}^2)^2)^{t/2} \right)^{1/t}, \quad \|p\|_\infty = \max_{(i,j) \in \Omega} \sqrt{(p_{i,j}^1)^2 + (p_{i,j}^2)^2}. \end{aligned}$$

Further, problem-specific discrete concepts will be introduced later in this section.

3.1 TV-Regularized Denoising Problems

Let us first apply the discrete framework for the total-variation regularized L^2 - and L^1 -type denoising problems (see [13] for the L^2 case which is usually called the ROF model)

$$\min_{u \in U} F(u) + \alpha \|\nabla u\|_1, \quad F(u) = \begin{cases} \frac{1}{2} \|u - f\|_2^2 & \text{for } L^2\text{-denoising,} \\ \|u - f\|_1 & \text{for } L^1\text{-denoising,} \end{cases} \quad (7)$$

with $f : \Omega \rightarrow \mathbf{R}$ a given noisy image and $\alpha > 0$ a regularization parameter. As we are in finite dimensions, both F and $\alpha \|\cdot\|_1$ are continuous, so one can employ Fenchel-Rockafellar duality to obtain a equivalent saddle-point problem [7, 11] of type (2) with

$$X = U, \quad Y = V, \quad \mathcal{F} = F, \quad \mathcal{K} = \nabla \quad \text{and} \quad \mathcal{G} = \mathcal{I}_{\{\|v\|_\infty \leq \alpha\}}$$

while \mathcal{I}_C denotes the indicator function of the set C , i.e.,

$$\mathcal{I}_C(x) = \begin{cases} 0 & \text{if } x \in C, \\ \infty & \text{else.} \end{cases} \quad (8)$$

With these prerequisites, preconditioned Douglas-Rachford methods can be derived.

3.1.1 The L^2 -Denoising Case

Following [3], in the case of quadratic discrepancy, F is purely quadratic-linear (up to a constant), so one can use the iteration (PDRQ), i.e., the data

$$\mathcal{K} = \nabla, \quad Q = I, \quad f_0 = f, \quad \mathcal{G} = \mathcal{I}_{\{\|p\|_\infty \leq \alpha\}}.$$

The operator M should be a feasible preconditioner for $\sigma I - \sigma^2 \operatorname{div} \nabla = \sigma I - \sigma^2 \Delta$ (where $\Delta = \operatorname{div} \nabla$) and will be discussed in Subsection 3.1.3. The iteration then needs the resolvent $(I + \sigma \partial \mathcal{G})^{-1} = \mathcal{P}_\alpha$ which reads as [1]

$$\mathcal{P}_\alpha(p) = (I + \sigma \partial \mathcal{G})^{-1}(p) = \arg \min_{p' \in V} \frac{1}{2} \|p' - p\|_2^2 + \mathcal{I}_{\{\|p\|_\infty \leq \alpha\}}(p') = \frac{p}{\max(1, |p|/\alpha)} \quad (9)$$

with $|p| = \sqrt{(p^1)^2 + (p^2)^2}$ in the pointwise sense.

3.1.2 The L^1 -Denoising Case

Here, one may use the iteration (PDR). It turns out to be beneficial to scale the gradient with a factor $\tau > 0$, leading to the following data:

$$\mathcal{K} = \tau \nabla, \quad \mathcal{F} = \|\cdot - f\|_1, \quad \mathcal{G} = \mathcal{I}_{\{\|p\|_\infty \leq \frac{\alpha}{\tau}\}}.$$

Analogously, M should be a preconditioner for $I - \sigma^2 \Delta$. The resolvent of ∂F is moreover involved which is known to realize a soft-thresholding (or shrinkage) operation around f , i.e., $(I + \sigma \partial \mathcal{F})^{-1} = \mathcal{S}_\sigma(\cdot, f)$ with

$$\begin{aligned} (I + \sigma \partial \mathcal{F})^{-1}(u) &= \arg \min_{u' \in U} \frac{1}{2} \|u' - u\|_2^2 + \sigma \|u' - f\|_1 \\ &= \mathcal{S}_\sigma(u, f) = f + \text{sign}(u - f) \max(0, |u - f| - \sigma). \end{aligned} \quad (10)$$

The respective resolvent for $\partial \mathcal{G}$ is again given by (9).

Remark 1 Alternatively, (PDRQ) may also be employed by introducing an additional dual variable $v \in U$ and writing $\|u - f\|_1 = \max_{v \in U} \langle u, v \rangle - (\mathcal{I}_{\{\|v\|_\infty \leq 1\}}(v) + \langle f, v \rangle)$, resulting in the dual space $Y = V \times U$, operator $\mathcal{K}u = (\nabla u, u)$, $\mathcal{F} = 0$ and

$$\mathcal{G}(y) = \mathcal{G}(p, v) = G_1(p) + G_2(v), \quad G_1 = \mathcal{I}_{\{\|p\|_\infty \leq \alpha\}}, \quad G_2 = \mathcal{I}_{\{\|v\|_\infty \leq 1\}} + \langle f, \cdot \rangle.$$

The proximal mapping for the dual functional can be computed separately for p and v . For G_1 , this is again given by (9), while for G_2 , we have

$$\begin{aligned} (I + \sigma \partial G_2)^{-1}(v) &= \arg \min_{v' \in U} \frac{1}{2} \|v' - v\|_2^2 + \sigma \langle f, v' \rangle + \mathcal{I}_{\{\|v\|_\infty \leq 1\}}(v') \\ &= \max(-1, \min(1, v - \sigma f)). \end{aligned} \quad (11)$$

We will compare, in Section 6, the performance of both variants.

3.1.3 The Preconditioner

In both the L^2 and L^1 case, M is required to be a feasible preconditioner for operators of type $T = \lambda I - \mu \Delta$ for $\lambda, \mu > 0$ where $\Delta = \text{div } \nabla$ can be interpreted as a discrete Laplace operator with homogeneous Neumann boundary conditions [15]. In other words: $Tu = b$ corresponds to a discrete version of the boundary value problem

$$\begin{cases} \lambda u - \mu \Delta u = b & \text{in } \Omega, \\ \frac{\partial u}{\partial \nu} = 0 & \text{on } \partial \Omega. \end{cases} \quad (12)$$

With the choice of ∇u , we get a finite-difference equation with five-point stencils according to Table 4. As in [3], we will use a symmetric Gauss-Seidel update in conjunction with a Red-Black enumeration scheme as follows. Let

$$\Omega_{\text{red}} = \{(i, j) \in \Omega \mid i + j \text{ even}\}, \quad \Omega_{\text{black}} = \{(i, j) \in \Omega \mid i + j \text{ odd}\},$$

and write $u = (u_{\text{red}}, u_{\text{black}})$ where u_{red} and u_{black} represent the image u on Ω_{red} and Ω_{black} , respectively. A Gauss-Seidel update then performs an update according to red \rightarrow black while its adjoint corresponds to the reverse order black \rightarrow red. One can easily see that a update of red and black pixels, respectively, is idempotent. Thus, n steps of symmetric Red-Black Gauss-Seidel,

$\begin{bmatrix} \lambda + 2\mu & -\mu \\ -\mu & \end{bmatrix}$	$\begin{bmatrix} -\mu & \lambda + 3\mu & -\mu \\ -\mu & \end{bmatrix}$	$\begin{bmatrix} -\mu & \lambda + 2\mu \\ -\mu & \end{bmatrix}$
$=\mathcal{S}_{NW}$	$=\mathcal{S}_N$	$=\mathcal{S}_{NE}$
$\begin{bmatrix} -\mu & \lambda + 3\mu & -\mu \\ -\mu & \end{bmatrix}$	$\begin{bmatrix} -\mu & \lambda + 4\mu & -\mu \\ -\mu & \end{bmatrix}$	$\begin{bmatrix} -\mu & \lambda + 3\mu \\ -\mu & \end{bmatrix}$
$=\mathcal{S}_W$	$=\mathcal{S}_O$	$=\mathcal{S}_E$
$\begin{bmatrix} -\mu & \lambda + 2\mu & -\mu \\ -\mu & \end{bmatrix}$	$\begin{bmatrix} -\mu & \lambda + 3\mu & -\mu \\ -\mu & \end{bmatrix}$	$\begin{bmatrix} -\mu & \lambda + 2\mu \\ -\mu & \end{bmatrix}$
$=\mathcal{S}_{SW}$	$=\mathcal{S}_S$	$=\mathcal{S}_{SE}$

Table 4: Finite-difference stencils for $\lambda u - \mu \Delta u$ with homogeneous Neumann boundary conditions. The highlighted entry denotes the center element and u is assumed to be extended by arbitrary values outside of Ω .

denoted by $\text{SRBGS}_{\lambda,\mu}^n$, corresponds to the following procedure. Denoting $\mathcal{N}(i,j) = \{(i',j') \in \Omega \mid |i - i'| + |j - j'| = 1\}$ the set of neighbor points of (i,j) in Ω and $c_{i,j} = \#\mathcal{N}(i,j)$ (which is either 2, 3 or 4, depending on whether (i,j) is a corner, edge or interior point), we have

$$\left\{ \begin{array}{l} \text{SRBGS}_{\lambda,\mu}^n(u^k, b^k) = (u_{\text{red}}^{k+1}, u_{\text{black}}^{k+1}), \\ (u_{\text{red}}^{k+(\nu+1/2)/n})_{i,j} = \frac{1}{\lambda + c_{i,j}\mu} (b_{i,j}^k + \mu \sum_{(i',j') \in \mathcal{N}(i,j)} (u_{\text{black}}^{k+\nu/n})_{i',j'}) \quad (i,j) \in \Omega_{\text{red}}, \\ (u_{\text{black}}^{k+(\nu+1)/n})_{i,j} = \frac{1}{\lambda + c_{i,j}\mu} (b_{i,j}^k + \mu \sum_{(i',j') \in \mathcal{N}(i,j)} (u_{\text{red}}^{k+(\nu+1/2)/n})_{i',j'}) \quad (i,j) \in \Omega_{\text{black}}, \\ \nu = 0, \dots, n-1, \\ (u_{\text{red}}^{k+1})_{i,j} = \frac{1}{\lambda + c_{i,j}\mu} (b_{i,j}^k + \mu \sum_{(i',j') \in \mathcal{N}(i,j)} (u_{\text{black}}^{k+1})_{i',j'}) \quad (i,j) \in \Omega_{\text{red}}. \end{array} \right. \quad (13)$$

According to Table 3, $\text{SRBGS}_{\lambda,\mu}^1$ corresponds to a feasible preconditioner as it realizes a symmetric Gauss-Seidel step and of course, n -fold application $\text{SRBGS}_{\lambda,\mu}^n$, is also feasible, see Proposition 1.

3.1.4 The Algorithms

Putting the building blocks together, we get a PDRQ method for L^2 -TV denoising according to Table 5 (which was already derived in [3]) as well as a PDR and PDRQ method for L^1 -TV denoising according to Tables 6 and 7, respectively. As U and V are finite-dimensional, all algorithms converge to a respective solution by virtue of Theorem 1 for any choice of $\sigma > 0$ and $\tau > 0$.

Remark 2 For a specific choice of the step-size σ and scaling τ , note that T is corresponding to the Neumann problem (12), which possesses a unique solution as $\lambda > 0$. However, for large μ/λ , the problem approximates a pure Neumann problem which requires compatibility conditions on b in order to be solvable. Numerically, the problem is ill-posed in this situation making the preconditioner less effective. Therefore, μ/λ should not be chosen too large. On the other hand,

PDRQ Objective: L^2 -TV denoising		$\min_{u \in U} \frac{1}{2} \ u - f\ _2^2 + \alpha \ \nabla u\ _1$
Initialization:	$(u^0, \bar{p}^0) \in U \times V$ initial guess, $\sigma > 0$ step-size, $n \geq 1$ inner iterations for symmetric Gauss-Seidel	
Iteration:	$u^{k+1} = \text{SRBGS}_{\sigma, \sigma^2}^n(u^k, \sigma(f + \text{div } \bar{p}^k))$ $p^{k+1} = \bar{p}^k + \sigma \nabla u^{k+1}$ $p_{\text{test}}^{k+1} = \mathcal{P}_\alpha(2p^{k+1} - \bar{p}^k)$ $\bar{p}^{k+1} = \bar{p}^k + p_{\text{test}}^{k+1} - p^{k+1}$	according to (13) according to (9)

Table 5: The preconditioned Douglas-Rachford iteration for L^2 -TV denoising.

PDR Objective: L^1 -TV denoising		$\min_{u \in U} \ u - f\ _1 + \alpha \ \nabla u\ _1$
Initialization:	$(u^0, \bar{u}^0, \bar{p}^0) \in U \times U \times V$ initial guess, $\sigma > 0$ step-size, $\tau > 0$ gradient scaling, $n \geq 1$ inner iterations for symmetric Gauss-Seidel	
Iteration:	$u^{k+1} = \text{SRBGS}_{1, (\sigma\tau)_2}^n(u^k, \bar{u}^k + \sigma\tau \text{div } \bar{p}^k)$ $p^{k+1} = \bar{p}^k + \sigma\tau \nabla u^{k+1}$ $u_{\text{test}}^{k+1} = \mathcal{S}_\sigma(2u^{k+1} - \bar{u}^k, f)$ $\bar{u}^{k+1} = \bar{u}^k + u_{\text{test}}^{k+1} - u^{k+1}$ $p_{\text{test}}^{k+1} = \mathcal{P}_{\alpha/\tau}(2p^{k+1} - \bar{p}^k)$ $\bar{p}^{k+1} = \bar{p}^k + p_{\text{test}}^{k+1} - p^{k+1}$	according to (13) according to (10) according to (9)

Table 6: The scaled preconditioned Douglas-Rachford iteration for L^1 -TV denoising.

PDRQ Objective: L^1 -TV denoising		$\min_{u \in U} \ u - f\ _1 + \alpha \ \nabla u\ _1$
Initialization:	$(u^0, \bar{p}^0, \bar{v}^0) \in U \times V \times U$ initial guess, $\sigma > 0$ step-size, $n \geq 1$ inner iterations for symmetric Gauss-Seidel	
Iteration:	$u^{k+1} = \text{SRBGS}_{\sigma^2, \sigma^2}^n(u^k, -\sigma\bar{v}^k + \sigma \text{div } \bar{p}^k)$ $p^{k+1} = \bar{p}^k + \sigma \nabla u^{k+1}$ $v^{k+1} = \bar{v}^k + \sigma u^{k+1}$ $p_{\text{test}}^{k+1} = \mathcal{P}_\alpha(2p^{k+1} - \bar{p}^k)$ $\bar{p}^{k+1} = \bar{p}^k + p_{\text{test}}^{k+1} - p^{k+1}$ $v_{\text{test}}^{k+1} = \max(-1, \min(1, 2v^{k+1} - \bar{v}^k - \sigma f))$ $\bar{v}^{k+1} = \bar{v}^k + v_{\text{test}}^{k+1} - v^{k+1}$	according to (13) according to (9) according to (11)

Table 7: The PDRQ variant for L^1 -TV denoising.

if μ/λ is too small, solving (12) is close to the inversion of the identity (up to a factor), hence the update step for u does not contribute much to the solution of the optimization problem. Therefore, μ/λ should not be chosen too low.

For the L^2 -TV-denoising problem these considerations suggest to take σ such that $1 + \sigma \|\Delta\|$ is moderate. Likewise, for L^1 -TV-denoising, it is reasonable to choose $\tau = c/\sigma$ with moderate $1 + c^2 \|\Delta\|$ for the PDR method and, in view of (9) and (10), to choose σ neither too large nor too small. For examples on how σ and τ might be chosen in order to make the methods efficient, we refer to Section 6, where numerical experiments are presented.

3.2 TV-Deblurring Problems

Next, the preconditioned Douglas-Rachford framework is applied to ill-posed linear inverse problems. We aim at applying it to deblurring problems of the type

$$\min_{u \in U} F(u) + \alpha \|\nabla u\|_1, \quad F(u) = \begin{cases} \frac{1}{2} \|u * \kappa - f\|_2^2 & \text{for } L^2\text{-deblurring,} \\ \|u * \kappa - f\|_1 & \text{for } L^1\text{-deblurring.} \end{cases} \quad (14)$$

Here, $\kappa : \Omega \rightarrow \mathbf{R}$ is a given convolution kernel satisfying $\sum_{i,j} \kappa_{i,j} \neq 0$. For sake of simplicity, we choose to use the *periodic convolution*, i.e., $u * \kappa$ denotes

$$(u * \kappa)_{i,j} = \sum_{i'=0}^{N_x-1} \sum_{j'=0}^{N_y-1} u_{i-i' \bmod N_x, j-j' \bmod N_y} \kappa_{i',j'}$$

with $a \bmod b$ being the unique number $c \in \{0, \dots, b-1\}$ such that there is a $d \in \mathbf{Z}$ with $a = db+c$. We will refer to the periodic convolution operator with κ as L and also introduce the periodic gradient $\nabla_p : U \rightarrow V$ according to

$$\nabla_p u = \begin{pmatrix} \partial_x^{p+} u \\ \partial_y^{p+} u \end{pmatrix}, \quad (\partial_x^{p+} u)_{i,j} = u_{i+1 \bmod N_x, j} - u_{i,j}, \quad (\partial_y^{p+} u)_{i,j} = u_{i,j+1 \bmod N_y} - u_{i,j}$$

for $(i,j) \in \Omega$ as well as the periodic divergence $\operatorname{div}_p = -\nabla_p^*$ and Laplacian $\Delta_p = \operatorname{div}_p \nabla_p$. Let us again discuss the preconditioned Douglas-Rachford iterations for L^2 - and L^1 -discrepancies separately.

3.2.1 The L^2 -Deblurring Case

The discrepancy functional $\mathcal{F}(u) = \frac{1}{2} \|u * \kappa - f\|_2^2$ in (14) can be rewritten to be quadratic-linear, hence, we employ again (PDRQ). Using the scaled gradient with a factor $\tau > 0$ data then reads as

$$\mathcal{K} = \tau \nabla, \quad Q = L^* L, \quad f_0 = L^* f, \quad \mathcal{G} = \mathcal{I}_{\{\|p\|_\infty \leq \alpha/\tau\}}.$$

Recall that L denotes the convolution operator with respect to κ , i.e., $Lu = u * \kappa$ for all $u \in U$. Let us choose a feasible preconditioner for $T = \sigma L^* L - (\sigma\tau)^2 \Delta$. One easily sees that $M = \sigma L^* L - (\sigma\tau)^2 \Delta_p$ satisfies $M - T \geq 0$ as

$$\langle (M - T)u, u \rangle = \langle -\Delta_p u, u \rangle - \langle -\Delta u, u \rangle = \|\nabla_p u\|^2 - \|\nabla u\|^2 \geq 0 \quad \text{for each } u \in U,$$

where the latter inequality is due to the fact that $\partial_x^+ u$ is just $\partial_x^{p+} u$ with some entries replaced by 0 (and the analog for $\partial_y^+ u$). Since L^* is the periodic convolution with the mirrored kernel, M is the periodic convolution with the kernel κ_M according to

$$\kappa_M = \sigma \kappa' * \kappa + (\sigma\tau)^2 \kappa_\Delta, \quad \begin{cases} \kappa'_{i,j} = \kappa_{N_x-i-1, N_y-j-1}, \\ (i,j) \in \Omega \end{cases} \quad \kappa_\Delta = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (15)$$

where the highlighted entry corresponds to $i = j = 0$. Hence, M is diagonalized by the discrete Fourier transform \mathcal{F} and can be inverted by

$$M^{-1}u = \mathcal{F}^{-1}\left(\frac{1}{\mathcal{F}\kappa_M}(\mathcal{F}u)\right) = \mathcal{F}^{-1}\left(\frac{\mathcal{F}u}{\sigma|\mathcal{F}\kappa|^2 + (\sigma\tau)^2 \mathcal{F}\kappa_\Delta}\right) \quad (16)$$

PDRQ Objective: L^2 -TV deblurring		$\min_{u \in U} \frac{1}{2} \ u * \kappa - f\ _2^2 + \alpha \ \nabla u\ _1$
Initialization:	$(u^0, \bar{p}^0) \in U \times V$ initial guess, $\sigma > 0$ step-size, $\tau > 0$ gradient scaling	
Iteration:	$u^{k+1} = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(f * \kappa') + \mathcal{F}(\tau \operatorname{div} \bar{p}^k + \sigma \tau^2 (\Delta - \Delta_p) u^k)}{ \mathcal{F}\kappa ^2 + \sigma \tau^2 \mathcal{F}\kappa_\Delta} \right)$	according to (15)
	$\bar{p}^{k+1} = \bar{p}^k + \sigma \tau \nabla u^{k+1}$	
	$p_{\text{test}}^{k+1} = \mathcal{P}_{\alpha/\tau}(2p^{k+1} - \bar{p}^k)$	according to (9)
	$\bar{p}^{k+1} = \bar{p}^k + p_{\text{test}}^{k+1} - p^{k+1}$	

Table 8: The scaled preconditioned Douglas-Rachford iteration for the solution of the L^2 -TV deblurring problem.

provided that the denominator is positive in each point. This is however the case as $(\mathcal{F}\kappa_\Delta)_{i,j} = 8 \sin^2(i\pi/N_x) + 8 \sin^2(j\pi/N_y)$ vanishes only for $i = j = 0$ and since $(\mathcal{F}\kappa)_{0,0} = \sum_{i,j} \kappa_{i,j} \neq 0$ by assumption. In particular, M is positive definite, leading to its feasibility for T .

Putting all together yields the PDRQ method for the solution of the L^2 -TV-deblurring problem, see Table 8. Of course, an implementation should precompute and store $f * \kappa'$ as well as $|\mathcal{F}\kappa|^2 + \sigma \tau^2 \mathcal{F}\kappa_\Delta$ and utilize the Fast Fourier transform (FFT) [1, 6] in order to gain maximal efficiency.

3.2.2 The L^1 -Deblurring Case

This case is slightly different from L^1 -TV-denoising as the proximal mapping associated with $\|L \cdot - f\|_1$ is not easy to compute. We therefore introduce another dual variable $v \in U$ and obtain (again, with a gradient scaling $\tau > 0$) the saddle-point formulation

$$\min_{u \in U} \max_{\substack{p \in V \\ v \in U}} \langle \tau \nabla u, p \rangle + \langle L u, v \rangle - G_1(p) - G_2(v). \quad (17)$$

$G_1 = \mathcal{I}_{\{\|p\|_\infty \leq \alpha/\tau\}}$ and $G_2 = \mathcal{I}_{\{\|v\|_\infty \leq 1\}} + \langle f, \cdot \rangle$ (also see Remark 1). For (PDRQ), this leads to the data

$$X = U, \quad Y = V \times U, \quad \mathcal{K} = \begin{bmatrix} \nabla \\ L \end{bmatrix}, \quad Q = 0, \quad f_0 = 0, \quad \mathcal{G}(y) = \mathcal{G}(p, v) = G_1(p) + G_2(v),$$

and we have to choose a preconditioner for $T = \sigma^2 \mathcal{K}^* \mathcal{K} = \sigma^2 L^* L - (\sigma \tau)^2 \Delta$. Again, $M = \sigma^2 L^* L - (\sigma \tau)^2 \Delta_p$ is feasible and can be computed as in (16) with a straightforward modification.

This results in the PDRQ algorithm for L^1 -TV-deblurring which can be found in Table 9. This version assumes that it is beneficial to compute the convolution via pointwise multiplication on the Fourier side, hence the transformed variable \hat{u}^k has been introduced. Again, one should consider precomputing $\mathcal{F}\kappa$ and $\sigma |\mathcal{F}\kappa|^2 + \sigma \tau^2 \mathcal{F}\kappa_\Delta$ as well as using the Fast Fourier transform in order to obtain maximal efficiency.

4 TGV-Regularized Denoising Problems

Having discussed the derivation of preconditioned Douglas-Rachford algorithms for TV regularization, we now turn to developing preconditioned iterative methods for total generalized variation (TGV) regularization, for simplicity of second order. The concept of TGV, introduced in [5], realizes a convex regularizing functional which automatically adapts to higher-order smoothness

PDRQ Objective: L^1 -TV deblurring		$\min_{u \in U} \ u * \kappa - f\ _1 + \alpha \ \nabla u\ _1$
Initialization:	$(u^0, \bar{p}^0, \bar{v}^0) \in U \times V \times U$ initial guess, $\hat{u}^0 = \mathcal{F}u^0$, $\sigma > 0$ step-size, $\tau > 0$ gradient scaling	
Iteration:	$\hat{u}^{k+1} = \frac{\mathcal{F}(\tau \operatorname{div} \bar{p}^k + \sigma \tau^2 (\Delta - \Delta_p) u^k) - \mathcal{F}\bar{v}^k \mathcal{F}\kappa}{\sigma \mathcal{F}\kappa ^2 + \sigma \tau^2 \mathcal{F}\kappa \Delta}$ $u^{k+1} = \mathcal{F}^{-1} \hat{u}^{k+1}$ $p^{k+1} = \bar{p}^k + \sigma \tau \nabla u^{k+1}$ $p_{\text{test}}^{k+1} = \mathcal{P}_{\alpha/\tau}(2p^{k+1} - \bar{p}^k)$ $\bar{p}^{k+1} = \bar{p}^k + p_{\text{test}}^{k+1} - p^{k+1}$ $v^{k+1} = \bar{v}^k + \sigma \mathcal{F}^{-1}(\hat{u}^{k+1} \mathcal{F}\kappa)$ $v_{\text{test}}^{k+1} = \max(-1, \min(1, 2v^{k+1} - \bar{v}^k - \sigma f))$ $\bar{v}^{k+1} = \bar{v}^k + v_{\text{test}}^{k+1} - v^{k+1}$	according to (9) according to (11)

Table 9: The scaled preconditioned Douglas-Rachford iteration for L^1 -TV deblurring.

while sharing the convenient properties of TV, in particular, the ability to model images with edges or, more generally, discontinuities on hypersurfaces. Especially, TGV is able to significantly reduce the so-called *staircasing effect* in the solutions, i.e., the undesired appearance of edges. It works convincingly well in various fields, e.g., for linear inverse problems [4], for diffusing tensor imaging [16], for JPEG decompression [2], for multichannel images [1] and in magnetic resonance imaging (MRI) [12].

The second-order total generalized variation $\operatorname{TGV}_\alpha^2$ is defined as in [4, 5].

Definition 2 Let $\Omega \subset \mathbf{R}^d$ be a bounded Lipschitz domain and $\alpha = (\alpha_0, \alpha_1) > 0$ regularization parameters. The *Total Generalized Variation* of second order of $u \in L^1_{loc}(\Omega)$ is the value assigned by the following functional

$$\operatorname{TGV}_\alpha^2(u) = \sup \left\{ \int_\Omega u \operatorname{div}^2 q \, dx \mid q \in \mathcal{C}_c^2(\Omega, S^{d \times d}), \|q\|_\infty \leq \alpha_0, \|\operatorname{div} q\|_\infty \leq \alpha_1 \right\}. \quad (18)$$

Here, $S^{d \times d}$ is the set of symmetric matrices and $\mathcal{C}_c^2(\Omega, S^{d \times d})$ denotes the vector space of compactly supported, twice continuously differentiable $S^{d \times d}$ -valued mappings. The divergences $\operatorname{div} q \in \mathcal{C}_c^1(\Omega, \mathbf{R}^d)$ and $\operatorname{div}^2 q \in \mathcal{C}_c(\Omega)$ are defined by

$$(\operatorname{div} q)_i = \sum_{j=1}^d \frac{\partial q_{ij}}{\partial x_j}, \quad \operatorname{div}^2 q = \sum_{i=1}^d \frac{\partial^2 q_{ii}}{\partial x_i^2} + 2 \sum_{i < j} \frac{\partial q_{ij}}{\partial x_i \partial x_j}$$

while the norms of $q \in \mathcal{C}_c(\Omega, S^{d \times d})$, $p \in \mathcal{C}_c(\Omega, \mathbf{R}^d)$ are given by

$$\|q\|_\infty = \sup_{x \in \Omega} \left(\sum_{i=1}^d |q_{ii}(x)|^2 + 2 \sum_{i < j} |q_{ij}(x)|^2 \right)^{1/2}, \quad \|p\|_\infty = \sup_{x \in \Omega} \left(\sum_{i=1}^d |p_i(x)|^2 \right)^{1/2}. \quad (19)$$

For our purposes, we will utilize that $\operatorname{TGV}_\alpha^2$ can be expressed as a minimum [4]:

$$\operatorname{TGV}_\alpha^2(u) = \min_{w \in \operatorname{BD}(\Omega)} \alpha_1 \|\nabla u - w\|_{\mathcal{M}} + \alpha_0 \|\mathcal{E}w\|_{\mathcal{M}} \quad (20)$$

where $\operatorname{BD}(\Omega)$ denotes the space of vector fields of *Bounded Deformation* [14] which is the set of vector fields whose weak symmetrized derivative $\mathcal{E}w = (\nabla w + \nabla w^T)/2$ is a matrix-valued

Radon measure. Moreover, $\|\cdot\|_{\mathcal{M}}$ denotes the Radon norm for the corresponding vector-valued and matrix-valued Radon measures (the L^1 -type norm for measures).

We are therefore concerned with discrete denoising problems of the type

$$\min_{\substack{u \in U \\ w \in V}} F(u) + \alpha_1 \|\nabla u - w\|_1 + \alpha_0 \|\mathcal{E}w\|_1, \quad F(u) = \begin{cases} \frac{1}{2} \|u - f\|_2^2 & \text{for } L^2\text{-denoising,} \\ \|u - f\|_1 & \text{for } L^1\text{-denoising,} \end{cases} \quad (21)$$

with the notation of Section 3 and \mathcal{E} being a discretization of the symmetrized derivative which will be described in the following. For this purpose, consider the spaces

$$U = \{u : \Omega \rightarrow \mathbf{R}\}, \quad V = \{u : \Omega \rightarrow \mathbf{R}^2\}, \quad W = \{u : \Omega \rightarrow S^{2 \times 2}\}.$$

where U and V have already been introduced in Section 3. We identify elements in $q \in W$ with $q = (q^1, q^2, q^3) \in U^3$ and employ the scalar product

$$\langle q, q' \rangle_W = \langle q^1, (q')^1 \rangle_U + \langle q^2, (q')^2 \rangle_U + 2 \langle q^3, (q')^3 \rangle_U.$$

Utilizing forward differences, the symmetrized derivative can be defined as follows

$$\mathcal{E}w = \begin{bmatrix} \partial_x^+ w^1 & \frac{1}{2}(\partial_y^+ w^1 + \partial_x^+ w^2) \\ \frac{1}{2}(\partial_y^+ w^1 + \partial_x^+ w^2) & \partial_y^+ w^2 \end{bmatrix} = \begin{bmatrix} \partial_x^+ w^1 \\ \partial_y^+ w^2 \\ \frac{1}{2}(\partial_y^+ w^1 + \partial_x^+ w^2) \end{bmatrix}$$

where the second equation has to be understood in terms of the identification $W = U^3$. Consequently, the negative adjoint realizes a discrete negative divergence operator according to $\langle \mathcal{E}w, q \rangle_W = -\langle w, \operatorname{div} q \rangle_V$ for all $w \in V$, $q \in W$, leading to

$$\operatorname{div} q = \begin{bmatrix} \partial_x^- q^1 + \partial_y^- q^3 \\ \partial_x^- q^3 + \partial_y^- q^2 \end{bmatrix}.$$

With $\Delta w = (\Delta w^1, \Delta w^2)$, we can express the operator $\mathcal{E}^* \mathcal{E}w = -\frac{1}{2} \Delta w - \frac{1}{2} \nabla^- \operatorname{div}^+ w$ where div^+ is the discrete divergence using forward differences and ∇^- is the discrete gradient using backward differences. Finally, we need to define meaningful norms: For $1 \leq t < \infty$ and $q \in W$, let

$$\|q\|_t = \left(\sum_{(i,j) \in \Omega} ((q_{i,j}^1)^2 + (q_{i,j}^2)^2 + 2(q_{i,j}^3)^2)^{t/2} \right)^{1/t}, \quad \|q\|_\infty = \max_{(i,j) \in \Omega} \sqrt{(q_{i,j}^1)^2 + (q_{i,j}^2)^2 + 2(q_{i,j}^3)^2}.$$

With these definitions, we are able to give a meaning to the discrete denoising problems (21). Proceeding in the lines of Subsection 3.1, we derive the preconditioned Douglas-Rachford algorithms separately for L^2 - and L^1 -denoising and develop a suitable preconditioner afterwards. For $\operatorname{TGV}_\alpha^2$, the latter turns out to require more effort than for the TV case.

4.1 L^2 - and L^1 -Denoising

We follow the lines of, e.g., [1] and write (21) as a saddle-point problem according to

$$\min_{\substack{u \in U \\ w \in V}} \max_{\substack{p \in V \\ q \in W}} \langle \nabla u - w, p \rangle_V + \langle \mathcal{E}w, q \rangle_W + F(u) - \mathcal{I}_{\{\|p\|_\infty \leq \alpha_1\}}(p) - \mathcal{I}_{\{\|q\|_\infty \leq \alpha_0\}}(q). \quad (22)$$

Therefore, $X = U \times V$, $Y = V \times W$ and, denoting $x = (u, w)$, $y = (p, q)$,

$$\mathcal{K} = \begin{bmatrix} \nabla & -I \\ 0 & \mathcal{E} \end{bmatrix}, \quad \mathcal{F}(u, w) = F(u), \quad \mathcal{G}(p, q) = \mathcal{I}_{\{\|p\|_\infty \leq \alpha_1\}}(p) + \mathcal{I}_{\{\|q\|_\infty \leq \alpha_0\}}(q) \quad (23)$$

for which the resolvent of $\partial\mathcal{G}$ is easy to compute: It is given by (9), i.e., the pair $(\mathcal{P}_{\alpha_1}(p), \mathcal{P}_{\alpha_0}(q))$ where

$$\mathcal{P}_{\alpha_1}(p) = \frac{p}{\max(1, \frac{|p|}{\alpha_1})}, \quad \mathcal{P}_{\alpha_0}(q) = \frac{q}{\max(1, \frac{|q|}{\alpha_0})} \quad (24)$$

and $|p| = \sqrt{(p^1)^2 + (p^2)^2}$ as well as $|q| = \sqrt{(q^1)^2 + (q^2)^2 + 2(q^3)^2}$ for $p \in V$ and $q \in W$ (also compare with (9)). In case of L^2 -denoising, (PDRQ) will be employed with

$$Q = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \quad f_0 = \begin{bmatrix} f \\ 0 \end{bmatrix}.$$

In case of L^1 -denoising, (PDR) will be used with a scaling of \mathcal{K} by a positive factor $\tau > 0$ as already utilized in Section 3. This leads to the following operators to be preconditioned:

$$T_{L^2\text{-TGV}} = \begin{bmatrix} \sigma I - \sigma^2 \Delta & \sigma^2 \operatorname{div} \\ -\sigma^2 \nabla & \sigma^2(I - \operatorname{div} \mathcal{E}) \end{bmatrix}, \quad T_{L^1\text{-TGV}} = \begin{bmatrix} I - (\sigma\tau)^2 \Delta & (\sigma\tau)^2 \operatorname{div} \\ -(\sigma\tau)^2 \nabla & (1 + (\sigma\tau)^2)I - (\sigma\tau)^2 \operatorname{div} \mathcal{E} \end{bmatrix}. \quad (25)$$

4.2 The Preconditioner

Choosing $\lambda_u, \lambda_w, \mu > 0$, our goal is now to find preconditioners for

$$T = \begin{bmatrix} \lambda_u I - \mu \Delta & \mu \operatorname{div} \\ -\mu \nabla & \lambda_w I - \mu \operatorname{div} \nabla \end{bmatrix}$$

which corresponds to a discrete solution of the system of partial differential equations

$$\begin{cases} \lambda_u u - \mu \Delta u + \mu \operatorname{div} w = b_u, \\ -\mu \nabla u + \lambda_w w - \mu \operatorname{div} \mathcal{E} w = b_w & \text{in } \Omega, \\ \frac{\partial u}{\partial \nu} = 0, \quad (\nabla w + \nabla w^T) \nu = 0 & \text{on } \partial\Omega. \end{cases} \quad (26)$$

By definition, feasible preconditioners require T to be invertible, which can easily be verified.

Lemma 1 *If $\lambda_w \geq \mu$, then T is a positive definite operator.*

Proof For $x = (u, w)$ we have

$$\begin{aligned} \langle Tx, x \rangle_X &= \lambda_u \|u\|_2^2 + \mu \|\nabla u\|_2^2 - \mu \langle u, \nabla^* w \rangle - \mu \langle w, \nabla u \rangle + \lambda_w \|w\|_2^2 + \mu \|\mathcal{E} w\|_2^2 \\ &\geq \mu \|\nabla u - w\|_2^2 + \lambda_u \|u\|_2^2. \end{aligned}$$

It could be seen that if $\langle Tx, x \rangle_X = 0$, then $u = 0$ and $w = 0$ leading to $x = 0$.

Like in the TV case, we like to employ a suitable Red-Black Gauss-Seidel method. However, as $-\operatorname{div} \mathcal{E}w = -\frac{1}{2}\Delta w - \frac{1}{2}\nabla^- \operatorname{div}^+$ contains discrete mixed derivatives, the corresponding finite-difference stencil will no longer depend only the 4-neighborhood of a pixel but also involve points from the 8-neighborhood. This problem can be circumvented by performing a suitable Red-Black Gauss-Seidel preconditioning on the equation $T'(u, w) = (b_u, w_u)$ according to

$$\underbrace{\begin{bmatrix} \lambda_u I - \mu \Delta & \mu \operatorname{div} \\ -\mu \nabla & \lambda_w I - \mu \Delta \end{bmatrix}}_{=T'} \begin{bmatrix} u \\ w \end{bmatrix} = \begin{bmatrix} b_u \\ b_w \end{bmatrix} \quad \text{or} \quad \begin{cases} \lambda_u u - \mu \Delta + \mu \operatorname{div} w = b_u, \\ -\mu \nabla u + \lambda_w w - \mu \Delta w = b_w \\ \frac{\partial u}{\partial \nu} = 0, \quad \frac{\partial w}{\partial \nu} = 0 \end{cases} \quad \begin{array}{ll} \text{in } \Omega, \\ \text{on } \partial \Omega. \end{array} \quad (27)$$

This is justified by the following lemma.

Lemma 2 *It holds that $T' - T \geq 0$.*

Proof For $x = (u, w)$ one computes

$$\langle (T' - T)x, x \rangle_X = \mu \|\nabla w\|^2 - \mu \|\mathcal{E}w\|^2 = \mu (\|\partial_y^+ w^1\|^2 + \|\partial_x^+ w^2\|^2 - 2 \underbrace{\|\frac{1}{2}(\partial_y^+ w^1 + \partial_x^+ w^2)\|^2}_{\leq \frac{1}{2}\|\partial_y^+ w^1\|^2 + \frac{1}{2}\|\partial_x^+ w^2\|^2}) \geq 0.$$

It is then clear that feasible preconditioners M for T' are also feasible for T . Now, as (27) is a finite-difference equation for $x = (u, w^1, w^2)$, it can be represented, as in the case of (12), by finite-difference stencils with the difference that each entry corresponds to a 3×3 matrix. Here, each grid point is associated with a triple $x_{i,j} = (u_{i,j}, w_{i,j}^1, w_{i,j}^2)$. Again, we have to distinguish whether a grid point is in the interior, on a boundary edge or a corner. Adopting the notation of Table 4, the center element can be computed to be

$$\begin{aligned} & \underbrace{\begin{bmatrix} \lambda_u + 2\mu & \mu & \mu \\ \mu & \lambda_w + 2\mu & 0 \\ \mu & 0 & \lambda_w + 2\mu \end{bmatrix}}_{=\mathcal{C}_{NW}} \quad \underbrace{\begin{bmatrix} \lambda_u + 3\mu & \mu & \mu \\ \mu & \lambda_w + 3\mu & 0 \\ \mu & 0 & \lambda_w + 3\mu \end{bmatrix}}_{=\mathcal{C}_N} \quad \underbrace{\begin{bmatrix} \lambda_u + 2\mu & 0 & \mu \\ 0 & \lambda_w + 2\mu & 0 \\ 0 & 0 & \lambda_w + 2\mu \end{bmatrix}}_{=\mathcal{C}_{NE}} \\ & \underbrace{\begin{bmatrix} \lambda_u + 3\mu & \mu & \mu \\ \mu & \lambda_w + 3\mu & 0 \\ \mu & 0 & \lambda_w + 3\mu \end{bmatrix}}_{=\mathcal{C}_W} \quad \underbrace{\begin{bmatrix} \lambda_u + 4\mu & \mu & \mu \\ \mu & \lambda_w + 4\mu & 0 \\ \mu & 0 & \lambda_w + 4\mu \end{bmatrix}}_{=\mathcal{C}_O} \quad \underbrace{\begin{bmatrix} \lambda_u + 3\mu & 0 & \mu \\ 0 & \lambda_w + 3\mu & 0 \\ 0 & 0 & \lambda_w + 3\mu \end{bmatrix}}_{=\mathcal{C}_E} \\ & \underbrace{\begin{bmatrix} \lambda_u + 2\mu & \mu & 0 \\ \mu & \lambda_w + 2\mu & 0 \\ 0 & 0 & \lambda_w + 2\mu \end{bmatrix}}_{=\mathcal{C}_{SW}} \quad \underbrace{\begin{bmatrix} \lambda_u + 3\mu & \mu & 0 \\ \mu & \lambda_w + 3\mu & 0 \\ 0 & 0 & \lambda_w + 3\mu \end{bmatrix}}_{=\mathcal{C}_S} \quad \underbrace{\begin{bmatrix} \lambda_u + 2\mu & 0 & 0 \\ 0 & \lambda_w + 2\mu & 0 \\ 0 & 0 & \lambda_w + 2\mu \end{bmatrix}}_{=\mathcal{C}_{SE}} \end{aligned}$$

The off-center entries are all the same for each pixel and only depend on the position relative to the center. Denoting

$$\mathcal{R} = \begin{bmatrix} -\mu & 0 & 0 \\ -\mu & -\mu & 0 \\ 0 & 0 & -\mu \end{bmatrix}, \quad \mathcal{U} = \begin{bmatrix} -\mu & 0 & -\mu \\ 0 & -\mu & 0 \\ 0 & 0 & -\mu \end{bmatrix}.$$

and their adjoints by \mathcal{R}^* and \mathcal{U}^* , respectively, the block stencil for (27) is given by

$$\begin{array}{ccc} \underbrace{\begin{bmatrix} \mathcal{C}_{NW} & \mathcal{R} \\ \mathcal{U}^* & \end{bmatrix}}_{=\mathcal{S}_{NW}} & \underbrace{\begin{bmatrix} \mathcal{R}^* & \mathcal{C}_N & \mathcal{R} \\ \mathcal{U}^* & & \end{bmatrix}}_{=\mathcal{S}_N} & \underbrace{\begin{bmatrix} \mathcal{R}^* & \mathcal{C}_{NE} \\ \mathcal{U}^* & \end{bmatrix}}_{=\mathcal{S}_{NE}} \\ \underbrace{\begin{bmatrix} \mathcal{U} & \\ \mathcal{C}_W & \mathcal{R} \\ \mathcal{U}^* & \end{bmatrix}}_{=\mathcal{S}_W} & \underbrace{\begin{bmatrix} \mathcal{U} & \\ \mathcal{R}^* & \mathcal{C}_O & \mathcal{R} \\ \mathcal{U}^* & & \end{bmatrix}}_{=\mathcal{S}_O} & \underbrace{\begin{bmatrix} \mathcal{U} & \\ \mathcal{R}^* & \mathcal{C}_E \\ \mathcal{U}^* & \end{bmatrix}}_{=\mathcal{S}_E} \\ \underbrace{\begin{bmatrix} \mathcal{U} & \\ \mathcal{C}_{SW} & \mathcal{R} \\ \mathcal{U}^* & \end{bmatrix}}_{=\mathcal{S}_{SW}} & \underbrace{\begin{bmatrix} \mathcal{U} & \\ \mathcal{R}^* & \mathcal{C}_S & \mathcal{R} \\ \mathcal{U}^* & & \end{bmatrix}}_{=\mathcal{S}_S} & \underbrace{\begin{bmatrix} \mathcal{U} & \\ \mathcal{R}^* & \mathcal{C}_{SE} \\ \mathcal{U}^* & \end{bmatrix}}_{=\mathcal{S}_{SE}} \end{array} \quad (28)$$

adopting again the notation for the different types of grid points from Table 4 and highlighting the center element. Based on this, a block Red-Black symmetric Gauss-Seidel iteration can be performed, needing only forward evaluation of $\mathcal{U}, \mathcal{U}^*, \mathcal{R}, \mathcal{R}^*$ and the inverses of the center matrices $\mathcal{C}_{\{\cdot\}}$ which can be computed explicitly. Denoting by $b_{i,j} = ((b_u)_{i,j}, (b_w^1)_{i,j}, (b_w^2)_{i,j})$ for $(i, j) \in \Omega$, $\mathcal{C}_{i,j}$ the corresponding center element for the grid point (i, j) , introducing x_{red} and x_{black} on Ω_{red} and Ω_{black} , respectively and extending them outside their domain by 0, n -fold application of the block Red-Black symmetric Gauss-Seidel method, denoted $\text{BSRBGS}_{\lambda_u, \lambda_w, \mu}^n$, corresponds to

$$\left\{ \begin{array}{l} \text{BSRBGS}_{\lambda_u, \lambda_w, \mu}^n(x^k, b^k) = (x_{\text{red}}^{k+1}, x_{\text{black}}^{k+1}) \\ (x_{\text{red}}^{k+(\nu+1/2)/n})_{i,j} = \mathcal{C}_{i,j}^{-1}(b_{i,j} - \mathcal{R}^*(x_{\text{black}}^{k+\nu/n})_{i-1,j} - \mathcal{R}(x_{\text{black}}^{k+\nu/n})_{i+1,j} \\ \quad - \mathcal{U}(x_{\text{black}}^{k+\nu/n})_{i,j-1} - \mathcal{U}^*(x_{\text{black}}^{k+\nu/n})_{i,j+1}) \quad (i, j) \in \Omega_{\text{red}}, \\ (x_{\text{black}}^{k+(\nu+1)/n})_{i,j} = \mathcal{C}_{i,j}^{-1}(b_{i,j} - \mathcal{R}^*(x_{\text{red}}^{k+(\nu+1/2)/n})_{i-1,j} - \mathcal{R}(x_{\text{red}}^{k+(\nu+1/2)/n})_{i+1,j} \\ \quad - \mathcal{U}(x_{\text{red}}^{k+(\nu+1/2)/n})_{i,j-1} - \mathcal{U}^*(x_{\text{red}}^{k+(\nu+1/2)/n})_{i,j+1}) \quad (i, j) \in \Omega_{\text{black}}, \\ \nu = 0, \dots, n-1, \\ (x_{\text{red}})^{k+1})_{i,j} = \mathcal{C}_{i,j}^{-1}(b_{i,j} - \mathcal{R}^*(x_{\text{black}}^{k+1})_{i-1,j} - \mathcal{R}(x_{\text{black}}^{k+1})_{i+1,j} \\ \quad - \mathcal{U}(x_{\text{black}}^{k+1})_{i,j-1} - \mathcal{U}^*(x_{\text{black}}^{k+1})_{i,j+1}). \end{array} \right. \quad (29)$$

Let us show feasibility. This is implied by the following general theorem about block symmetric Gauss-Seidel and block SSOR:

Proposition 2 Let $N, m \geq$ and $T \in \mathbf{R}^{mN \times mN}$ be a symmetric, positive definite block matrix with blocks $T_{i,j} \in \mathbf{R}^{m \times m}$ for $i, j = 1, \dots, N$ and $\omega \in]0, 2[$. Then, block SSOR, i.e.,

$$\left\{ \begin{array}{ll} x_i^{k+1/2} = (1 - \omega)x_i^k + \omega T_{i,i}^{-1} \left(b_i - \sum_{j < i} T_{i,j} x_j^{k+1/2} - \sum_{j > i} T_{i,j} x_j^k \right) & i = 1, \dots, N, \\ x_i^{k+1} = (1 - \omega)x_i^{k+1/2} + \omega T_{i,i}^{-1} \left(b_i - \sum_{j > i} T_{j,i} x_i^{k+1} - \sum_{j < i} T_{j,i} x_i^{k+1/2} \right) & i = N, \dots, 1, \end{array} \right. \quad (30)$$

is well-defined and feasible for the solution of $Tx = b$.

Proof Denoting $T = D - E - E^*$ with $D = \text{diag}(T_{1,1}, \dots, T_{N,N})$ the block diagonal matrix of T , and E, E^* the negative block lower and upper triangular matrix, respectively, the two steps in (30) correspond to the application of the preconditioner $M_0 = \frac{1}{\omega}D - E$ followed by M_0^* . By Proposition 1 (i), we only need to check the positive definiteness of $M_0 - \frac{1}{2}T$. Note that since

T is positive definite, each $T_{i,i}$ is positive definite, hence $\langle T_{i,i}\xi, \xi \rangle \geq c\|\xi\|^2$ for $c > 0$ and each $\xi \in \mathbf{R}^m$, $i = 1, \dots, N$. Hence, for $x \in \mathbf{R}^{mN}$,

$$\langle (M_0 - \frac{1}{2}T)x, x \rangle = (\frac{1}{\omega} - \frac{1}{2})\langle Dx, x \rangle + \frac{1}{2}\langle (E^* - E)x, x \rangle = (\frac{1}{\omega} - \frac{1}{2})\langle Dx, x \rangle \geq c(\frac{1}{\omega} - \frac{1}{2})\|x\|^2.$$

Hence, by Proposition 1 (ii) and Lemmas 1 and 2, BSRBGS $_{\lambda_u, \lambda_w, \mu}^n$ as described in (29) is feasible for the solution of (27) provided that $\lambda_w \geq \mu$. It can therefore be written as

$$x^{k+1} = x^k + M^{-1}(b^k - T'x^k)$$

with $M - T' \geq 0$. To apply the preconditioner for T , i.e., the discrete solution of (26), we have replace T' by T which means to correct the data according to

$$x^{k+1} = x^k + M^{-1}(b^k + (T' - T)x^k - T'x^k).$$

The corrected data then reads as

$$(b')^k = b^k + (T' - T)x^k = \begin{bmatrix} b_u^k \\ b_w^k \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{\mu}{2}(\nabla^- \operatorname{div}^+ - \Delta) \end{bmatrix} \begin{bmatrix} u^k \\ w^k \end{bmatrix} = \begin{bmatrix} b_u^k \\ b_w^k + \frac{\mu}{2}(\nabla^- \operatorname{div}^+ - \Delta)w^k \end{bmatrix}. \quad (31)$$

In the algorithms, the block symmetric Gauss-Seidel preconditioner according to (29) is then applied to the data x^k and $(b')^k$.

Remark 3 An alternative to this approach would be to apply one block symmetric Gauss-Seidel step according to (29) for T and iterate n times. This results in a preconditioner which performs, n times, a correction according to (31) and a red \rightarrow black \rightarrow red update according to (29). Compared to (29), n more corrections and red updates have to be performed.

4.3 The Algorithms

The PDRQ algorithm for the solution of the L^2 -TGV $_{\alpha}^2$ denoising problem with the block symmetric Gauss-Seidel preconditioner as previously described is summarized in Table 10. Likewise, the PDR algorithm for L^1 -TGV $_{\alpha}^2$ denoising can be found in Table 11. For the latter, it can be noted that \mathcal{F} does not depend on w , hence $(I + \sigma\partial\mathcal{F})^{-1}$ is the identity with respect to the w -component leading to $\bar{w}^k = w^k$ for each k if $w^0 = \bar{w}^0$. Hence, \bar{w}^k can be replaced by w^k , a step which has already been incorporated in the algorithm. It is then clear by Theorem 1 that both algorithms converge. The u -part then gives a solution to the original denoising problem.

5 The Primal-Dual Gap as Stopping Criterion

For convex-concave saddle-point problems (2) which correspond to the solution of the primal and dual problem (3), the so-called *primal-dual gap* provides, if finite, an upper estimate on the functional distance to a minimizer and maximizer of the primal and dual problem, respectively, in case that both exist. It is defined as

$$\mathfrak{G}(x, y) = \mathcal{F}(x) + \mathcal{G}^*(\mathcal{K}x) + \mathcal{F}^*(-\mathcal{K}^*y) + \mathcal{G}(y). \quad (32)$$

The primal-dual gap vanishes exactly in the saddle-points, i.e., $\mathfrak{G}(x^*, y^*) = 0$ if and only if (x^*, y^*) solves (2). Moreover, in case Fenchel-Rockafellar duality holds,

$$\begin{aligned} \mathfrak{G}(x, y) &= \left(\mathcal{F}(x) + \mathcal{G}^*(\mathcal{K}x) - \left(\min_{x' \in X} \mathcal{F}(x') + \mathcal{G}^*(\mathcal{K}x') \right) \right) \\ &\quad + \left(\mathcal{F}^*(-\mathcal{K}^*y) + \mathcal{G}(y) - \left(\min_{y' \in Y} \mathcal{F}^*(-\mathcal{K}^*y') + \mathcal{G}(y') \right) \right), \end{aligned}$$

PDRQ Objective: L^2 -TGV denoising		$\min_{u \in U} \frac{1}{2} \ u - f\ _2^2 + \text{TGV}_\alpha^2(u)$
Initialization:	$(u^0, w^0, \bar{p}^0, \bar{q}^0) \in U \times V \times V \times W$ initial guess, $\sigma > 0$ step-size, $n \geq 1$ inner iterations for block symmetric Gauss-Seidel	
Iteration:	$b_u^k = \sigma(f + \operatorname{div} \bar{p}^k)$ $b_w^k = \sigma(\bar{p}^k + \operatorname{div} \bar{q}^k) + \frac{1}{2}(\nabla^- \operatorname{div}^+ - \Delta)w^k$ $(u^{k+1}, w^{k+1}) = \text{BSRBGS}_{\sigma, \sigma^2, \sigma^2}^n(u^k, w^k, b_u^k, b_w^k)$ according to (29) $p^{k+1} = \bar{p}^k + \sigma(\nabla u^{k+1} - w^{k+1})$ $q^{k+1} = \bar{q}^k + \sigma \mathcal{E} w^{k+1}$ $p_{\text{test}}^{k+1} = \mathcal{P}_{\alpha_1}(2p^{k+1} - \bar{p}^k)$ according to (24) $\bar{p}^{k+1} = \bar{p}^k + p^{k+1} - p^{k+1}$ $q_{\text{test}}^{k+1} = \mathcal{P}_{\alpha_0}(2q^{k+1} - \bar{q}^k)$ according to (24) $\bar{q}^{k+1} = \bar{q}^k + q_{\text{test}}^{k+1} - q^{k+1}$	

Table 10: The preconditioned Douglas-Rachford iteration for L^2 -TGV denoising.

PDR Objective: L^1 -TGV denoising		$\min_{u \in U} \ u - f\ _1 + \text{TGV}_\alpha^2(u)$
Initialization:	$(u^0, w^0, \bar{u}^0, \bar{p}^0, \bar{q}^0) \in U \times V \times U \times V \times W$ initial guess, $\sigma > 0$ step-size, $\tau > 0$ gradient scaling, $n \geq 1$ inner iterations for block symmetric Gauss-Seidel	
Iteration:	$b_u^k = \bar{u}^k + \sigma \tau \operatorname{div} \bar{p}^k$ $b_w^k = w^k + \sigma \tau (\bar{p}^k + \operatorname{div} \bar{q}^k + \frac{1}{2}(\nabla^- \operatorname{div}^+ - \Delta)w^k)$ $(u^{k+1}, w^{k+1}) = \text{BSRBGS}_{1, 1 + (\sigma \tau)^2, (\sigma \tau)^2}^n(u^k, w^k, b_u^k, b_w^k)$ according to (29) $p^{k+1} = \bar{p}^k + \sigma \tau (\nabla u^{k+1} - w^{k+1})$ $q^{k+1} = \bar{q}^k + \sigma \tau \mathcal{E} w^{k+1}$ $u_{\text{test}}^{k+1} = \mathcal{S}_\sigma(2u^{k+1} - \bar{u}^k)$ according to (10) $\bar{u}^{k+1} = \bar{u}^k + u_{\text{test}}^{k+1} - u^{k+1}$ $p_{\text{test}}^{k+1} = \mathcal{P}_{\alpha_1}(2p^{k+1} - \bar{p}^k)$ according to (24) $\bar{p}^{k+1} = \bar{p}^k + p^{k+1} - p^{k+1}$ $q_{\text{test}}^{k+1} = \mathcal{P}_{\alpha_0}(2q^{k+1} - \bar{q}^k)$ according to (24) $\bar{q}^{k+1} = \bar{q}^k + q_{\text{test}}^{k+1} - q^{k+1}$	

Table 11: The scaled preconditioned Douglas-Rachford iteration for L^1 -TGV denoising.

so, if $\mathfrak{G}(x, y) \leq \varepsilon$ for some $\varepsilon > 0$, the pair (x, y) is ε -optimal for both the primal and dual problem.

In order to use the primal-dual gap in conjunction with (PDR) and (PDRQ), it has to be finite for quantities which are easy to derive from the iterates. For the PDR method, we can see that in case of weak convergence,

$$\begin{cases} x_{\text{test}}^{k+1} = (I + \sigma \partial \mathcal{F})^{-1}[2x^{k+1} - \bar{x}^k] = \bar{x}^{k+1} + x^{k+1} - \bar{x}^k \rightharpoonup x^* \\ y_{\text{test}}^{k+1} = (I + \sigma \partial \mathcal{G})^{-1}[2y^{k+1} - \bar{y}^k] = \bar{y}^{k+1} + y^{k+1} - \bar{y}^k \rightharpoonup y^* \end{cases}$$

where (x^*, y^*) is a saddle point. As x_{test}^k and y_{test}^k are in the range of the resolvents of the subdifferential, the associated functional values must be finite, i.e., $\mathcal{F}(x_{\text{test}}^k) < \infty$, $\mathcal{G}(y_{\text{test}}^k) < \infty$. However, we cannot guarantee that $\mathcal{G}^*(\mathcal{K}x_{\text{test}}^k) < \infty$ and $\mathcal{F}^*(-\mathcal{K}^*y_{\text{test}}^k) < \infty$ and this has to be derived from the functionals. Likewise, for the PDRQ method, we can choose $x_{\text{test}}^k = x^k$ and

obtain finite $\mathcal{F}(x_{\text{test}}^k)$. If Q is positive definite, then also $\mathcal{F}(-\mathcal{K}^*y_{\text{test}}^k) < \infty$. In general, however, Q is only assumed to be positive semi-definite, so $\mathcal{F}(-\mathcal{K}^*y_{\text{test}}^k)$ might be infinite.

One possibility to circumvent the problem of infinite primal-dual gaps is to consider the *restricted primal-dual gap* as done in [6] which requires an a-priori bound on the distance of the set of saddle points to the initial guess. As this restricted gap might be difficult to compute in practice, we pursue the following approach: Deriving a-priori a suitable set in which the saddle points are contained, we possibly modify \mathcal{F} and \mathcal{G} outside this set such that the solution set remains the same and \mathcal{F}^* as well as \mathcal{G}^* is finite everywhere. This way, the modified gap is always finite and can be used as a stopping criterion.

In the following, we discuss problems in which either the primal-dual gap is always finite on the iterates or the outlined approach can successfully be carried out. In particular, we derive a modified primal-dual gap for TGV.

5.1 L^2 -TV Denoising

For the L^2 -TV denoising problem, the primal-dual gap is well-known, see [6], for instance. It reads as

$$\mathfrak{G}_{L^2\text{-TV}}(u, p) = \frac{\|u - f\|_2^2}{2} + \alpha \|\nabla u\|_1 + \frac{\|\operatorname{div} p + f\|_2^2}{2} - \frac{\|f\|_2^2}{} + \mathcal{I}_{\{\|p\|_\infty \leq \alpha\}}(p) \quad (33)$$

which is always finite for the pair $(u^{k+1}, p_{\text{test}}^{k+1})$ provided by the algorithm in Table 5 and can directly be used as stopping criterion.

Note that an optimal pair (u^*, p^*) satisfies $-p^* \in \partial(\alpha \|\cdot\|_1)(\nabla u^*)$ and $u^* = \operatorname{div} p^* + f$, so

$$\begin{aligned} \mathfrak{G}_{L^2\text{-TV}}(u, p) &\geq \frac{\|u - f\|_2^2}{2} + \alpha \|\nabla u\|_1 - \frac{\|u^* - f\|_2^2}{2} - \alpha \|\nabla u^*\|_1 \\ &\geq \frac{\|u - f\|_2^2}{2} - \frac{\|u^* - f\|_2^2}{2} + \langle \nabla u - \nabla u^*, -p^* \rangle \\ &= \frac{\|u\|_2^2}{2} - \frac{\|u^*\|_2^2}{2} + \langle u - u^*, \operatorname{div} p^* + f \rangle = \frac{\|u - u^*\|_2^2}{2}, \end{aligned}$$

meaning that the primal-dual gap can also be used to control the approximation error with respect to the solution in norm.

5.2 L^2 -TGV Denoising

If α TV is replaced by TGV_α^2 , the data (23) has to be used for the saddle-point problem. One computes the primal-dual gap function as

$$\begin{aligned} \mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q) &= \frac{\|u - f\|_2^2}{2} + \alpha_1 \|\nabla u - w\|_1 + \alpha_0 \|\mathcal{E}w\|_1 + \frac{\|\operatorname{div} p + f\|_2^2}{2} - \frac{\|f\|_2^2}{2} \\ &\quad + \mathcal{I}_{\{0\}}(p + \operatorname{div} q) + \mathcal{I}_{\{\|p\|_\infty \leq \alpha_1\}}(p) + \mathcal{I}_{\{\|q\|_\infty \leq \alpha_0\}}(q). \end{aligned} \quad (34)$$

Here, the algorithm according to Table 10 generates $(p_{\text{test}}^{k+1}, q_{\text{test}}^{k+1})$ for which $\mathcal{I}_{\{0\}}(p_{\text{test}}^{k+1} + \operatorname{div} q_{\text{test}}^{k+1})$ might be infinite, rendering this primal-dual gap useless as stopping criterion.

To circumvent this problem, first suppose that we have a L^1 -norm bound C_w available for w^* belonging to a solution pair (u^*, w^*) for problem (21), i.e., (u^*, w^*) optimal implies $\|w^*\|_1 \leq C_w$. Then, we can choose

$$\mathcal{F}(u, w) = \frac{\|u - f\|_2^2}{2} + \mathcal{I}_{\{\|w\|_1 \leq C_w\}}(w).$$

This does obviously does not change the solution set of problem (21). On the dual side, however, $\mathcal{I}_{\{0\}}(p + \operatorname{div} q)$ becomes $C_w \|p + \operatorname{div} q\|_\infty$. For w given, taking $\max(C_w, \|w\|_1)$ instead of C_w leads to a finite $\mathcal{F}(u, w)$ and a primal-dual gap according to

$$\begin{aligned}\mathfrak{G}_{L^2\text{-TGV}}^{C_w}(u, w, p, q) &= \frac{\|u - f\|_2^2}{2} + \alpha_1 \|\nabla u - w\|_1 + \alpha_0 \|\mathcal{E}w\|_1 + \frac{\|\operatorname{div} p + f\|_2^2}{2} - \frac{\|f\|_2^2}{} \\ &\quad + \max(C_w, \|w\|_1) \|p + \operatorname{div} q\|_\infty + \mathcal{I}_{\{\|p\|_\infty \leq \alpha_1\}}(p) + \mathcal{I}_{\{\|q\|_\infty \leq \alpha_0\}}(q)\end{aligned}\quad (35)$$

which is finite for $(u^{k+1}, w^{k+1}, p_{\text{test}}^{k+1}, q_{\text{test}}^{k+1})$ according to Table 10 and can be used as a stopping criterion.

Lemma 3 *Let (u^*, w^*) be a solution of (21). Then, for $(u, w, p, q) \in U \times V \times V \times W$,*

$$\frac{\|u - u^*\|_2^2}{2} \leq \mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q) \quad \text{and} \quad \|w^*\|_1 \leq 2(\sqrt{32N_x N_y \mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q)} + \|\nabla u\|_1)$$

If $\|w^*\|_1 \leq C_w$, then the inequalities also hold for $\mathfrak{G}_{L^2\text{-TGV}}^{C_w}$ instead of $\mathfrak{G}_{L^2\text{-TGV}}$.

Proof As $\mathfrak{G}_{L^2\text{-TGV}}$ is a primal-dual gap function, we have

$$\begin{aligned}\mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q) &\geq \frac{\|u - f\|_2^2}{2} + \alpha_1 \|\nabla u - w\|_1 + \alpha_0 \|\mathcal{E}w\|_1 - \frac{\|u^* - f\|_2^2}{2} \\ &\quad - \alpha_1 \|\nabla u^* - w^*\|_1 - \alpha_0 \|\mathcal{E}w^*\|_1.\end{aligned}$$

Choosing a dual solution pair (p^*, q^*) (which has to exist as Fenchel-Rockafellar duality is obviously applicable), optimality conditions tell that $u^* = \operatorname{div} p^* + f$, $p^* + \operatorname{div} q^* = 0$, $-p^* \in \partial(\alpha_1 \|\cdot\|_1)(\nabla u^* - w^*)$ and $-q^* \in \partial(\alpha_0 \|\cdot\|_1)(\mathcal{E}w^*)$. Hence,

$$\begin{aligned}\mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q) &\geq \frac{\|u - f\|_2^2}{2} - \frac{\|u^* - f\|_2^2}{2} + \langle (\nabla u - w) - (\nabla u^* - w^*), -p^* \rangle \\ &\quad + \langle \mathcal{E}w - \mathcal{E}w^*, -q^* \rangle \\ &= \frac{\|u\|_2^2}{2} - \frac{\|u^*\|_2^2}{2} + \langle u - u^*, f \rangle + \langle u - u^*, \operatorname{div} p^* \rangle + \langle w - w^*, p^* + \operatorname{div} q^* \rangle \\ &= \frac{\|u - u^*\|_2^2}{2}.\end{aligned}$$

This proves the first estimate. Proceeding further, we get by the Cauchy-Schwarz estimate and by $\|\nabla\| \leq \sqrt{8}$ that

$$\|\nabla u - \nabla u^*\|_1 \leq \sqrt{N_x N_y} \|\nabla(u - u^*)\|_2 \leq \sqrt{8N_x N_y} \|u - u^*\|_2 \leq \sqrt{32N_x N_y \mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q)}.$$

Consequently,

$$\|\nabla u^*\|_1 \leq \sqrt{32N_x N_y \mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q)} + \|\nabla u\|_1$$

while plugging in $(u^*, 0)$ into the objective functional in (21) gives $\alpha_1 \|\nabla u^* - w^*\|_1 + \alpha_0 \|\mathcal{E}w^*\|_1 \leq \alpha_1 \|\nabla u^*\|_1$ such that

$$\|w^*\|_1 \leq \|\nabla u^* - w^*\|_1 + \|\nabla u^*\|_1 \leq 2\|\nabla u^*\|_1 \leq 2(\sqrt{32N_x N_y \mathfrak{G}_{L^2\text{-TGV}}(u, w, p, q)} + \|\nabla u\|_1).$$

Now, if $\|w^*\|_1 \leq C_w$, the same argumentation can be carried out for $\mathfrak{G}_{L^2\text{-TGV}}^{C_w}$ instead of $\mathfrak{G}_{L^2\text{-TGV}}$.

Lemma 3 gives us a way to obtain the bound C_w and to improve it if possible. We can amend the PDRQ method as follows. Choosing initial values (\bar{p}^0, \bar{q}^0) such that $\|\bar{p}^0\|_\infty \leq \alpha_1$, $\|\bar{q}^0\|_\infty \leq \alpha_0$ and $\bar{p}^0 + \operatorname{div} \bar{q}^0 = 0$, we update C_w according to

$$\begin{cases} (C_w)_0 = 2(\sqrt{32N_x N_y \mathfrak{G}_{L^2-\text{TGV}}(u^0, w^0, \bar{p}^0, \bar{q}^0)} + \|\nabla u^0\|_1), \\ (C_w)_{k+1} = 2(\sqrt{32N_x N_y \mathfrak{G}_{L^2-\text{TGV}}^{(C_w)_k}(u^{k+1}, w^{k+1}, p_{\text{test}}^{k+1}, q_{\text{test}}^{k+1})} + \|\nabla u^{k+1}\|_1) \end{cases} \quad (36)$$

with $(u^{k+1}, w^{k+1}, p_{\text{test}}^{k+1}, q_{\text{test}}^{k+1})$ provided by the PDRQ algorithm in Table 10. The functional $\mathfrak{G}_{L^2-\text{TGV}}^{(C_w)_k}$ evaluated at these points provides, in particular, an effective primal-dual gap and can be used as stopping criterion.

6 Numerical Experiments

We implemented and performed numerical computations for each of the proposed algorithms using Matlab (MATLAB and Image Processing Toolbox Release 2012b, The MathWorks, Inc., Natick, Massachusetts, United States) with 8 cores each at 3.40GHz. Comparisons with general-purpose primal-dual algorithms for the solution of (2) were performed, using the methods in [6] as a reference. Additionally, numerical tests with ADMM or split Bregman [8, 10] were performed where applicable.

6.1 TV-Denoising Problems

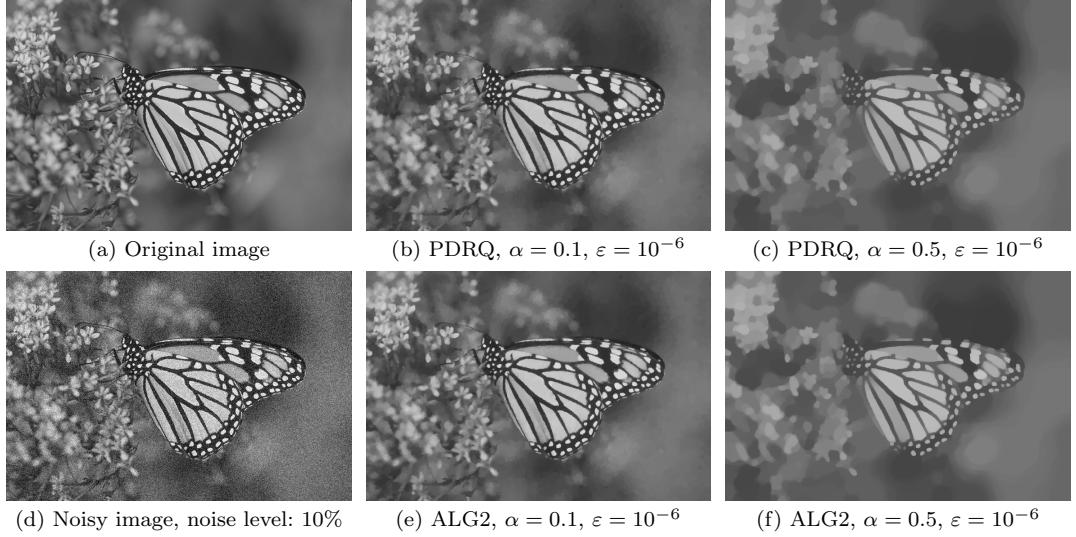
6.1.1 L^2 -Denoising

For L^2 -TV denoising problems, besides the proposed PDRQ, the accelerated first order primal-dual algorithm [6] is also chosen. Their parameter setting is as follows.

- ALG2: $\mathcal{O}(1/k^2)$ accelerated primal-dual algorithm introduced in [6] with adaptive step sizes, $\tau_0 = 1/L$, $\tau_k \sigma_k L^2 = 1$, $\gamma = 0.7$ and $L = 8$. Here we use the same notations for the parameters τ , σ together with θ as in [6] throughout this section.
- PDRQ: Preconditioned Douglas-Rachford method for pure quadratic-linear primal (or dual) functionals in (2). A symmetric Red-Black Gauss-Seidel preconditioner is employed (see Table 5). The notation PDRQ(σ, n) represents the step size σ and n iterations for the preconditioner.

The stopping criterion is chosen as the normalized primal-dual gap $\mathfrak{G}_{L^2-\text{TV}}/(N_x N_y)$ given by (33) and $N_x N_y$ being the number of pixels in the image, see also [3, 6].

For the computations, a butterfly image with size 768×512 , see Figure 1(a), was corrupted by 10% Gaussian noise, see Figure 1(d) and denoising was performed with different regularization parameters and stopping tolerances, see Figure 1 for some of the results. Additionally, Figure 1 contains a table which shows the respective iteration numbers and CPU timings. In a moderate range of the primal-dual gap, PDRQ turns out to be faster than ALG2, in particular for higher regularization parameters. In the detailed comparison for $\alpha = 0.5$ in Figure 2, it can be seen that this range roughly spans 10^{-3} and 10^{-7} both in terms of iteration number and computation time. Below 10^{-7} , the optimal asymptotic convergence speed of $\mathcal{O}(1/k^2)$ of ALG2 becomes visible which PDRQ apparently does not admit, making ALG2 faster there.



$\alpha = 0.1$		$\alpha = 0.5$	
$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$
ALG2	19 (0.72s)	96 (3.69s)	115 (4.38s)
PDRQ	(3,1) 12 (0.76s)	(3,1) 72 (4.49s)	(12,1) 61 (3.76s)
			(12,1) 313 (19.23s)

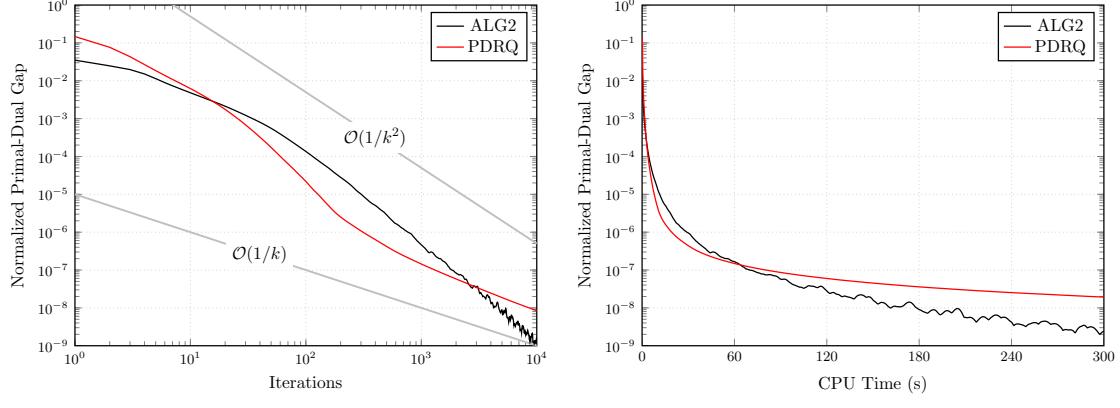
Fig. 1: L^2 -TV denoising: Results and performance. (a) shows the original 768×512 butterfly image while (d) is a noisy version which has been corrupted by 10% Gaussian noise. (b) and (c) are the denoised images by the PDRQ algorithm with different regularization parameter α and normalized primal-dual gap less than 10^{-6} , respectively. (e) and (f) are the respective outcomes of Chambolle-Pock's ALG2 algorithm. The performance of the algorithms used for comparison can be seen in the table. Iteration number and computation time are shown in the format $k(t)$. Additionally, the parameters (σ, n) are given for PDRQ.

6.1.2 L^1 -Denoising

Using an L^1 discrepancy for TV-regularized denoising according to (7) is often beneficial to remove outliers as it is robust with respect to geometry perturbations. We performed experiments for the following algorithms.

- ALG1: $\mathcal{O}(1/k)$ primal-dual algorithm introduced in [6] with constant step sizes, the dual step size $\tau_0 = 0.02$, $\tau\sigma L^2 = 1$, $\theta = 1$ with $L = \sqrt{8}$ as proposed in [6].
- ADMM: Alternating direction method of multipliers [8, 10] based on part 4.4 of [8] with $\beta = 1.5$ and $\alpha = 30$ where β is the coefficient of the L^1 data fidelity term and $\alpha/2$ is the coefficient of the augmented term of ADMM. We use 15 inner symmetric Red-Black Gauss-Seidel iterations to approximately solve the linear elliptic equation.
- PDR: Preconditioned Douglas-Rachford method with symmetric Red-Black Gauss-Seidel preconditioner and scaled gradient according to Table 6. Here, $\sigma = 0.1$, $\tau = 1/\sigma$ and $n = 1$.
- PDRQ: Preconditioned Douglas-Rachford method for the pure quadratic case with symmetric Red-Black Gauss-Seidel preconditioner according to Table 7. Here, $\sigma = 16$ and $n = 1$.

As a practical primal-dual gap is not available, the stopping criterion is chosen as the normalized error of the primal energy $(E^k - E^*)/E^*$, i.e., E^k is the functional value in (7) for u^k and E^* is



(a) Numerical convergence rate of normalized primal-dual gap compared with iteration number.
(b) Numerical convergence rate of normalized primal-dual gap compared with iteration time.

Fig. 2: L^2 -TV denoising: Numerical convergence rates. The normalized primal-dual gap is compared in terms of iteration number and computation time for Figure 1(d) with $\alpha = 0.5$. Note the double-logarithmic and semi-logarithmic scale, respectively, which is used in the plots.

obtained by a large number of iterations, we use 10^5 in this article. It is chosen as the minimal value with respect to all algorithms used for comparison.

Computations were performed on the 256×256 Lena image (Figure 3(a)) corrupted by 25% salt-and-pepper noise (Figure 3(d)) with regularization parameter $\alpha = 2/3$, see Figure 3 for the outcome for some of the algorithms. The table in Figure 4(a) show the numerical evaluation with comparison of the iteration numbers and CPU time cost for the four algorithms. Figure 4(b) is the detailed comparison of iteration numbers. It could be seen that both PDR, PDRQ and ALG1 effectively admit the asymptotic convergence rate of $O(1/k^2)$ which is good for L^1 -type problems but not covered by the theory. In the example, both PDR and PDRQ admit the best performance, with PDRQ being faster for low tolerances.

6.2 TV-Deblurring Problems

6.2.1 L^2 -Deblurring

We consider the model of L^2 -TV deblurring as stated in (14). For comparison, we choose again ALG1 from [6], utilizing the following setup.

- ALG1: $\mathcal{O}(1/k)$ primal-dual algorithm introduced with constant steps, $\tau = 6$, $\tau\sigma L^2 = 1$, $\theta = 1$ with $L = \sqrt{8}$ as in [6].
- PDRQ: Preconditioned Douglas-Rachford method with scaled gradient and preconditioner $M = \sigma L^* L - (\sigma\tau)^2 \Delta_p$ according to Table 8. Here we choose, for step size, $\sigma = 15$ and the scaling parameter as $\tau = 1/\sigma$.

Again, stopping is performed based on the normalized error of energy.

Computations have been carried out for the 512×512 house image in Figure 5(a) which has been degraded by motion blur of approximately 40 pixels and Gaussian noise with standard deviation 0.01, see Figure 5(b). The regularization parameter α has been chosen as 0.0005, such

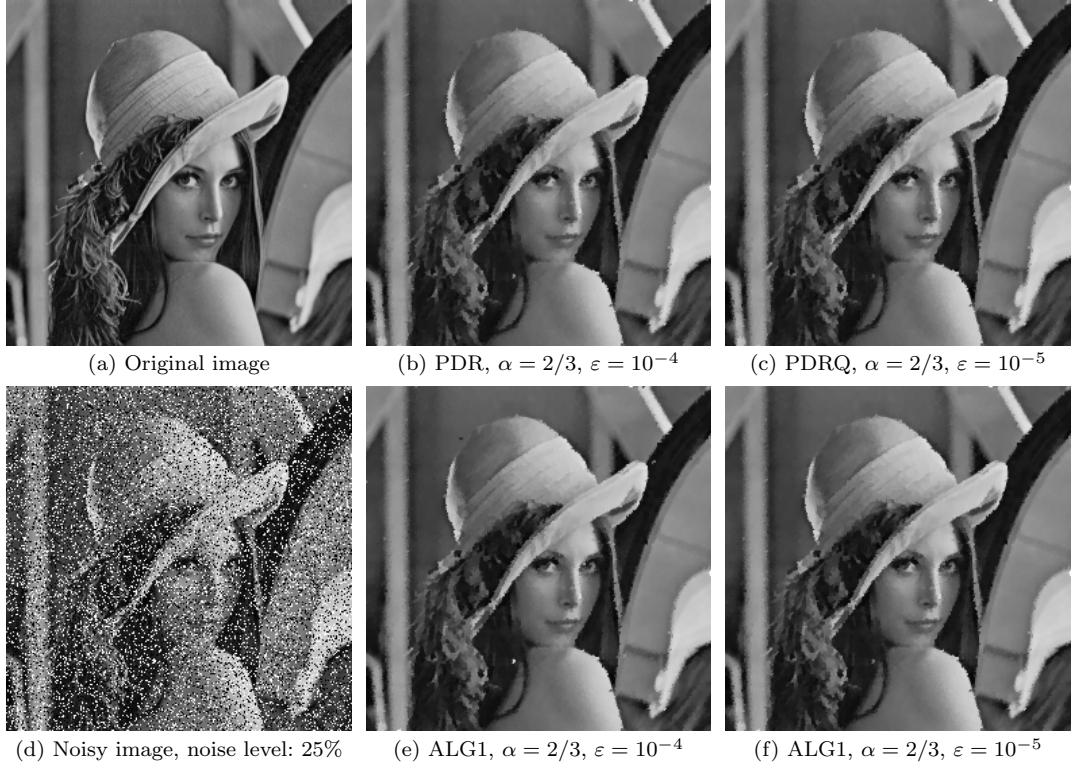
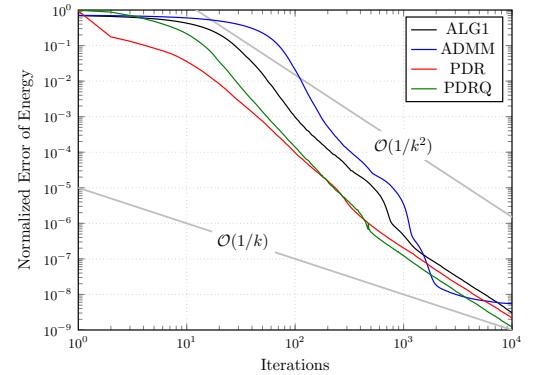


Fig. 3: L^1 -TV denoising: Results. (a) shows the 256×256 input Lena image while (d) is a noisy image for which 25% of the pixels are corrupted by salt-and-pepper noise. (b) and (c) are the denoised images by the PDRQ algorithm with regularization parameter $\alpha = 2/3$ and different tolerances ε for the normalized error of energy. (e) and (f) are the respective outcomes of Chambolle-Pock's ALG1 algorithm. Note that the solutions are not unique.

$\alpha = 2/3$		
	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-5}$
ALG1	214 (1.34s)	528 (2.95s)
ADMM	250 (8.07s)	617 (21.54s)
PDR	100 (0.69s)	235 (1.69s)
PDRQ	111 (0.65s)	227 (1.39s)
		$\varepsilon = 10^{-6}$
		779 (4.31s)
		903 (32.06s)
		475 (3.66s)
		450 (3.08s)

(a) Comparison of iteration numbers and CPU time cost.



(b) Numerical convergence rate compared with iteration number.

Fig. 4: L^1 -TV denoising: Performance table and numerical convergence rates. (a) shows the performance of the four algorithms in terms of iteration numbers and CPU times. (b) depicts the numerical performance rate in a double-logarithmic scale.

that TV-regularization is effective, see Figure 5 for the results, in particular a comparison to the Wiener filter as well as a table with performance results. Figure 6 shows numerical convergence rates and CPU timings. Note that the proposed PDRQ outperforms ALG1 with appropriate step size. Also, both PDRQ and ALG1 converge fast, however, PDRQ needs less iterations.

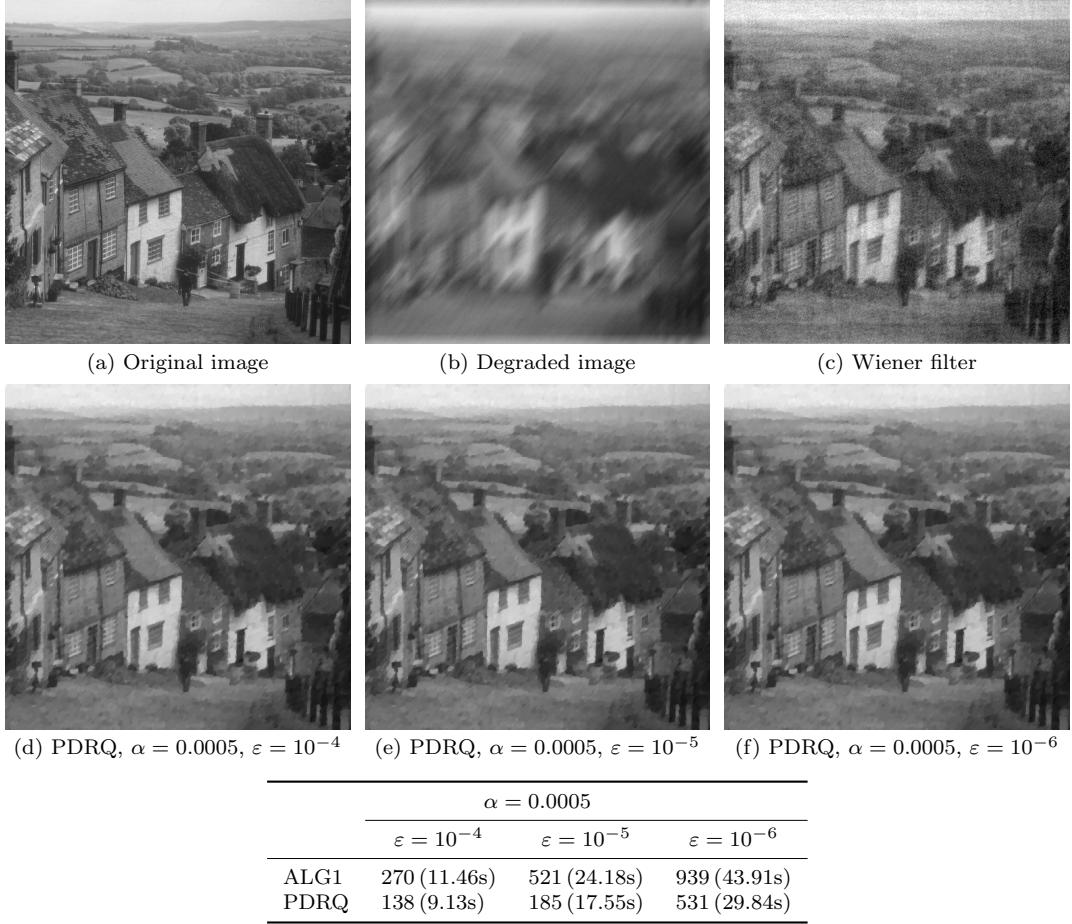


Fig. 5: L^2 -TV deblurring: Results and performance. (a) shows the 512×512 input image and (b) is the degraded image (≈ 40 pixels motion blur, Gaussian noise, standard deviation 0.01). (c) is the result of Wiener filtering and (d), (e) and (f) are the results of the PDRQ algorithm with normalized error of energy less than ε . The table shows the iteration number and CPU time needed for the normalized error of energy to get below ε .

6.2.2 L^1 -Deblurring

Computations were performed for the TV-deblurring algorithm with L^1 discrepancy using the following algorithms and setup.

- ALG1: $\mathcal{O}(1/k)$ primal-dual algorithm with constant step sizes, dual step size $\tau = 0.05$, $\tau\sigma L^2 = 1$, $\theta = 1$ with $L = \sqrt{9}$ as proposed in [6].

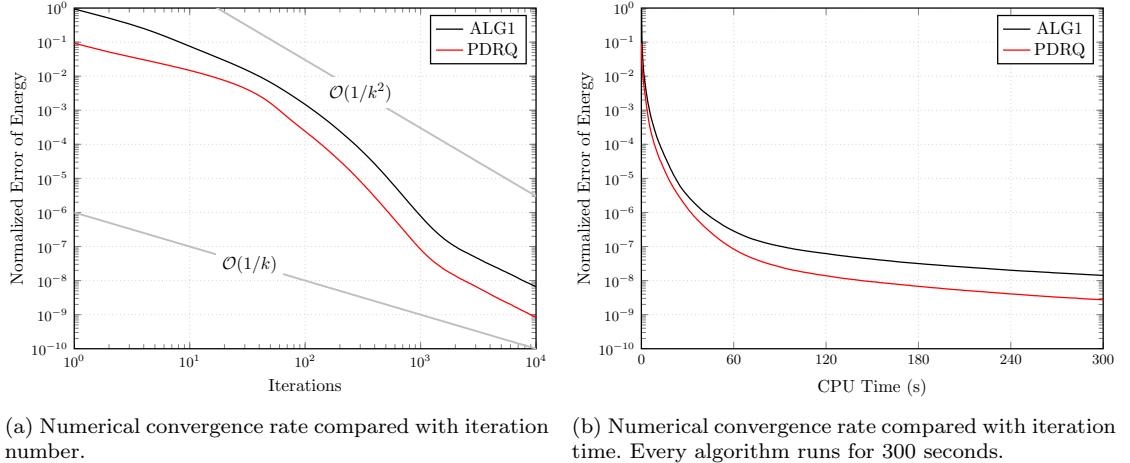


Fig. 6: L^2 -TV deblurring: Numerical convergence rates. The plots show a comparison of iteration numbers and CPU time. Again, a double-logarithmic and semi-logarithmic scale, respectively, is used in the plots.

- PDRQ: The scaled preconditioned Douglas-Rachford splitting method with preconditioner $M = \sigma^2 L^* L - (\sigma\tau)^2 \Delta_p$ according to Table 9, step-size $\sigma = 5$ and scaling parameter $\tau = 1/\sigma$.

The algorithm again stops once the normalized error of energy falls below a certain tolerance.

Computations were carried out for the 490×490 original image in Figure 7(a) subjected to motion blur of 30 pixels and corruption of 50% of the pixels by salt-and-pepper noise, see 7(d). With α chosen as 0.1, the PDRQ algorithm already produces visually appealing reconstructions after 50 iterations, see Figure 7 for an overview of the results. Performance values and plots can be found in Table 8 and represents the iteration numbers and iteration time cost for different stopping values. It could also be seen that a significant speed-up with respect to time is achievable with PDRQ. The numerical convergence rate of PDRQ and ALG1 seems to be faster than $\mathcal{O}(1/k)$ and a little slower than $\mathcal{O}(1/k^2)$.

6.3 TGV-Denoising Problems

6.3.1 L^2 -Denoising

The following algorithms and the parameter settings are used in our numerical experiments and comparison.

- ALG1: $\mathcal{O}(1/k)$ primal-dual algorithm discussed in [1, 6] with constant step sizes, the dual step size $\tau = 0.05$, $\tau\sigma L^2 = 1$, $\theta = 1$ with $L = \sqrt{12}$.
- PDRQ: The preconditioned Douglas-Rachford splitting method for quadratic problem according to Table 10. Here, 1 inner iteration for the block symmetric Red-Black Gauss-Seidel preconditioner is performed. The step size is chosen as σ and the method is denoted by PDRQ(σ) in the evaluations.

Stopping criterion is the normalized modified primal-dual gap $\mathfrak{G}_{L^2-\text{TGV}}^{C_w}/(N_x N_y)$ according to (35) with C_w , the L^1 -estimate for w , updated according to (36).

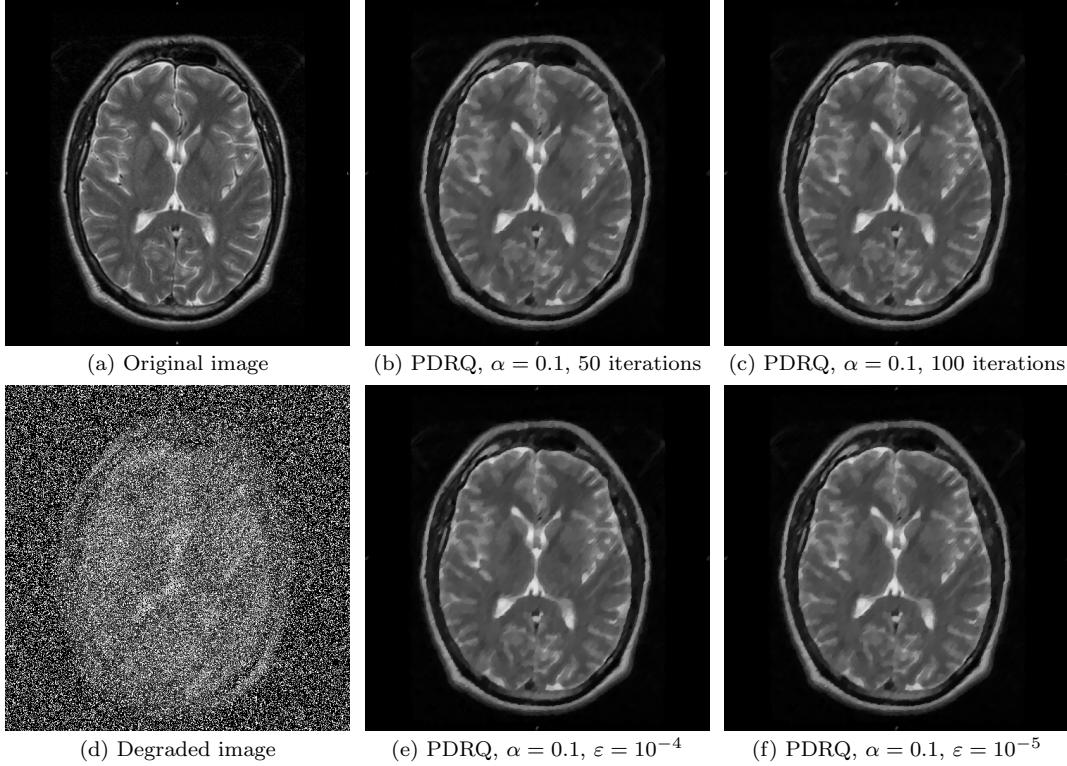


Fig. 7: L^1 -TV deblurring: Results. (a) shows the 490×490 input brain image which is gratefully taken from [9]. (d) is the degraded image with motion blur of 30 pixels and 50% of salt-and-pepper noise corruption. (b) and (c) are the results of by the PDRQ algorithm after 50 and 100 iterations. (e) and (f) show the reconstructions for the normalized error of energy less than ε .

We tested the algorithms on the 512×357 pixel image shown in Figure 9(a) subjected to 5% and 10% percent Gaussian noise, see Figure 9(d). Choosing $\alpha_0 = 2\alpha_1$ as in [5], we performed L^2 -TGV denoising with parameters $\alpha_1 = 0.05$ and $\alpha_1 = 0.1$, respectively, depending on the noise level. Results and a numerical performance table are depicted in Figure 9. Figure 10 shows a detailed comparison with respect to iteration number and iteration time for the two algorithms. It may be seen PDRQ is faster than ALG1, in particular if one aims at highly accurate numerical solutions. The plot also reveals that the normalized primal-dual gap admits oscillations. This numerical instability can be regarded as the price one has to pay for a provable upper bound for the distance to the minimizer as well as for the distance $\frac{1}{2N_x N_y} \|u^k - u^*\|_2^2$,

6.3.2 L^1 -Denoising

For the final denoising experiment, we again compare PDR and ALG1 with the following parameters.

- ALG1: $\mathcal{O}(1/k)$ primal-dual algorithm discussed in [1, 6] with constant step sizes, dual step size $\tau = 0.01$, $\tau\sigma L^2 = 1$, $\theta = 1$ with $L = \sqrt{12}$.

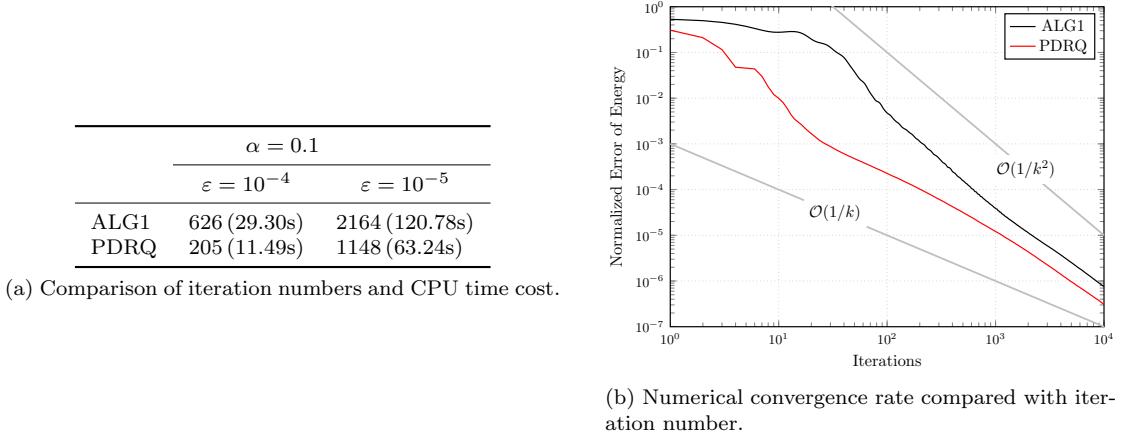
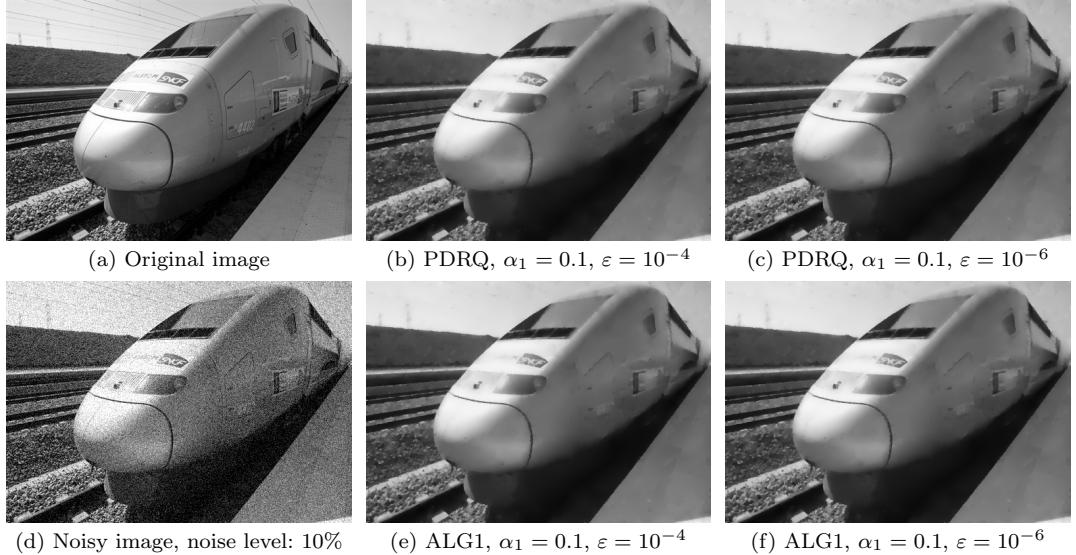
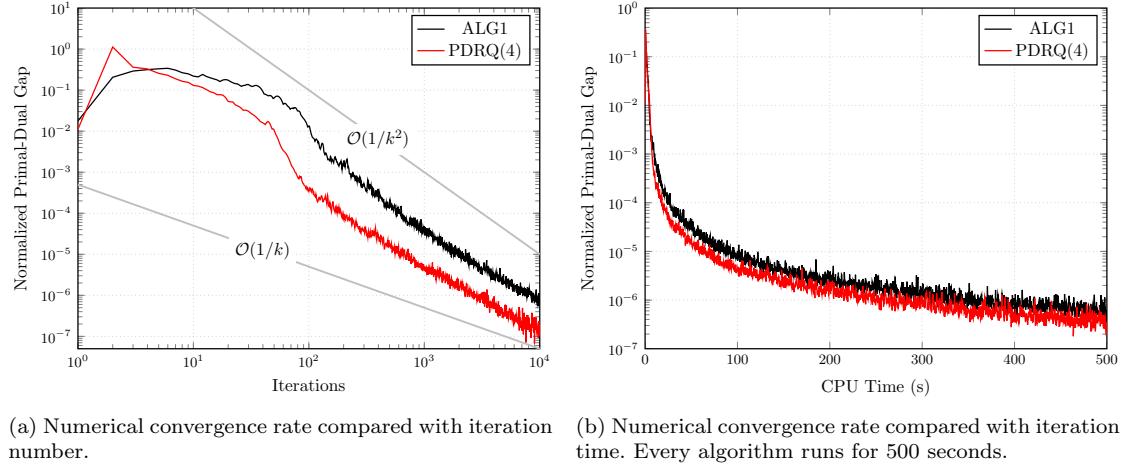


Fig. 8: L^1 -TV deblurring: Performance table and numerical convergence rates. (a) shows the performance of the compared algorithms in terms of iteration numbers and CPU times. (b) depicts the numerical performance rate in a double-logarithmic scale.



$\alpha_1 = 0.05$, noise level: 5%		$\alpha_1 = 0.1$, noise level: 10%	
$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$
ALG1	338 (14.65s)	3867 (167.44s)	544 (23.59s)
PDRQ	(3,1) 146 (14.05s)	(3,1) 1449 (139.27s)	(4,1) 190 (18.13s)

Fig. 9: L^2 -TGV denoising: Results and performance. (a) shows the 512×357 input train image and (d) is a noisy image which has been corrupted by 10% Gaussian noise. (b) and (c) are the denoised images by the PDRQ(4) algorithm with normalized primal-dual gap function less than ε . (e) and (f) are the respective outcomes of Chambolle-Pock's ALG1 algorithm. The table shows iteration number and computation time $k(t)$ for both algorithms and noise level of 5% and 10%. For PDRQ, step size and inner iterations for the preconditioner are denoted by (σ, n) .



(a) Numerical convergence rate compared with iteration number. (b) Numerical convergence rate compared with iteration time. Every algorithm runs for 500 seconds.

Fig. 10: L^2 -TGV denoising: Numerical convergence rates. The plots show a comparison of iteration numbers and CPU time. Again, a double-logarithmic and semi-logarithmic scale, respectively, is used in the plots.

- PDR: The scaled preconditioned Douglas-Rachford splitting method with symmetric block Red-Black Gauss-Seidel preconditioner according to Table 11. The step size is chosen as $\sigma = 0.1$, the scaling τ as $3/\sigma$ and 1 inner iteration for the preconditioner is performed.

For comparison, the iteration is stopped when the normalized error of energy $(E^k - E^*)/E^*$ falls below a tolerance where, again, E^* is precomputed.

The L^1 -TGV denoising algorithms were tested on the 510×383 image in Figure 11(a) for which 30% of the pixels have been corrupted by salt-and-pepper noise (Figure 11(d)). With the regularization chosen as $\alpha_1 = 1.0$ and $\alpha_0 = 2\alpha_1$, results for the compared methods and different stopping tolerances are depicted in Figure 11 while a performance table and a plot showing numerical performance rates can be found in Figure 12. Here, it can be seen that PDR is significantly faster than ALG1 with respect to iteration number and iteration time with speed-ups ranging between 2 to 3.

7 Discussion and Conclusions

We derived and investigated numerically concrete preconditioned Douglas-Rachford methods for various discrete variational imaging problems. They all base on the abstract algorithms introduced in [3]. Different preconditioning strategies for both TV and TGV regularized problems have been developed. While for TV-regularized denoising, the symmetric Red-Black Gauss-Seidel is an effective preconditioner, we proposed a respective symmetric block Red-Black Gauss-Seidel iteration for TGV-based denoising problems. In both cases, these preconditioners are tailored for the underlying differential structure which is an discrete elliptic equation in the TV-case and an discrete elliptic system of PDEs in the TGV-case. For L^2 -TGV denoising problems, we moreover derived an effective primal-dual gap which gives a provable bound to the distance in functional values and which can also be used to estimate the L^2 norm to the optimal solution.

The numerical results show that PDR and its variants have the potential to bring out appealing benefits and fast algorithms, provided that efficient preconditioners are found. The algorithm

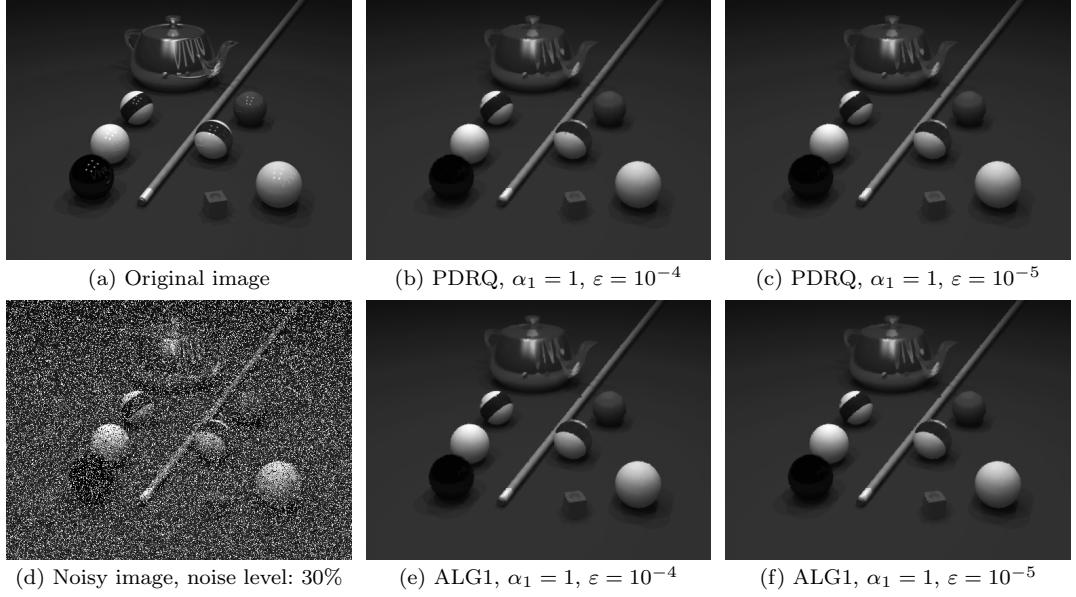
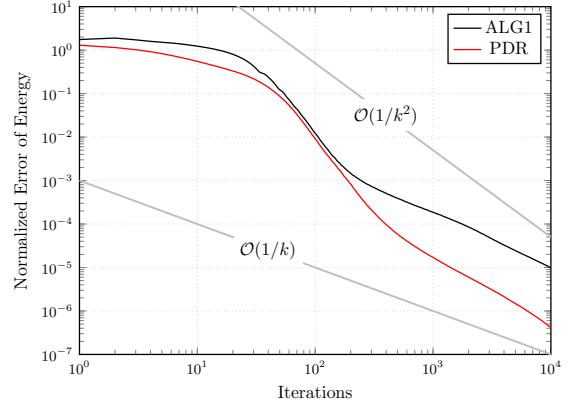


Fig. 11: L^1 -TGV denoising: Results. (a) shows the 510×383 original image while (d) is its noisy version which has been corrupted by 30% salt-and-pepper noise. (b) and (c) are the denoised images by the PDR algorithm with the stopping criterion of the normalized error of primal energy less than ε . (e) and (f) are the respective outcomes of Chambolle-Pock's ALG1 algorithm.

$\alpha_1 = 1.0$			
	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-5}$	
ALG1	1761 (73.34s)	9897 (414.32s)	
PDR	396 (35.92s)	1408 (126.84s)	

(a) Comparison of iteration numbers and CPU time cost.



(b) Numerical convergence rate compared with iteration number.

Fig. 12: L^1 -TGV denoising: Performance table and numerical convergence rates. (a) shows the performance of the compared algorithms in terms of iteration numbers and CPU times. (b) depicts the numerical performance rate in a double-logarithmic scale.

are most beneficial in situations where accelerated primal-dual methods can not be applied which is the case for problems which are not strictly convex. This is the case for L^1 discrepancies as well as for TGV regularization. Indeed, significant speed-ups were achieved in these situations. Nevertheless, there are some open questions that need to be figured out. The possibility to accelerate

the convergence speed of PDR and its variants to the optimal convergence speed of first-order methods, i.e., $\mathcal{O}(1/k^2)$, is, for instance, an important aspect to investigate in the future.

Acknowledgements The work of Kristian Bredies and Hongpeng Sun is supported by the Austrian Science Fund (FWF) under grant SFB32 (SFB “Mathematical Optimization and Applications in the Biomedical Sciences”).

References

1. Bredies K (2014) Recovering piecewise smooth multichannel images by minimization of convex functionals with total generalized variation penalty. In: Bruhn A, Pock T, Tai XC (eds) Efficient Algorithms for Global Optimization Methods in Computer Vision, Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp 44–77
2. Bredies K, Holler M (2012) Artifact-free JPEG decompression with total generalized variation. Proceedings of VISAPP 2012 - International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications pp 12–21
3. Bredies K, Sun HP (2014) Preconditioned Douglas-Rachford splitting methods for convex-concave saddle-point problems. SFB-Report-2014-002, University of Graz
4. Bredies K, Valkonen T (2011) Inverse problems with second-order total generalized variation constraints. Proceedings of SampTA 2011 - 9th International Conference on Sampling Theory and Applications, Singapore
5. Bredies K, Kunisch K, Pock T (2010) Total generalized variation. SIAM Journal on Imaging Sciences 3(3):492–526
6. Chambolle A, Pock T (2011) A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision 40(1):120–145
7. Ekeland I, Témam R (1999) Convex Analysis and Variational Problems. No. 1 in Classics in Applied Mathematics, Society for Industrial and Applied Mathematics
8. Esser E (2009) Applications of Lagrangian-based alternating direction methods and connections to split Bregman. Tech. rep., CAM report 09-2009, UCLA
9. Farrall A (2011) Human brain MRI. <http://www.anatomy.mvm.ed.ac.uk/museum/explore-anatomy.php>, [Online; accessed 07-February-2014]
10. Goldstein T, Osher S (2009) The split Bregman method for L^1 -regularized problems. SIAM Journal on Imaging Sciences 2(2):323–343
11. Ito K, Kunisch K (2008) Lagrange Multiplier Approach to Variational Problems and Applications. Advances in Design and Control, Society for Industrial and Applied Mathematics
12. Knoll K, Bredies K, Pock T, Stollberger R (2011) Second order total generalized variation (TGV) for MRI. Magnetic Resonance in Medicine 65(2):480–491
13. Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60(1-4):259–268
14. Temam R (1985) Mathematical Problems in Plasticity. Bordas
15. Thomas JW (1999) Numerical Partial Differential Equations: Conservation Laws and Elliptic Equations, Texts in Applied Mathematics, vol 33. Springer New York
16. Valkonen T, Bredies K, Knoll F (2013) Total generalized variation in diffusion tensor imaging. SIAM Journal on Imaging Sciences 6(1):487–525