SMAI-JCM
SMAI Journal of Computational Mathematics

A General Decomposition Method for a Convex Problem Related to Total Variation Minimization

Stephan Hilb & Andreas Langer

Volume 11 (2025), p. 693-726.
https://doi.org/10.5802/smai-jcm.140

The SMAI Journal of Computational Mathematics is a member of the Centre Mersenne for Open Scientific Publishing
http://www.centre-mersenne.org/
Submissions at https://smai-jcm.centre-mersenne.org/ojs/submission
e-ISSN: 2426-8399

SMAI Journal of Computational Mathematics
Vol. 11, 693-726 (2025)

A General Decomposition Method for a Convex Problem Related to Total Variation Minimization

Stephan Hilb [1]
Andreas Langer [2]

[1] Stuttgart, Germany
E-mail address: stephan@ecshi.net
[2] Department for Mathematical Sciences, Lund University, Sweden
E-mail address: andreas.langer@math.lth.se.

Abstract. We consider sequential and parallel decomposition methods for a dual problem of a general total variation minimization problem with applications in several image processing tasks, like image inpainting, estimation of optical flow and reconstruction of missing wavelet coefficients. The convergence of these methods to a solution of the global problem is analysed in a Hilbert space setting and a convergence rate is provided. Thereby, these convergence result hold not only for exact local minimization but also if the subproblems are just solved approximately. As a concrete example of an approximate local solution process a surrogate technique is presented and analysed. Further, the obtained convergence rate is compared with related results in the literature and shown to be in agreement with or even improve upon them. Numerical experiments are presented to support the theoretical findings and to show the performance of the proposed decomposition algorithms in image inpainting, optical flow estimation and wavelet inpainting tasks.

2020 Mathematics Subject Classification. 68U10, 94A08, 49M27, 65K10, 90C06.

Keywords. domain decomposition, total variation minimization, convex optimization, image restoration, convergence analysis, subspace correction.

1. Introduction

The dimensionality of images has been tremendously increased in recent years due to the improvement of hardware. In order to further post-process such large-scale data in a distributed parallel or memory-constrained setting, decomposition methods may be used, which split the original problem into a sequence of smaller subproblems that can be solved independently of each other while still approaching the original solution by means of an iterative algorithm. One particular approach are domain decomposition algorithms [14, 45, 47] which subdivide the problem domain. Typical examples of post-processing images include the removal of noise (denoising), the completion of missing data (inpainting) and the analysis of the data, as the computation of the optical flow in image sequences. In such applications one is usually interested in solutions in which edges are preserved. The total variation (TV) is well-know to promote discontinuities and hence is widely used in image processing tasks. Thereby one may consider the following regularized TV-model, cf. [22],

$$\inf_{u\in L^2(\Omega)^m\cap BV(\Omega)^m} \frac{1}{2}\|Tu-g\|^2_{L^2(\Omega)} + \frac{\beta}{2}\|u\|^2_{L^2(\Omega)^m} + \lambda\int_\Omega |Du|, \tag{1.1}$$

This work was partly funded by the Ministerium für Wissenschaft, Forschung und Kunst of Baden-Württemberg (Az: 7533.-30-10/56/1) through the RISC-project "Automatische Erkennung von bewegten Objekten in hochauflösenden Bildsequenzen mittels neuer Gebietszerlegungsverfahren" and by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2075 – 390740016.
https://doi.org/10.5802/smai-jcm.140

where $\Omega \subseteq \mathbb{R}^d$, $d \in \mathbb{N}$, is an open, bounded and simply connected domain with Lipschitz boundary, $g \in L^2(\Omega)$ describes the observed data, $T : L^2(\Omega)^m \to L^2(\Omega)$ with $m \in \mathbb{N}$ is a linear bounded operator, $\beta \geq 0$, $\lambda > 0$, and $\int_\Omega |Du|$ denotes the total variation of u in Ω defined by

$$\int_\Omega |Du| := \sup \left\{ \int_\Omega u \cdot \operatorname{div} v \, \mathrm{d}x \; : \; \begin{array}{l} v \in (C_0^\infty(\Omega))^{d\times m}, \\ |v(x)|_F \leq 1 \text{ for almost every (f.a.e.) } x \in \Omega \end{array} \right\}, \tag{1.2}$$

with $|\cdot|_F : \mathbb{R}^{d\times m} \mapsto \mathbb{R}$ being the Frobenius norm, cf. [21]. The operator $\operatorname{div} : H_0^{\operatorname{div}}(\Omega)^m \to (L^2(\Omega))^m$ describes the divergence with respect to d (i.e. column-wise), where $H_0^{\operatorname{div}}(\Omega)^m = \{v \in L^2(\Omega)^{d\times m} : \operatorname{div} v \in L^2(\Omega)^m\}$. We recall that $BV(\Omega)^m$, i.e. the space of functions with bounded variation, equipped with the norm $\|\cdot\|_{BV} := \|\cdot\|_{L^1(\Omega)} + \int_\Omega |D\cdot|$ is a Banach space [2, Theorem 10.1.1]. Note that $m \in \mathbb{N}$ describes the number of output channels, e.g. for grey-scale images we set $m = 1$ while for motion fields we have $m = d$.

The crucial difficulty of deriving decomposition methods for total variation minimization problems lies in the fact that the total variation is non-differentiable and non-additive with respect to a disjoint splitting of the domain Ω. In fact, let Ω_1 and Ω_2 be a disjoint decomposition of Ω, then we have the following splitting property, cf. [1, Theorem 3.84],

$$\int_\Omega |D(u_{|\Omega_1} + u_{|\Omega_2})| = \int_{\Omega_1} |D(u_{|\Omega_1})| + \int_{\Omega_2} |D(u_{|\Omega_2})| + \int_{\partial\Omega_1 \cap \partial\Omega_2} |u^+_{|\Omega_1} - u^-_{|\Omega_2}| \, \mathrm{d}\mathcal{H}^{d-1}(x), \tag{1.3}$$

where $\mathcal{H}^d$ denotes the Hausdorff measure of dimension d and the symbols u^+ and u^- the "interior" and "exterior" trace of u on $\partial\Omega_1 \cap \partial\Omega_2$ respectively. That is the total variation of a function of the whole domain equals the sum of the total variation on the subdomains plus the size of the possible jumps at the interface. Exactly these jumps at the interfaces are important as we want to preserve crossing discontinuities and the correct matching where the solution is continuous. A failure of a decomposition method for total variation minimization has been reported in [17] with respect to a wavelet space decomposition. There a condition is derived which allows to check for global optimality of a limit point generated by the decomposition method. Although the method seems to work fine in practice, a counterexample showed in [17] that this condition does not hold in general. Nevertheless, this condition may be utilized in order to check a posteriori whether the splitting method found a good numerical approximation. First domain decomposition methods for total variation minimization are presented in [18, 19, 30]. Although their convergence and monotonic decay of the energy is theoretically ensured, the convergence to the solution of the global problem cannot be guaranteed in general, as counterexamples in [28, 31] illustrate. However, in [23, 24] an estimate of the distance of the numerical solution generated by such a decomposition method to the true minimizer of the original problem is derived. Utilizing this estimate demonstrated in [23, 24] that the splitting methods work in practice quite well for total variation minimization, as they indeed generate sequences for which this estimate indicates convergence to the global minimizer.

To overcome the difficulties due to the minimization of a non-smooth and non-additive objective in (1.1) a predual problem of (1.1), as in [10, 25] for the case $T = I$, $\beta = 0$, and $m = 1$, may be considered. In fact, a predual formulation of (1.1) can be derived which involves constrained minimization of a smooth functional [21].

Proposition 1.1 (cf. [21])**.** *Let $V := H_0^{\operatorname{div}}(\Omega)^m$, $W := L^2(\Omega)^m$. Problem* (1.1) *is dual (in the sense of Fenchel duality* [16]*) to*

$$\inf_{p\in K} \left\{ \mathcal{D}(p) := \frac{1}{2}\|\Lambda p - T^* g\|^2_{B^{-1}} \right\}, \tag{1.4}$$

where $K := \{p \in V : |p(x)|_F \leq \lambda$ f.a.e. $x \in \Omega\}$, $\Lambda : V \to W$, $\Lambda p = \operatorname{div} p$, $T^* : L^2(\Omega) \to W$ is the adjoint operator of T, $B : W \to W$ denotes the operator $B := T^*T + \beta I$ and the norm is given by $\|u\|^2_{B^{-1}} := \langle u, B^{-1}u\rangle_W$ for $u \in W$, where $\langle\cdot,\cdot\rangle_W$ denotes the W-inner product.

The unique solution $\widehat{u}$ *of* (1.1) *is related to any solution* $\widehat{p}$ *of* (1.4) *by*

$$\widehat{u} = B^{-1}(-\Lambda\widehat{p} + T^*g) \qquad \text{and} \qquad \forall\, p \in K : \langle \Lambda^*\widehat{u}, p - \widehat{p}\rangle_V \leq 0, \tag{1.5}$$

where $\Lambda^* : W^* \to V^*$ *is the adjoint operator of* Λ *and* $\langle\cdot,\cdot\rangle_V$ *denotes the duality pairing between* V *and its dual space* V^*.

To guarantee the existence of a minimizer of (1.4) in the rest of the paper we assume that the bilinear form $a_B : W \times W \to \mathbb{R}$, $a_B(u,v) := \langle Tu, Tv\rangle_{L^2(\Omega)} + \beta\langle u, v\rangle_W$ is coercive [21, Theorem 3.2], i.e. $a_B(u,u) \geq c_B\|u\|_W^2$ with coercivity constant $c_B > 0$ and $\|\cdot\|_W$ being the norm induced by the W-inner product. In particular this implies the coercivity of $\mathcal{D}$, i.e. for any feasible sequence $(p^n)_{n\in\mathbb{N}} \subseteq K$

$$\|p^n\|_V \to \infty \implies \mathcal{D}(p^n) \to \infty \tag{1.6}$$

with $\|\cdot\|_V$ being the norm associated to V [21]. Further the coercivity of a_B guarantees the invertibility of B [21]. Proposition 1.1 allows one to solve for $\widehat{p}$ in the predual domain of (1.4) and to later assemble the original solution $\widehat{u}$ using the optimality relation (1.5). Note that although (1.4) does not have a unique solution in general, the relation (1.5) still provides the possibility of obtaining the unique minimizer of (1.1).

Here and in the sequel we write V and W instead of $H_0^{\mathrm{div}}(\Omega)^m$ and $L^2(\Omega)^m$ as the below presented algorithms (see Algorithm 1 and Algorithm 2) as well as the associated theory also holds for problems of type (1.4) in a general Hilbert space setting, see Remark 6.6 below. We similarly write Λ in place of div to reflect the abstract setting with general bounded linear operators; see again Remark 6.6.

Based on a dualization as in Proposition 1.1 for the setting $T = I$, $\beta = 0$ and $m = 1$ in [10, 25] convergent overlapping and nonoverlapping domain decomposition methods are introduced. While the convergence in [25] for a nonoverlapping splitting is proven in a discrete setting, in [10] for an overlapping splitting even a convergence rate in a continuous setting is guaranteed. These two papers allowed to derive overlapping [29] and nonoverlapping [31] domain decomposition methods for the primal problem (1.1) in the respective setting and for $T = I$, $\beta = 0$ and $m = 1$, together with a convergence analysis which ensures that a minimizer of the global problem is indeed approached. Since then a series of splitting techniques for total variation minimization have been presented in the literature [32, 33, 34, 35, 37, 41, 42, 43]. For an introduction to domain decomposition approaches for total variation minimization we refer the reader to [28, 36].

In this paper we generalize the overlapping splitting method from [10], which is restricted to denoising, to the more general problem (1.4) where T can be any arbitrary linear and bounded operator, and hence to applications like inpainting and calculating the optical flow in image sequences. Further while the analysis of the decomposition method in [10] assumes exact local minimization, in our case an approximate local minimization is sufficient. This requires a new convergence analysis which differs significantly from the one in [10]. In particular, we utilize ideas from alternating minimization, see [7], which in the end even allow us to improve the convergence rate of [10] by a constant. We provide a particular example in which the subproblems are approximated (solved inexactly) by so-called *surrogate* functionals, as in [18, 19, 24]. For solving the subspace problems we adjust the semi-implicit algorithm by Chambolle [8] to our setting. While the algorithm in [8] is derived in a discrete setting and for image denoising problems only (i.e. $T = I$), we adjust it to our problem, where T might be any bounded and linear operator as already mentioned above, and Hilbert space setting.

We would like to mention that recently in [41] a general framework for analysing additive Schwarz methods of convex optimization problems as gradient methods has been presented. For the special case of parallel decomposition their result covers ours, while we extend our results to sequential decomposition which [41] does not cover. In particular, our analysis differs from the one in [41]. While in [41] the investigation is based on formulating the parallel decomposition as a gradient method, we use techniques from alternating minimization to analyze the decomposition method directly. This direct tackling of the convergence analysis allowed us additionally to cover the sequential decomposition as

well. We also consider a slightly different notion of approximate minimization (see Definition 3.1) for the local subproblems which does not seem to map to the approximate notion considered in [41] in an obvious way.

The rest of the paper is organized as follows: In Section 2 we introduce the setting of our decomposition which is based on the definition of a partition of unity operator. The proposed parallel and sequential decomposition algorithms are described in Section 3 and their convergence analysis is presented in Section 4. In particular we prove the convergence of the presented algorithms to a solution of the global problem together with a convergence rate. In Section 5 we compare our findings with related results already presented in the literature. As our proposed decomposition algorithms allow for approximate solutions of the subproblems, in Section 6 we present a concrete example for such a case utilizing the surrogate technique. For solving the constituted subdomain problems we describe in Section 7 both the semi-implicit algorithm of Chambolle [8] and the fast iterative shrinkage-thresholding algorithm (FISTA) by Beck and Teboulle [6] in a general Hilbert space setting for our type of problems. In Section 8 numerical experiments are presented verifying the theoretical sublinear convergence of the proposed decomposition algorithms as well as showing the practical behaviour. We conclude in Section 9 with some final remarks.

2. **Fundamentals**

As we are presenting a decomposition method for (1.4) in the sequel we use the notations and definitions of Proposition 1.1. Further for a bounded linear operator $A : H_1 \to H_2$ between two Hilbert spaces H_1 and H_2 we use $\|A\|$ for the respective operator norm. If H_1 and H_2 are function spaces, we say A is local if $\operatorname{supp}(A\varphi) \subseteq \operatorname{supp}(\varphi)$ for all $\varphi \in H_1$.

2.1. **Decomposition Setting**

We will analyze a decomposition algorithm for problem (1.4) by splitting the closed convex set K into subsets using a suitable partition of unity, with a splitting strategy similar to that in [46]. More precisely, we define a partition of unity by bounded linear operators $\theta_i : V \to V$, $i = 1, \dots, M \in \mathbb{N}$ such that

$$I = \sum_{i=1}^{M} \theta_i \qquad \text{and} \qquad K = \sum_{i=1}^{M} \theta_i K. \tag{2.1}$$

Note here that, since θ_i is a bounded linear operator, $\theta_i K = \{\theta_i p : p \in K\} \subseteq V$ stays closed and convex.

For such a partition of unity we define the decomposition neighbourhood graph $G = (V_G, E_G)$ with vertices $V_G = \{1, \dots, M\}$ and edges $(i, j) \in E_G$ whenever there exists $\varphi \in V$ such that $\operatorname{supp}(\theta_i\varphi) \cap \operatorname{supp}(\theta_j\varphi) \neq \emptyset$. We denote by $M_C \in \mathbb{N}$, $1 \leq M_C \leq M$ the chromatic number of G, i.e. the minimum number of colors required to color all vertices such that vertex pairs connected by edges have distinct colors.

Example 2.1. Under the assumption of Proposition 1.1 the requirements for the partition given in (2.1) are in particular fulfilled by the following domain decomposition formulation. Let Ω_i, $i = 1, \dots, M \in \mathbb{N}$ be bounded open sets with Lipschitz boundary such that $\bigcup_{i=1}^{M} \Omega_i = \Omega$. Denote by $\widetilde{\theta}_i : \Omega \to [0, 1]$, $i = 1, \dots, M$ a partition of unity satisfying

(i) $\widetilde{\theta}_i \in W^{1,\infty}(\Omega)$,
(ii) $1 = \sum_{i=1}^{M} \widetilde{\theta}_i$,
(iii) $\operatorname{supp} \widetilde{\theta}_i \subseteq \overline{\Omega}_i$.

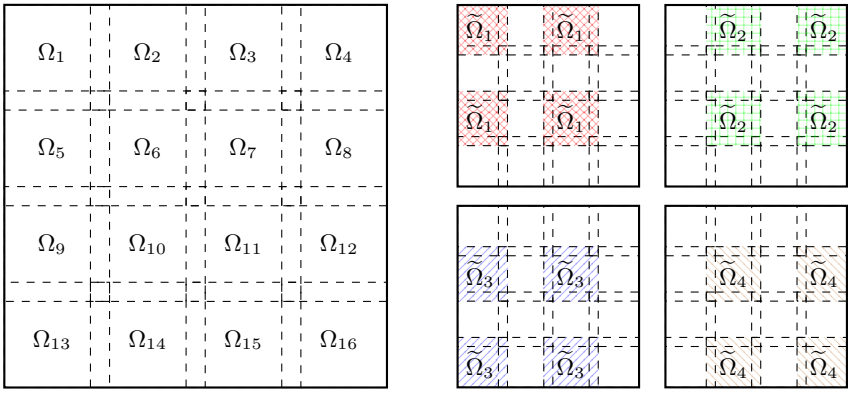


FIGURE 2.1. Decomposition of a rectangular spatial domain into $M = 16$ overlapping subdomains $\Omega_1, \ldots, \Omega_M$ and grouping of disjoint domains using the coloring technique into $M_C = 4$ domains $\widetilde{\Omega}_1, \ldots, \widetilde{\Omega}_{M_C}$

We then define the partition of unity operator $\theta_i : V \to V$ by pointwise multiplication

$$(\theta_i p)(x) := \widetilde{\theta}_i(x) p(x), \tag{2.2}$$

for all $p \in V$. This yields an overlapping domain decomposition such that for any $i \in \{1, \ldots, M\}$ there exists at least one $j \in \{1, \ldots, M\} \setminus \{i\}$ with $\Omega_i \cap \Omega_j \neq \emptyset$. An example of a decomposition into $M = 16$ overlapping subdomains is shown in Figure 2.1.

Lemma 2.2. *Under the assumptions of Proposition 1.1 the partition of unity operators* $(\theta_i)_{i=1}^M$ *defined in* (2.2) *satisfy the requirements of* (2.1).

Proof. Linearity of θ_i, $i = 1, \ldots, M$, is inherited from the pointwise multiplicative definition in (2.2). Let $p \in V = H_0^{\mathrm{div}}(\Omega)^m$, then

$$\begin{aligned} \|\widetilde{\theta}_i p\|_{L^2} &\leq \|\widetilde{\theta}_i\|_{L^\infty} \|p\|_{L^2}, \\ \|\operatorname{div}(\widetilde{\theta}_i p)\|_{L^2} &= \|\nabla \widetilde{\theta}_i p + \widetilde{\theta}_i \operatorname{div} p\|_{L^2} \\ &\leq \|\nabla \widetilde{\theta}_i\|_{L^\infty} \|p\|_{L^2} + \|\widetilde{\theta}_i\|_{L^\infty} \|\operatorname{div} p\|_{L^2}, \end{aligned}$$

and thus, since $\widetilde{\theta}_i \in W^{1,\infty}(\Omega)$ and in particular $\nabla \widetilde{\theta}_i \in L^\infty(\Omega)$, we have proven that $\theta_i : V \to V$ is indeed well-defined and bounded.
Due to the pointwise nature of (2.2) we see that for $p \in V$:

$$\left(\sum_{i=1}^M \theta_i p \right)(x) = \sum_{i=1}^M \widetilde{\theta}_i(x) p(x) = p(x)$$

which shows $I = \sum_{i=1}^M \theta_i$.
We have $K \subseteq \sum_{i=1}^M \theta_i K$ per definition. To show the other inclusion let $p^i \in \theta_i K$, $i = 1, \ldots, M$. Then we see that in a pointwise fashion

$$\left| \sum_{i=1}^M p^i(x) \right|_F \leq \sum_{i=1}^M |p^i(x)|_F \leq \sum_{i=1}^M \widetilde{\theta}_i(x) \lambda \leq \lambda,$$

thus showing that $\sum_{i=1}^M p^i \in K$. ∎

3. Algorithm

Let us first introduce our notion of approximate minimization.

Definition 3.1. For $q \in V$, $\rho \in (0,1]$ we call

$$\operatorname*{arg\,min}_{p\in K}^{\rho,q} \mathcal{D}(p) := \left\{ \widetilde{p} \in K : \mathcal{D}(q) - \mathcal{D}(\widetilde{p}) \geq \rho(\mathcal{D}(q) - \mathcal{D}(\widehat{p})), \widehat{p} \in \operatorname*{arg\,min}_{p\in K} \mathcal{D}(p) \right\} \tag{3.1}$$

the set of ρ-approximate minimizers of $\mathcal{D}$ on K with respect to q.

The condition in (3.1) means that the improvement in functional value needs to be at least within a constant factor of the remaining difference in functional value towards a true minimizer. For $\rho = 1$ and arbitrary $q \in V$ this reduces to the usual notion of minimizers.

We present the decomposition procedures in Algorithms 1 and 2.

Algorithm 1 Parallel decomposition

Require: $p^0 \in K$ and $\sigma \in (0, \frac{1}{M}]$, $\rho \in (0,1]$
1: **for** $n = 0, 1, 2, \ldots$ **do**
2: **for** $i = 1, \ldots, M$ **do**
3: $\widetilde{v}_i^n \in \arg\min_{v_i \in \theta_i K}^{\rho, \theta_i p^n} \mathcal{D}(p^n + (v_i - \theta_i p^n))$
4: **end for**
5: $p^{n+1} = p^n + \sum_{i=1}^M \sigma(\widetilde{v}_i^n - \theta_i p^n)$
6: **end for**

Algorithm 2 Sequential decomposition

Require: $p^0 \in K$ and $\sigma \in (0,1]$, $\rho \in (0,1]$
1: **for** $n = 0, 1, 2, \ldots$ **do**
2: $p_0^n = p^n$
3: **for** $i = 1, \ldots, M$ **do**
4: $\widetilde{v}_i^n \in \arg\min_{v_i \in \theta_i K}^{\rho, \theta_i p^n} \mathcal{D}(p_{i-1}^n + (v_i - \theta_i p^n))$
5: $p_i^n = p_{i-1}^n + \sigma(\widetilde{v}_i^n - \theta_i p^n)$
6: **end for**
7: $p^{n+1} = p_M^n$
8: **end for**

Note that by grouping disjoint subdomains using a coloring on the decomposition neighbourhood graph, one may equivalently use $M = M_C$ for Algorithm 1 and thus alleviate the constraint on σ. For Algorithm 2 such grouping can be beneficial as well to enable parallel minimization of the subproblems within each group, this does however yield a different ordering of the subproblems (lexicographical per color), leading to a not-necessarily equivalent algorithm.

To treat both algorithms in a similar way, we use the convention $p_{i-1}^n := p^n$ for Algorithm 1 independent of $i \in \{1, \ldots, M\}$. Having defined $\widetilde{v}_i^n \in \theta_i K$ for $i \in \{1, \ldots, M\}$ we also set $\widetilde{p}_i^n := p_{i-1}^n + (\widetilde{v}_i^n - \theta_i p^n) \in K$.

We observe that in each step $n \in \mathbb{N}_0$, $i \in \{1, \ldots, M\}$ of Algorithms 1 and 2 the subproblem

$$\inf_{v_i \in \theta_i K} \frac{1}{2} \|\Lambda v_i - f_i^n\|_{B^{-1}}^2 \tag{3.2}$$

with $f_i^n = T^* g - \Lambda(p_{i-1}^n - \theta_i p^n)$ needs to be solved approximately. For local operators Λ and B^{-1} these problems may be solved on each subdomain $\operatorname{supp}(\theta_i) := \overline{\bigcup_{\varphi \in V} \operatorname{supp}(\theta_i \varphi)} \subseteq \overline{\Omega}$, as all quantities

in (3.2) can be suitably restricted. If Λ or B^{-1} on the other hand are non-local then in order to avoid having to solve the subproblems globally on Ω a surrogate technique will be introduced in Section 6 below.

4. **Convergence Analysis**

In this section we analyse Algorithms 1 and 2 with respect to their convergence. In particular we first show monotonicity of the energy with respect to the iterates followed by the main results. Subsequently we collect useful statements which finally enable us to prove our main result at the end of this section.

4.1. **Monotonicity**

We first establish monotonicity of the iterates.

Lemma 4.1. *The iterates $(p^n)_{n\in\mathbb{N}}$ of Algorithms 1 and 2 with corresponding constraints on σ satisfy*

$$\mathcal{D}(p^n) - \mathcal{D}(p^{n+1}) \geq \rho\sigma \sum_{i=1}^{M} \left(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widehat{p}_i^n)\right) \geq 0,$$

where $\widehat{p}_i^n := p_{i-1}^n + (\widehat{v}_i^n - \theta_i p^n)$, $\widehat{v}_i^n \in \arg\min_{v_i \in \theta_i K} \mathcal{D}(p_{i-1}^n + (v_i - \theta_i p^n))$ denotes any exact minimizer in the i-th substep of the corresponding algorithm. The non-negative sequence $(\mathcal{D}(p^n))_{n\in\mathbb{N}}$ is in particular monotonically decreasing and thus convergent.

Proof. The update step for p^{n+1} in the parallel case of Algorithm 1 is given as

$$p^{n+1} = p^n + \sigma \sum_{i=1}^{M} (\widetilde{v}_i^n - \theta_i p^n) = (1 - \sigma M) p^n + \sigma \sum_{i=1}^{M} \left(p^n + (\widetilde{v}_i^n - \theta_i p^n)\right).$$

We denote $\widetilde{p}_i^n = p_{i-1}^n + (\widetilde{v}_i^n - \theta_i p^n) = p^n + (\widetilde{v}_i^n - \theta_i p^n)$. Since $\sigma \in (0, \frac{1}{M}]$, convexity of $\mathcal{D}$ yields

$$\mathcal{D}(p^{n+1}) \leq (1 - \sigma M)\mathcal{D}(p^n) + \sigma \sum_{i=1}^{M} \mathcal{D}(\widetilde{p}_i^n).$$

We use this and the definition of $\widetilde{v}_i^n$ to estimate

$$\begin{aligned}\mathcal{D}(p^n) - \mathcal{D}(p^{n+1}) &\geq \sigma M \mathcal{D}(p^n) - \sigma \sum_{i=1}^{M} \mathcal{D}(\widetilde{p}_i^n) = \sigma \sum_{i=1}^{M} \Big(\mathcal{D}(p^n) - \mathcal{D}(\widetilde{p}_i^n)\Big) \\ &\geq \rho\sigma \sum_{i=1}^{M} \Big(\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}_i^n)\Big),\end{aligned}$$

where we denoted $\widehat{p}_i^n = p_{i-1}^n + (\widehat{v}_i^n - \theta_i p^n) = p^n + (\widehat{v}_i^n - \theta_i p^n)$ in the last inequality.
For the sequential case of Algorithm 2 we have similarly

$$p_i^n = p_{i-1}^n + \sigma(\widetilde{v}_i^n - \theta_i p^n) = (1-\sigma)p_{i-1}^n + \sigma\left(p_{i-1}^n + (\widetilde{v}_i^n - \theta_i p^n)\right) = (1-\sigma)p_{i-1}^n + \sigma\widetilde{p}_i^n$$

and thus $\mathcal{D}(p_i^n) \leq (1-\sigma)\mathcal{D}(p_{i-1}^n) + \sigma\mathcal{D}(\widetilde{p}_i^n)$. Rewriting we see that

$$\mathcal{D}(p_{i-1}^n) - \mathcal{D}(p_i^n) \geq \sigma(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widetilde{p}_i^n)) \geq \rho\sigma(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widehat{p}_i^n)), \tag{4.1}$$

where we again used the definition of $\widetilde{v}_i^n$ in the second inequality. A telescope sum over $i = 1, \ldots, M$ then yields

$$\mathcal{D}(p^n) - \mathcal{D}(p^{n+1}) = \sum_{i=1}^{M} (\mathcal{D}(p_{i-1}^n) - \mathcal{D}(p_i^n)) \geq \rho\sigma \sum_{i=1}^{M} (\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widehat{p}_i^n)).$$

■

In particular, Lemma 4.1 shows monotonicity of energies, i.e. $\mathcal{D}(p^n) \geq \mathcal{D}(p^{n+1})$. Because of the coercivity (1.6), the set of iterates $\{p^n : n \in \mathbb{N}_0\} \subseteq V$ is therefore bounded. We thus denote for some

fixed minimizer $\widehat{p} \in K$ of $\mathcal{D}$ the finite radius

$$R_{\widehat{p}} := \sup\{\|p - \widehat{p}\|_V : p \in K, \mathcal{D}(p) \leq \mathcal{D}(p^0)\} < \infty. \tag{4.2}$$

4.2. **Main Results**

Now we are able to state our main convergence result providing a convergence rate of the proposed algorithms to a minimizer of the original (global) problem.

Theorem 4.2. *Let $(p^n)_{n\in\mathbb{N}_0}$ be the iterates from either one of Algorithms 1 and 2 and let $\widehat{p} \in K$ denote a minimizer of $\mathcal{D}$. Algorithms 1 and 2 converge in the sense that $\mathcal{D}(p^n) \to \mathcal{D}(\widehat{p})$. More specifically,*

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \begin{cases} (1 - \frac{\rho\sigma}{2\alpha})^n (\mathcal{D}(p^0) - \mathcal{D}(\widehat{p})) & \text{if } n \leq n_0 \\ \frac{2\Phi^2}{\rho\sigma}\alpha^2 (n - n_0 + 1)^{-1} & \text{if } n \geq n_0, \end{cases}$$

where $\alpha := 1 + M\sigma\sqrt{2 - \rho + 2\sqrt{1-\rho}}$ for Algorithm 2 and $\alpha := 1$ for Algorithm 1, $\Phi := \sqrt{\|B^{-1}\|}\|\Lambda\| C_\theta R_{\widehat{p}}$ with $C_\theta := \left(\sum_{i=1}^M \|\theta_i\|^2\right)^{\frac{1}{2}}$ and $n_0 := \min\{n \in \mathbb{N}_0 : \mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) < \Phi^2\alpha\}$.

Proof. The proof is deferred to Section 4.4. ∎

Theorem 4.2 provides a convergence rate for Algorithms 1 and 2 and allows for a rough comparison. Considering exact minimization, i.e. $\rho = 1$ and the maximally allowed values of σ, i.e. $\sigma = 1$ for Algorithm 2 (sequential) and $\sigma = \frac{1}{M}$ for Algorithm 1 (parallel) we obtain from Theorem 4.2 for these two algorithms the following convergence rate coefficients:

- Algorithm 1: $\frac{2\Phi^2}{\rho\sigma}\alpha^2 = 2\Phi^2 M =: C_1$ and $1 - \frac{\rho\sigma}{2\alpha} = 1 - \frac{1}{2M} =: \widetilde{C}_1$
- Algorithm 2: $\frac{2\Phi^2}{\rho\sigma}\alpha^2 = 2\Phi^2(1+M)^2 =: C_2$ and $1 - \frac{\rho\sigma}{2\alpha} = 1 - \frac{1}{2(M+1)} =: \widetilde{C}_2$.

Hence we obtain $C_1 < C_2$ and $\widetilde{C}_1 < \widetilde{C}_2$, i.e. the estimates of Algorithm 1 are better than those of Algorithm 2 for each of the two convergence regimes. Further, we have that $\Phi^2\alpha = \Phi^2 =: c_1$ for Algorithm 1 and $\Phi^2\alpha = \Phi^2(M+1) =: c_2$ for Algorithm 2, showing that $c_1 < c_2$, which shows that the linear convergence regime for Algorithm 1 is larger than for Algorithm 2. Combining this with the observation that in practice sequential decomposition algorithms generally tend to outperform parallel ones [39], this indicates that the estimate for Algorithm 2 may still have room for improvement.

We recall that Algorithms 1 and 2 compute an approximate solution to (1.4). Thanks to (1.5) we can relate this approximation to the primal variable, which may represent a restored image in imaging applications. In this vein the difference in the predual energy can be related to the L^2-error of the primal variable in the following way:

Proposition 4.3. *Let $\widehat{p} \in V$ be a minimizer of (1.4) and $\widehat{u} \in W$ be the minimizer of (1.1). Then for all $p \in V$ and $u := B^{-1}(-\Lambda p + T^* g)$ we have*

$$\frac{c_B}{2}\|u - \widehat{u}\|_W^2 \leq \mathcal{D}(p) - \mathcal{D}(\widehat{p}).$$

Proof. Due to coercivity of a_B we have for $v \in W$

$$\begin{aligned} c_B\|B^{-1}v\|_W^2 \leq a_B(B^{-1}v, B^{-1}v) &= \left\langle (T^*T + \beta I)B^{-1}v, B^{-1}v\right\rangle_W \\ &= \langle v, B^{-1}v\rangle_W = \|v\|_{B^{-1}}^2. \end{aligned}$$

By expanding the quadratic functional $\mathcal{D}$ at $\widehat{p}$ and using optimality of $\widehat{p}$, i.e. $\langle \mathcal{D}'(\widehat{p}), p-\widehat{p}\rangle \geq 0$, we then see that

$$\begin{aligned}\mathcal{D}(p)-\mathcal{D}(\widehat{p}) &= \langle \mathcal{D}'(\widehat{p}), p-\widehat{p}\rangle_V + \frac{1}{2}\langle \Lambda^* B^{-1}\Lambda(p-\widehat{p}), p-\widehat{p}\rangle_V \\ &\geq \frac{1}{2}\|\Lambda(p-\widehat{p})\|^2_{B^{-1}} \geq \frac{c_B}{2}\|B^{-1}\Lambda(p-\widehat{p})\|^2_W = \frac{c_B}{2}\|u-\widehat{u}\|^2_W,\end{aligned}$$

since due to Proposition 1.1 $\widehat{u}$ is given by $\widehat{u} = B^{-1}(-\Lambda\widehat{p} + T^*g)$. ∎

4.3. **Collection of Useful Results**

Here we collect some statements which we use to prove Theorem 4.2.

Definition 4.4. For $p, q \in V$ we introduce the notation

$$\langle p, q\rangle_* := \langle \Lambda^* B^{-1}\Lambda p, q\rangle_V, \qquad |p|_* := \sqrt{\langle p, p\rangle_*}.$$

Note that we have $|p|^2_* = \|\Lambda p\|^2_{B^{-1}}$ in particular and that $\langle\cdot,\cdot\rangle_*$ and $|\cdot|_*$ are not necessarily positive definite.

Lemma 4.5. *Let $\mathcal{D}' : V \to V^*$ be the Fréchet derivative of $\mathcal{D}$. For any $p, q, r \in V$ we have*

(i) $\mathcal{D}(p) - \mathcal{D}(q) = \langle \mathcal{D}'(q), p-q\rangle_V + \frac{1}{2}|p-q|^2_*$,

(ii) $\langle \mathcal{D}'(p) - \mathcal{D}'(q), r\rangle_V = \langle p-q, r\rangle_*$.

Proof.

(i) We expand the quadratic functional $\mathcal{D}$ at q to obtain

$$\begin{aligned}\mathcal{D}(p) &= \mathcal{D}(q) + \langle \mathcal{D}'(q), p-q\rangle_V + \frac{1}{2}\langle \Lambda^* B^{-1}\Lambda(p-q), p-q\rangle_V \\ &= \mathcal{D}(q) + \langle \mathcal{D}'(q), p-q\rangle_V + \frac{1}{2}|p-q|^2_*.\end{aligned}$$

(ii) We see directly

$$\begin{aligned}\langle \mathcal{D}'(p) - \mathcal{D}'(q), r\rangle_V &= \langle \Lambda^* B^{-1}(\Lambda p - T^* g) - \Lambda^* B^{-1}(\Lambda q - T^* g), r\rangle_V \\ &= \langle \Lambda^* B^{-1}\Lambda(p-q), r\rangle_V = \langle p-q, r\rangle_*.\end{aligned}$$

∎

We note that Lemma 4.5 actually holds true for any quadratic functional.

The following result is inspired by the work [7], while similar results can also be found in earlier works, see for example [15]. It helps us to derive the convergence rate of our presented decomposition algorithms by explicitly approximating the descending energy.

Lemma 4.6. *Let $c > 0$ and $(a_k)_{k\in\mathbb{N}_0} \subseteq \mathbb{R}^+$ be a sequence such that for all $k \in \mathbb{N}_0$:*

$$a_k - a_{k+1} \geq c a_k^2.$$

Then $\lim_{k\to\infty} a_k \to 0$ with rate

$$0 < a_k < \frac{1}{ck + \frac{1}{a_0}} < \frac{1}{ck}$$

for all $k \in \mathbb{N}$.

Proof. Although the proof is standard, we state it for completeness and proceed similar to [7]. Since the iterates a_k, $k \in \mathbb{N}_0$ are monotonically decreasing, we can write for $k \in \mathbb{N}_0$:

$$\frac{1}{a_{k+1}} - \frac{1}{a_k} = \frac{a_k - a_{k+1}}{a_{k+1}a_k} \geq \frac{ca_k}{a_{k+1}} > c$$

and use it to reduce the telescope sum for $k > 0$:

$$\frac{1}{a_k} = \sum_{j=0}^{k-1} \left(\frac{1}{a_{j+1}} - \frac{1}{a_j} \right) + \frac{1}{a_0} > ck + \frac{1}{a_0}.$$

Inverting the inequality yields the statement. ∎

Lemma 4.7. *Let $a, b > 0$, $c, x, y \geq 0$ such that for all $\mu \in (0, 1]$ the inequality*

$$y \leq a\mu + \frac{b}{\mu}x + c\sqrt{x}$$

holds. Then the following split inequality holds:

$$y \leq \begin{cases} (2b + c\frac{\sqrt{b}}{\sqrt{a}})x & \text{if } x > \frac{a}{b}, \\ (2\sqrt{ab} + c)\sqrt{x} & \text{if } x \leq \frac{a}{b}, \end{cases} \tag{4.3}$$

or equivalently

$$x \geq \begin{cases} (2b + c\frac{\sqrt{b}}{\sqrt{a}})^{-1}y & \text{if } y > 2a + c\frac{\sqrt{a}}{\sqrt{b}}, \\ (2\sqrt{ab} + c)^{-2}y^2 & \text{if } y \leq 2a + c\frac{\sqrt{a}}{\sqrt{b}}. \end{cases}$$

Proof. To obtain (4.3) we perform a case analysis on x.
If $x > \frac{a}{b}$ we choose $\mu = 1$ to arrive at

$$y \leq a + bx + c\sqrt{x} < 2bx + c\sqrt{x} \leq \left(2b + c\frac{\sqrt{b}}{\sqrt{a}} \right) x.$$

For $0 < x \leq \frac{a}{b}$ we minimize the expression by choosing $\mu = \frac{\sqrt{b}}{\sqrt{a}}\sqrt{x}$ and get

$$y \leq a\mu + b\frac{\sqrt{a}}{\sqrt{b}}\sqrt{x} + c\sqrt{x} = (2\sqrt{ab} + c)\sqrt{x}.$$

If $x = 0$, we have $y \leq a\mu$ for all $\mu > 0$. That is $y = \inf_{\mu > 0} a\mu = 0$.
These three cases together yield the estimate (4.3).
Noting that the right-hand side of (4.3) is continuous and monotone in x, the case distinction can equivalently be written in terms of y by splitting at $x = \frac{a}{b}$, $y = (2b + c\frac{\sqrt{b}}{\sqrt{a}})\frac{a}{b} = 2a + c\sqrt{ab}$. Seperately solving the inequalities for x thus yields the equivalent representation. ∎

Lemma 4.8. *For all $p \in V$ we have*

$$\sum_{i=1}^{M} |\theta_i p|_*^2 \leq \|B^{-1}\| \|\Lambda\|^2 C_\theta^2 \|p\|_V^2$$

with $C_\theta^2 := \sum_{i=1}^{M} \|\theta_i\|^2$, where θ_i are defined as in Section 2.1.

Proof. Application of the Cauchy–Schwarz inequality yields

$$\sum_{i=1}^{M} |\theta_i p|_*^2 = \sum_{i=1}^{M} \langle \Lambda\theta_i p, B^{-1}\Lambda\theta_i p\rangle_W$$
$$\leq \sum_{i=1}^{M} \|B^{-1}\|\|\Lambda\|^2\|\theta_i\|^2\|p\|_V^2 = \|B^{-1}\|\|\Lambda\|^2 \left(\sum_{i=1}^{M}\|\theta_i\|^2\right)\|p\|_V^2. \qquad \blacksquare$$

In the following we employ ideas from alternating minimization [7] to achieve a convergence rate estimate.

Lemma 4.9. *Let $(p_i^n)_{n\in\mathbb{N}_0}$ be generated by Algorithm 2, then we have*

$$\frac{1}{2}|p_{i-1}^n - p_i^n|_*^2 \leq \frac{\sigma}{\rho}\big(2-\rho+2\sqrt{1-\rho}\big)\big(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(p_i^n)\big).$$

Proof. Let $\omega > 0$ to be chosen later and denote $\widetilde{p}_i^n := p_{i-1}^n + (\widetilde{v}_i^n - \theta_i p^n)$, where $\widetilde{v}_i^n$ is defined by Algorithm 2. Then we have

$$\begin{aligned}\frac{1}{2\sigma^2}|p_{i-1}^n - p_i^n|_*^2 &= \frac{1}{2\sigma^2}|\sigma(\widetilde{v}_i^n - \theta_i p^n)|_*^2\\ &= \frac{1}{2}|p_{i-1}^n - \widetilde{p}_i^n|_*^2\\ &\leq \frac{1}{2}\Big((1+\omega)|p_{i-1}^n - \widehat{p}_i^n|_*^2 + (1+\omega^{-1})|\widetilde{p}_i^n - \widehat{p}_i^n|_*^2\Big)\\ &\leq (1+\omega)(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widehat{p}_i^n)) + (1+\omega^{-1})(\mathcal{D}(\widetilde{p}_i^n) - \mathcal{D}(\widehat{p}_i^n)),\end{aligned}$$

where $\widehat{p}_i^n$ is defined as in Lemma 4.1 and the last inequality is obtained by using Lemma 4.5 (i) and the optimality of $\widehat{p}_i^n$. By utilizing (3.1) and (4.1) we further estimate

$$\begin{aligned}\frac{1}{2\sigma^2}|p_{i-1}^n - p_i^n|_*^2 &\leq \frac{1+\omega}{\rho}(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widetilde{p}_i^n)) + \frac{(1+\omega^{-1})(1-\rho)}{\rho}(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widetilde{p}_i^n)) \qquad \text{(due to (3.1))}\\ &= \frac{1}{\rho}\big(1+\omega+(1+\omega^{-1})(1-\rho)\big)(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(\widetilde{p}_i^n))\\ &\leq \frac{1}{\sigma\rho}\big(1+\omega+(1+\omega^{-1})(1-\rho)\big)(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(p_i^n)). \qquad \text{(using (4.1))}\end{aligned}$$

Choosing $\omega := \sqrt{1-\rho}$ so as to minimize the expression we arrive at

$$\frac{1}{2\sigma^2}|p_{i-1}^n - p_i^n|_*^2 \leq \frac{1}{\sigma\rho}\big(2-\rho+2\sqrt{1-\rho}\big)\big(\mathcal{D}(p_{i-1}^n) - \mathcal{D}(p_i^n)\big). \qquad \blacksquare$$

Proposition 4.10. *Let $(p^n)_{n\in\mathbb{N}_0}$ be the iterates from either one of Algorithms 1 and 2 and let $\widehat{p}\in K$ denote a minimizer of $\mathcal{D}$.*

Then $\mathcal{D}(p^n) \to \mathcal{D}(\widehat{p})$ as $n\to\infty$ owing to

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \begin{cases} \frac{2}{\rho\sigma}\alpha\big(\mathcal{D}(p^n) - \mathcal{D}(p^{n+1})\big) & \text{if } \mathcal{D}(p^n) - \mathcal{D}(p^{n+1}) > \frac{1}{2}\sigma\rho\Phi^2,\\ \sqrt{\frac{2}{\rho\sigma}}\Phi\alpha\sqrt{\mathcal{D}(p^n) - \mathcal{D}(p^{n+1})} & \text{otherwise,}\end{cases}$$

where $\alpha := 1 + M\sigma\sqrt{2-\rho+2\sqrt{1-\rho}}$ for Algorithm 2 and $\alpha := 1$ for Algorithm 1, and $\Phi := \sqrt{\|B^{-1}\|}\|\Lambda\|C_\theta R_{\widehat{p}}$.

Proof. Using convexity we expand

$$\begin{aligned}\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) &\leq \langle \mathcal{D}'(p^n), p^n - \widehat{p}\rangle_V \\ &= \sum_{i=1}^{M} \langle \mathcal{D}'(p^n), \theta_i(p^n - \widehat{p})\rangle_V \\ &= \sum_{i=1}^{M} \left(\langle \mathcal{D}'(p^n_{i-1}), \theta_i(p^n - \widehat{p})\rangle_V + \sum_{j=1}^{i-1} \langle \mathcal{D}'(p^n_{j-1}) - \mathcal{D}'(p^n_j), \theta_i(p^n - \widehat{p})\rangle_V \right). \end{aligned} \tag{4.4}$$

For the latter equality we used a telescopic sum and that $p^n = p^n_0$. Let $\Phi_n := (\sum_{i=1}^M |\theta_i(p^n - \widehat{p})|_*^2)^{\frac{1}{2}}$, $\widehat{v}^n_i \in \arg\min_{v_i \in \theta_i K} \mathcal{D}(p^n_{i-1} + (v_i - \theta_i p^n))$ and $\widehat{p}^n_i := p^n_{i-1} + (\widehat{v}^n_i - \theta_i p^n)$. We now estimate the first summand in the expansion above:

$$\begin{aligned}&\sum_{i=1}^{M} \langle \mathcal{D}'(p^n_{i-1}), \theta_i(p^n - \widehat{p})\rangle_V \\ &\quad= \frac{1}{\mu} \sum_{i=1}^{M} \langle \mathcal{D}'(p^n_{i-1}), \mu\theta_i(p^n - \widehat{p})\rangle_V \\ &\quad= \frac{\mu}{2} \sum_{i=1}^{M} |\theta_i(p^n - \widehat{p})|_*^2 + \frac{1}{\mu} \sum_{i=1}^{M} \left(\mathcal{D}(p^n_{i-1}) - \mathcal{D}(p^n_{i-1} - \mu\theta_i(p^n - \widehat{p})) \right) && \text{(Lemma 4.5 (i))} \\ &\quad= \frac{\Phi_n^2 \mu}{2} + \frac{1}{\mu} \sum_{i=1}^{M} \Big(\mathcal{D}(p^n_{i-1}) - \mathcal{D}\big(p^n_{i-1} + ((1-\mu)\theta_i p^n + \mu\theta_i\widehat{p} - \theta_i p^n)\big) \Big) \\ &\quad\leq \frac{\Phi_n^2 \mu}{2} + \frac{1}{\mu} \sum_{i=1}^{M} (\mathcal{D}(p^n_{i-1}) - \mathcal{D}(\widehat{p}^n_i)) && \text{(optimality)} \\ &\quad\leq \frac{\Phi_n^2 \mu}{2} + \frac{1}{\mu\rho\sigma} (\mathcal{D}(p^n) - \mathcal{D}(p^{n+1})), && \text{(Lemma 4.1)} \end{aligned}$$

where optimality was used by realizing that $(1-\mu)\theta_i p^n + \mu\theta_i\widehat{p} \in \theta_i K$. For the second summand and for Algorithm 2 we see

$$\begin{aligned}&\sum_{i=1}^{M} \sum_{j=1}^{i-1} \langle \mathcal{D}'(p^n_{j-1}) - \mathcal{D}'(p^n_j), \theta_i(p^n - \widehat{p})\rangle_V \\ &\quad= \sum_{i=1}^{M} \sum_{j=1}^{i-1} \langle p^n_{j-1} - p^n_j, \theta_i(p^n - \widehat{p})\rangle_* && \text{(Lemma 4.5 (ii))} \\ &\quad\leq \sum_{i=1}^{M} \sum_{j=1}^{i-1} |p^n_{j-1} - p^n_j|_* |\theta_i(p^n - \widehat{p})|_* \\ &\quad\leq M \left(\sum_{j=1}^{M} |p^n_{j-1} - p^n_j|_*^2 \right)^{\frac{1}{2}} \left(\sum_{i=1}^{M} |\theta_i(p^n - \widehat{p})|_*^2 \right)^{\frac{1}{2}} \\ &\quad\leq M\Phi_n \left(\sum_{j=1}^{M} |p^n_{j-1} - p^n_j|_*^2 \right)^{\frac{1}{2}}. \end{aligned}$$

Applying Lemma 4.9 completes the estimate of the second summand, yielding

$$\sum_{i=1}^{M}\sum_{j=1}^{i-1}\langle \mathcal{D}'(p_{i-1}^n)-\mathcal{D}'(p_i^n),\theta_j(p^n-\widehat{p})\rangle_V \le M\Phi_n\sqrt{\frac{2\sigma}{\rho}(2-\rho+2\sqrt{1-\rho})}(\mathcal{D}(p^n)-\mathcal{D}(p^{n+1}))^{\frac{1}{2}}.$$

Combining both estimates and roughly bounding $\Phi_n \le \Phi$ due to Lemma 4.8 we have

$$\mathcal{D}(p^n)-\mathcal{D}(\widehat{p}) \le \frac{\Phi^2\mu}{2}+\frac{1}{\mu\rho\sigma}(\mathcal{D}(p^n)-\mathcal{D}(p^{n+1}))+M\Phi\sqrt{\frac{2\sigma}{\rho}(2-\rho+2\sqrt{1-\rho})}(\mathcal{D}(p^n)-\mathcal{D}(p^{n+1}))^{\frac{1}{2}}.$$

Invoking Lemma 4.7 with $a=\frac{\Phi^2}{2}$, $b=\frac{1}{\rho\sigma}$ and $c=M\Phi\sqrt{\frac{2\sigma}{\rho}(2-\rho+2\sqrt{1-\rho})}$ yields the split bound with the following coefficients:

$$\begin{aligned}
2b+c\sqrt{\frac{b}{a}} &= \frac{2}{\rho\sigma}+M\Phi\sqrt{\frac{2\sigma}{\rho}(2-\rho+2\sqrt{1-\rho})}\sqrt{\frac{2}{\sigma\rho\Phi^2}}\\
&= \frac{2}{\rho\sigma}(1+M\sigma\sqrt{2-\rho+2\sqrt{1-\rho}}),\\
2\sqrt{ab}+c &= 2\sqrt{\frac{\Phi^2}{2\rho\sigma}}+M\Phi\sqrt{\frac{2\sigma}{\rho}(2-\rho+2\sqrt{1-\rho})}\\
&= \sqrt{\frac{2}{\rho\sigma}}\Phi\left(1+M\sigma\sqrt{2-\rho+2\sqrt{1-\rho}}\right),
\end{aligned}$$

which concludes the proof for Algorithm 2.
For Algorithm 1, examining the proof above, we notice that for the parallel version we have $p_i^n = p_{i-1}^n = p^n$ for $i=1,\dots,M-1$ and thus the second summand in (4.4) vanishes completely. This allows us to invoke Lemma 4.7 with $c=0$ and leads to the desired statement. ∎

Now we are able to prove our main result.

4.4. **Proof of Theorem 4.2**

We first observe that since $(\mathcal{D}(p^n))_{n\in\mathbb{N}_0}$ is monotonically decreasing, n_0 is well-defined and we have $\mathcal{D}(p^n)-\mathcal{D}(\widehat{p}) \ge \Phi^2\alpha$ for all $n\in\mathbb{N}_0$, $n<n_0$ and likewise $\mathcal{D}(p^n)-\mathcal{D}(\widehat{p}) < \Phi^2\alpha$ for all $n\in\mathbb{N}_0$, $n\ge n_0$.

We now make use of Proposition 4.10. The equivalence in Lemma 4.7 with $x := \mathcal{D}(p^{n-1})-\mathcal{D}(p^n)$, $y := \mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p})$, $a := \frac{\Phi\alpha}{2}$, $b := \frac{\alpha}{\rho\sigma}$ and $c := 0$ then yields

$$\mathcal{D}(p^{n-1})-\mathcal{D}(p^n) \ge \begin{cases} \frac{\rho\sigma}{2\alpha}(\mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p})) & \text{if } \mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p}) > \Phi^2\alpha \\ \frac{\rho\sigma}{2\Phi^2\alpha^2}(\mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p}))^2 & \text{if } \mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p}) \le \Phi^2\alpha, \end{cases}$$

which can also be rewritten as

$$\mathcal{D}(p^{n-1})-\mathcal{D}(p^n) \ge \begin{cases} \frac{\rho\sigma}{2\alpha}(\mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p})) & \text{if } n-1<n_0 \\ \frac{\rho\sigma}{2\Phi^2\alpha^2}(\mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p}))^2 & \text{if } n-1\ge n_0. \end{cases}$$

In the former case we invert the inequality and add $\mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p})$ to arrive at

$$\mathcal{D}(p^n)-\mathcal{D}(\widehat{p}) \le \left(1-\frac{\rho\sigma}{2\alpha}\right)(\mathcal{D}(p^{n-1})-\mathcal{D}(\widehat{p})),$$

which recursively yields the required statement for all $n\le n_0$. In the latter case we may assume without loss of generality that $n_0=0$ since we can shift the sequence if necessary. Thus for all $n\in\mathbb{N}_0$:

$$\mathcal{D}(p^n)-\mathcal{D}(p^{n+1}) \ge \frac{\rho\sigma}{2\Phi^2\alpha^2}(\mathcal{D}(p^n)-\mathcal{D}(\widehat{p}))^2.$$

Invoking Lemma 4.6 with constant $c := \frac{\rho\sigma}{2\Phi^2\alpha^2}$ we obtain

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \frac{1}{cn + \frac{1}{\mathcal{D}(p^0)-\mathcal{D}(\widehat{p})}} \leq \frac{1}{cn + \frac{1}{\Phi^2\alpha}} \leq \frac{1}{cn + \frac{\rho\sigma}{2\Phi^2\alpha^2}} = \frac{1}{c(n+1)}$$

since $0 \leq \sigma, \rho \leq 1$ and $\alpha \geq 1$, thereby showing the second inequality, which completes the proof.

5. Comparison

We conclude that in special cases the results obtained here are either in agreement with or may improve upon other known estimates. In particular, in the following we compare our findings with the ones in [41] and [10].

5.1. Gradient Method Framework [41]

In the special case of parallel decomposition, i.e. $\alpha = 1$ (see Theorem 4.2), and exact local solutions, i.e. $\rho = 1$, the framework of [41] is applicable to our model and their estimate [41, Algorithm 4.1] reproduces ours. We show this by specializing and transforming their estimate.

Using notation from [41] we employ [41, Algorithm 4.1] by setting $E(u) = F(u) + G(u) := \mathcal{D}(u) + \chi_K(u)$, where $\chi_K(u) = 0$ if $u \in K$ and ∞ otherwise. The space decomposition is specified by the images of θ_k, $k = 1, \ldots, M$, i.e. $V_k := \operatorname{im}\theta_k \subseteq V$ with $R_k^* : V_k \to V$ then being the inclusion map. We chose to use exact local solvers, i.e. $\rho = 1$ in our notation, since it is not obvious to us how our notion of approximate minimization maps to theirs. In particular, d_k and G_k are chosen as in [41, (4.3)] and $\omega := \omega_0 := 1$. We now verify [41, Assumptions 4.1 to 4.3] in order to apply [41, Theorem 4.7]. [41, Assumption 4.1] is fulfilled due to Lemmas 4.5 and 4.8 with $C_{0,K} := C_\theta\|\Lambda\|\sqrt{\|B^{-1}\|}$ and $q := 2$. We fulfill [41, Assumption 4.2] by choosing $\tau_0 := \frac{1}{N}$ (their τ corresponds to our σ). [41, Assumption 4.3] is trivialized in the case of exact local solvers. Applying [41, Theorem 4.7] with $C_{q,\tau} = 2$ and $\kappa = \frac{1}{\tau}C_\theta^2\|\Lambda\|^2\|B^{-1}\|$ yields

$$\mathcal{D}(p^1) - \mathcal{D}(\widehat{p}) \leq \left(1 - \sigma\left(1 - \frac{1}{2}\right)\right)(\mathcal{D}(p^0) - \mathcal{D}(\widehat{p})) = \left(1 - \frac{\sigma}{2}\right)(\mathcal{D}(p^0) - \mathcal{D}(\widehat{p})) \tag{5.1}$$

if $\mathcal{D}(p^0) - \mathcal{D}(\widehat{p}) \geq \tau R_{\widehat{p}}^2\kappa = \Phi^2$ and

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \frac{C_{q,r}R_{\widehat{p}}^2\kappa}{(n+1)^{q-1}} = \frac{2\Phi^2}{\sigma}(n+1)^{-1} \tag{5.2}$$

otherwise. Applying estimate (5.1) recursively and shifting the sequence by n_0 for the estimate (5.2) finally yields the formulation

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \begin{cases} (1 - \frac{\sigma}{2})^n(\mathcal{D}(p^0) - \mathcal{D}(\widehat{p})) & \text{if } n \leq n_0 \\ \frac{2\Phi^2}{\sigma}(n - n_0 + 1)^{-1} & \text{if } n \geq n_0, \end{cases}$$

which is in agreement with Theorem 4.2.

5.2. Decomposition of the Rudin–Osher–Fatemi Model [10]

In order to compare with the convergence rate in [10], we specialize our model to their setting by chosing $V = H_0^{\mathrm{div}}(\Omega)$, $\Lambda = \operatorname{div} : V \to L^2(\Omega)$, $T = I : L^2(\Omega) \to L^2(\Omega)$, $\beta = 0$ (thus $B = I$) and $\rho = 1$. Next we introduce some notation from [10], namely $C_0, \delta > 0$ such that $\|\nabla\widetilde{\theta}_i\|_{L^\infty} \leq \frac{C_0}{\delta}$ for $i = 1, \ldots, M$, cf. [10, (2.10)], $\zeta^0 := 2(\mathcal{D}(p^0) - \mathcal{D}(\widehat{p}))$ (our $\mathcal{D}$ has an additional factor of $\frac{1}{2}$) and

$N_0 := \max_{x\in\Omega} |\{i \in \{1,\dots,M\} : x \in \Omega_i\}|$. Then [10, Theorem 3.1] and [10, Theorem 3.6] provide the following estimate:

$$\frac{1}{2}\|u^n - \widehat{u}\|_{L^2}^2 \le \mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \le Cn^{-1} \tag{5.3}$$

where $u^n := -\operatorname{div} p^n + g$, $\widehat{u} := -\operatorname{div}\widehat{p} + g$ and

$$C := \frac{1}{2}\zeta^0\left(\frac{2}{\sigma}(2M+1)^2 + 8\sqrt{2}C_0\lambda|\Omega|^{\frac{1}{2}}(\zeta^0)^{-\frac{1}{2}}\frac{M\sqrt{N_0}}{\delta\sqrt{\sigma}} + \sqrt{2} - 1\right)^2.$$

Note that we used our notation for M and σ.

In order to compare favorably in this setting, we slightly refine the estimate $\Phi_n \le \Phi$ from the proof of Proposition 4.10. First, we quantify an estimate from the proof of Lemma 2.2. For all $p \in V$ we have

$$\begin{aligned}
\sum_{i=1}^M \|\operatorname{div}\theta_i p\|_{L^2}^2 &\le \sum_{i=1}^M \|\nabla\widetilde{\theta}_i \cdot p + \widetilde{\theta}_i \operatorname{div} p\|_{L^2}^2 \\
&\le \sum_{i=1}^M \Big((1+\omega)\|\nabla\widetilde{\theta}_i \cdot p\|_{L^2}^2 + (1+\omega^{-1})\|\widetilde{\theta}_i \operatorname{div} p\|_{L^2}^2 \\
&= (1+\omega)\int_\Omega \sum_{i=1}^M |\nabla\widetilde{\theta}_i \cdot p|^2 \,\mathrm{d}x + (1+\omega^{-1})\int_\Omega \sum_{i=1}^M |\widetilde{\theta}_i \operatorname{div} p|^2 \,\mathrm{d}x \\
&\le (1+\omega)\int_\Omega \left(\sum_{i=1}^M |\nabla\widetilde{\theta}_i|^2\right)|p|^2 \,\mathrm{d}x + (1+\omega^{-1})\int_\Omega \left(\sum_{i=1}^M |\widetilde{\theta}_i|^2\right)|\operatorname{div} p|^2 \,\mathrm{d}x \\
&\le (1+\omega)N_0 \max_i\{\|\nabla\widetilde{\theta}_i\|_{L^\infty}^2\}\|p\|_{L^2}^2 + (1+\omega^{-1})\|\operatorname{div} p\|_{L^2}^2 \\
&\le (1+\omega)N_0\frac{C_0^2}{\delta^2}\|p\|_{L^2}^2 + (1+\omega^{-1})\|\operatorname{div} p\|_{L^2}^2,
\end{aligned}$$

for any $\omega > 0$. The pointwise box-constraints $|p| \le \lambda$ imply $\|p^n - \widehat{p}\|_{L^2}^2 = \int_\Omega |p^n - \widehat{p}|^2 \,\mathrm{d}x \le (2\lambda)^2|\Omega|$. Combining this allows us to estimate

$$\begin{aligned}
\Phi_n^2 := \sum_{i=1}^M |\theta_i(p^n - \widehat{p})|_*^2 &= \sum_{i=1}^M \|\operatorname{div}\theta_i(p^n - \widehat{p})\|_{L^2}^2 \\
&\le (1+\omega)N_0\frac{C_0^2}{\delta^2}\|p^n - \widehat{p}\|_{L^2}^2 + (1+\omega^{-1})\|\operatorname{div}(p^n - \widehat{p})\|_{L^2}^2 \\
&\le (1+\omega)\cdot 4\lambda^2|\Omega|N_0\frac{C_0^2}{\delta^2} + (1+\omega^{-1})\zeta^0 \\
&= \left(2\lambda|\Omega|^{\frac{1}{2}}N_0^{\frac{1}{2}}\frac{C_0}{\delta} + (\zeta^0)^{\frac{1}{2}}\right)^2 =: \widetilde{\Phi}^2
\end{aligned}$$

by optimally choosing $\omega := (4\lambda^2|\Omega|N_0\frac{C_0^2}{\delta^2})^{-\frac{1}{2}}(\zeta^0)^{\frac{1}{2}}$. We therefore conclude that in this specific setting Theorem 4.2 holds true with Φ replaced by $\widetilde{\Phi}$. Their and our estimate thus amount to

$$\begin{aligned}
&\frac{1}{2}\|u^n - \widehat{u}\|_{L^2}^2 \le Cn^{-1}, \\
&\frac{1}{2}\|u^n - \widehat{u}\|_{L^2}^2 \le \frac{2\widetilde{\Phi}^2}{\sigma}\alpha^2(n - n_0 + 1)^{-1},
\end{aligned}$$

where for the lower estimate α and n_0 are defined as in Theorem 4.2 and $n \geq n_0$. Noting that $\rho = 1$ we rewrite the involved constants as

$$\begin{aligned} C &= \left(\left(\sqrt{2}\frac{(2M+1)^2}{\sigma} + 1 - \frac{\sqrt{2}}{2} \right) \sqrt{\zeta^0} + 8\frac{M}{\sqrt{\sigma}} \lambda |\Omega|^{\frac{1}{2}} \sqrt{N_0} \frac{C_0}{\delta} \right)^2, \\ \frac{2\widetilde{\Phi}^2 \alpha^2}{\sigma} &\leq \frac{2(1+\sigma M)^2}{\sigma} \left(2\lambda |\Omega|^{\frac{1}{2}} \sqrt{N_0} \frac{C_0}{\delta} + \sqrt{\zeta^0} \right)^2 \\ &= \left(\sqrt{2}\frac{1+\sigma M}{\sqrt{\sigma}} \sqrt{\zeta^0} + 2\sqrt{2}\frac{1+\sigma M}{\sqrt{\sigma}} \lambda |\Omega|^{\frac{1}{2}} \sqrt{N_0} \frac{C_0}{\delta} \right)^2. \end{aligned}$$

We see that $\frac{2\tilde{\Phi}^2 \alpha^2}{\sigma} \leq C$ by comparing the relevant terms before $\sqrt{\zeta^0}$ and $\lambda |\Omega|^{\frac{1}{2}} \sqrt{N_0} \frac{C_0}{\delta}$ under the square separately using $0 < \sigma \leq 1$ and $M \geq 1$:

$$\begin{aligned} \sqrt{2}\frac{1+\sigma M}{\sqrt{\sigma}} &\leq \frac{\sqrt{2}}{\sqrt{\sigma}}(1+M) < \sqrt{2}\frac{(2M+1)^2}{\sigma} \leq \sqrt{2}\frac{(2M+1)^2}{\sigma} + 1 - \frac{\sqrt{2}}{2} \\ 2\sqrt{2}\frac{1+\sigma M}{\sqrt{\sigma}} &\leq 3\frac{1+M}{\sqrt{\sigma}} < 4\frac{2M}{\sqrt{\sigma}} = 8\frac{M}{\sqrt{\sigma}}. \end{aligned}$$

Consequently, Theorem 4.2 provides a strictly better estimate than [10, Theorems 3.1, 3.6] both for sufficiently large $n \in \mathbb{N}$ and for all $n \in \mathbb{N}$ whenever $n_0 = 0$ (i.e. the initial guess is close enough). While we expect Theorem 4.2 to still be better than [10, Theorems 3.1, 3.6] for $n_0 > 0$, a complete comparison in that case seems to be more involved and remains to be done.

6. **Surrogate Technique**

In this section we present and analyze the convergence of Algorithms 1 and 2 when the respective subproblems are solved inexactly by so-called surrogate iterations. A surrogate iteration substitutes minimization of one functional with minimization of different, simpler functionals at the cost of an additional iterative process. For example, one can substitute the minimization problem $\inf_{p \in K} \frac{1}{2}\|\Lambda p - T^* g\|^2_{B^{-1}}$, described in Proposition 1.1, by the iteration

$$\inf_{p^{n+1} \in K} \frac{1}{2}\|\Lambda p^{n+1} - f^n\|^2_W, \qquad f^n = \Lambda p^n - \frac{1}{\tau} B^{-1}(\Lambda p^n - T^* g),$$

producing iterates $(p^n)_{n \in \mathbb{N}}$ for some initialization $p^0 \in V$ that converge to the same minimizer, provided $\tau \in (\|B^{-1}\|, \infty)$ [12, 13]. This surrogate technique is also known as forward-backward splitting [11], where the underlying idea is formulated in a more general setting. Though its properties have been studied extensively in e.g. [38], we will analyze it as a nested subalgorithm of our decomposition scheme for approximate minimization following the notion from Definition 3.1. Actually an approximate solution is obtained, if only a finite number of surrogate iterations $N_{\text{sur}} \in \mathbb{N}$, see Algorithm 3, are performed. The main motivation for the surrogate technique in our case is to rid the local problems from the dependence on the potentially non-local operator B^{-1}.

To that end for $n \in \mathbb{N}_0$ we introduce an auxiliary functional $\mathcal{D}_i^{s,n}$ defined as

$$\mathcal{D}_i^{s,n}(v_i, w_i) := \mathcal{D}(p_{i-1}^n + (v_i - \theta_i p^n)) + \frac{1}{2}\|\Lambda(v_i - w_i)\|^2_{\tau I - B^{-1}}$$

with $\tau > \|B^{-1}\|$ for $v_i, w_i \in \theta_i K$ and $i = 1, \ldots, M$, whereby $\|u\|^2_{\tau I - B^{-1}} := \langle u, (\tau I - B^{-1})u \rangle_W$ for $u \in W$.

Algorithm 3 Surrogate approximation

Require: $N_{\mathrm{sur}} \in \mathbb{N}$, $n \in \mathbb{N}_0$, $i \in \{1, \dots, M\}$, $p^n \in K$, $p_{i-1}^n \in K$
Ensure: $\widetilde{v}_i^n \in \theta_i K$
1: $v_i^{n,0} = \theta_i p_{i-1}^n$
2: **for** $\ell = 0, 1, \dots, N_{\mathrm{sur}} - 1$ **do**
3: $\quad v_i^{n,\ell+1} \in \arg\min_{v_i \in \theta_i K} \mathcal{D}_i^{s,n}(v_i, v_i^{n,\ell})$
4: **end for**
5: $\widetilde{v}_i^n = v_i^{n,N_{\mathrm{sur}}}$

We note that the subproblems in Algorithm 3 can be written as

$$\begin{aligned} &\inf_{v_i \in \theta_i K} \mathcal{D}_i^{s,n}(v_i, v_i^{n,\ell}) \\ \iff &\inf_{v_i \in \theta_i K} \frac{1}{2}\|\Lambda(p_{i-1}^n + (v_i - \theta_i p^n)) - T^* g\|_{B^{-1}}^2 + \frac{1}{2}\|\Lambda(v_i - v_i^{n,\ell})\|_{\tau I - B^{-1}}^2 \\ \iff &\inf_{v_i \in \theta_i K} \frac{1}{2}\|\Lambda v_i - f_i^n\|_W^2, \end{aligned} \tag{6.1}$$

where $f_i^n = \Lambda v_i^{n,\ell} - \frac{1}{\tau} B^{-1}(\Lambda(p_{i-1}^n + (v_i^{n,\ell} - \theta_i p^n)) - T^* g)$. The dependence on the operator B^{-1} has thereby been moved into the preparation of fixed data f_i^n for every subproblem, while the subproblem itself for fixed f_i^n is independent of B^{-1}.

Algorithm 3 generates ρ-approximate minimizers $\widetilde{v}_i^n$ of $\arg\min_{v_i \in \theta_i K} \mathcal{D}(p_{i-1}^n + (v_i - \theta_i p^n))$ for a suitable $\rho \in (0, 1]$ as requested in Algorithms 1 and 2. In fact, due to the convergence of the surrogate iterations, for any $\rho \in (0, 1]$ there exists an $N_{\mathrm{sur}} \in \mathbb{N}$ such that $\widetilde{v}_i^n$ is indeed a respective ρ-approximation. Hence Algorithm 3 can be used to generate an approximate solution $\widetilde{v}_i^n$ in Algorithms 1 and 2.

Following ideas from [38, Proposition 2.2] we show below, that the surrogate approximation converges linearly and any fixed number of surrogate iterations N_{sur} is enough to receive the convergence rate from Theorem 4.2 for the resulting combined algorithm.

Lemma 6.1. *Using notation and assumptions from Algorithm 3 the functional $\mathcal{D}_i^n : V_i \to \mathbb{R}$,*

$$\mathcal{D}_i^n(v) := \mathcal{D}(p_{i-1}^n - \theta_i p^n + v),$$

has quadratic growth in the sense that

$$\mathcal{D}_i^n(v) - \mathcal{D}_i^n(\widehat{v}) \geq \frac{1}{2\|\tau I - B^{-1}\|\|B\|}\|\Lambda(v - \widehat{v})\|_{\tau I - B^{-1}}^2$$

for any minimizer $\widehat{v} \in \theta_i K$ of $\mathcal{D}_i^n$.

Proof. Using Lemma 4.5 and optimality of $\widehat{v} \in \theta_i K$ we see that

$$\begin{aligned} \mathcal{D}_i^n(v) - \mathcal{D}_i^n(\widehat{v}) &= \langle \mathcal{D}'(p_{i-1}^n + (\widehat{v} - \theta_i p^n)), v - \widehat{v}\rangle + \frac{1}{2}\|v - \widehat{v}\|_*^2 \\ &\geq \frac{1}{2}\|\Lambda(v - \widehat{v})\|_{B^{-1}}^2. \end{aligned}$$

Further noting that $\tau I - B^{-1}$ is positive definite, since $\tau > \|B^{-1}\|$,

$$\|\Lambda(v - \widehat{v})\|_{\tau I - B^{-1}}^2 \leq \|\tau I - B^{-1}\|\|\Lambda(v - \widehat{v})\|_W^2 \leq \|\tau I - B^{-1}\|\|B\|\|\Lambda(v - \widehat{v})\|_{B^{-1}}^2.$$

Combining both inequalities yields the statement. ∎

Proposition 6.2. *Using notation and assumptions from Algorithm 3 and Lemma 6.1 the surrogate iterates $(v_i^{n,\ell})_\ell$ satisfy*

$$\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(v_i^{n,\ell+1}) \geq \eta(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n))$$

for all $\ell \in \mathbb{N}$ and for any minimizer $\widehat{v}_i^n \in \theta_i K$ of $\mathcal{D}_i^n$, $i \in \{1,\dots,n\}$, $n \in \mathbb{N}_0$, while $\eta \in (0,1)$ is given by

$$\eta = \begin{cases} \frac{1}{4\|\tau I - B^{-1}\|\|B\|} & \text{if } \|\tau I - B^{-1}\|\|B\| \geq \frac{1}{2} \\ 1 - \|\tau I - B^{-1}\|\|B\| & \text{otherwise.} \end{cases}$$

Proof. Since $\mathcal{D}_i^{s,n}(v,w) = \mathcal{D}_i^n(v) + \frac{1}{2}\|\Lambda(w-v)\|^2_{\tau I - B^{-1}}$ we have

$$\begin{aligned}
&\mathcal{D}_i^n(v_i^{n,\ell+1}) + \frac{1}{2}\|\Lambda(v_i^{n,\ell} - v_i^{n,\ell+1})\|^2_{\tau I - B^{-1}} \\
&\quad = \mathcal{D}_i^{s,n}(v_i^{n,\ell+1}, v_i^{n,\ell}) \\
&\quad = \min_{v_i \in \theta_i K} \mathcal{D}_i^n(v_i) + \frac{1}{2}\|\Lambda(v_i^{n,\ell} - v_i)\|^2_{\tau I - B^{-1}} \\
&\quad \leq \min_{\mu \in [0,1]} \mathcal{D}_i^n((1-\mu)v_i^{n,\ell} + \mu\widehat{v}_i^n) + \frac{\mu^2}{2}\|\Lambda(v_i^{n,\ell} - \widehat{v}_i^n)\|^2_{\tau I - B^{-1}} \\
&\quad \leq \min_{\mu \in [0,1]} (1-\mu)\mathcal{D}_i^n(v_i^{n,\ell}) + \mu\mathcal{D}_i^n(\widehat{v}_i^n) + \frac{\mu^2}{2}\|\Lambda(v_i^{n,\ell} - \widehat{v}_i^n)\|^2_{\tau I - B^{-1}},
\end{aligned}$$

where we searched for the minimum along the line $v_i = (1-\mu)v_i^{n,\ell} + \mu\widehat{v}_i^n \in \theta_i K$, $\mu \in [0,1]$, and used convexity afterwards. After reordering we use the quadratic growth property from Lemma 6.1 to see that

$$\begin{aligned}
&\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(v_i^{n,\ell+1}) - \frac{1}{2}\|\Lambda(v_i^{n,\ell} - v_i^{n,\ell+1})\|^2_{\tau I - B^{-1}} \\
&\quad \geq \max_{\mu \in [0,1]} \mu(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n)) - \frac{\mu^2}{2}\|\Lambda(v_i^{n,\ell} - \widehat{v}_i^n)\|^2_{\tau I - B^{-1}} \\
&\quad \geq \max_{\mu \in [0,1]} (\mu - \mu^2\|\tau I - B^{-1}\|\|B\|)(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n)).
\end{aligned}$$

Discarding the last term on the left-hand side and evaluating the maximum optimally at $\mu = \min\{1, \frac{1}{2\|\tau I - B^{-1}\|\|B\|}\} \in (0,1]$ yields

$$\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(v_i^{n,\ell+1}) \geq \eta(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n))$$

where $\eta \in (0,1)$ is given by

$$\eta = \begin{cases} \frac{1}{4\|\tau I - B^{-1}\|\|B\|} & \text{if } \|\tau I - B^{-1}\|\|B\| \geq \frac{1}{2} \\ 1 - \|\tau I - B^{-1}\|\|B\| & \text{otherwise.} \end{cases}$$

■

Proposition 6.2 is sharp in the sense that for trivial $B^{-1} = I$ and minimizing $1 < \tau \to 1$, we recover the optimal factor $\eta \to 1$.

Lemma 6.3. *The surrogate iterates $(v_i^{n,\ell})_\ell$ from Algorithm 3 yield approximate solutions to the subproblems in the sense that*

$$\mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(v_i^{n,\ell}) \geq (1 - (1-\eta)^\ell)(\mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(\widehat{v}_i^n))$$

for any minimizer $\widehat{v}_i^n \in \theta_i K$ of $\mathcal{D}_i^n$, $i \in \{1\dots,M\}$, $n \in \mathbb{N}_0$ and $\eta \in (0,1)$ defined as in Proposition 6.2.

Proof. Elementary calculation using Proposition 6.2 yields a linear energy decrease

$$\begin{aligned}\mathcal{D}_i^n(v_i^{n,\ell+1}) - \mathcal{D}_i^n(\widehat{v}_i^n) &= -(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(v_i^{n,\ell+1})) + \mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n)\\ &\leq -\eta(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n)) + \mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n)\\ &= (1-\eta)(\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n))\end{aligned}$$

which we use to find

$$\begin{aligned}\mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(v_i^{n,\ell}) &= \mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(\widehat{v}_i^n) - (\mathcal{D}_i^n(v_i^{n,\ell}) - \mathcal{D}_i^n(\widehat{v}_i^n))\\ &\geq \mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(\widehat{v}_i^n) - (1-\eta)^\ell(\mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(\widehat{v}_i^n))\\ &\geq (1-(1-\eta)^\ell)(\mathcal{D}_i^n(v_i^{n,0}) - \mathcal{D}_i^n(\widehat{v}_i^n)).\end{aligned}$$

■

Finally, combining Theorem 4.2 with Lemma 6.3 then immediately yields the following corollary.

Corollary 6.4. *Algorithms 1 and 2 with subproblems solved using Algorithm 3 converge in the sense that* $\mathcal{D}(p^n) \to \mathcal{D}(\widehat{p})$. *Furthermore*

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \begin{cases}(1-\frac{\rho\sigma}{2\alpha})^n(\mathcal{D}(p^0) - \mathcal{D}(\widehat{p})) & \text{if } n \leq n_0\\ \frac{2\Phi^2}{\rho\sigma}\alpha^2(n-n_0+1)^{-1} & \text{if } n \geq n_0,\end{cases}$$

where $\alpha := 1 + M\sigma\sqrt{2-\rho+2\sqrt{1-\rho}}$ *for Algorithm 2 and* $\alpha := 1$ *for Algorithm 1,* $\Phi := \sqrt{\|B^{-1}\|}\|\Lambda\|C_\theta R_{\widehat{p}}$, $n_0 := \min\{n \in \mathbb{N}_0 : \mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) < \Phi^2\alpha\}$ *and*

$$\rho = (1-(1-\eta)^{N_{\mathrm{sur}}}), \qquad \eta = \begin{cases}\frac{1}{4\|\tau I - B^{-1}\|\|B\|} & \text{if } \|\tau I - B^{-1}\|\|B\| \geq \frac{1}{2}\\ 1 - \|\tau I - B^{-1}\|\|B\| & \text{otherwise,}\end{cases}$$

for any fixed number of inner surrogate iterations $N_{\mathrm{sur}} \in \mathbb{N}$.

Remark 6.5. In Algorithm 3 we specify a fixed number of surrogate iterations over all subdomain problems, i.e. N_{sur} is the same in each subdomain. However, one may indeed make N_{sur} dependent on Ω_i leading to $N_{sur,i}$ for $i = 1, \ldots, M$. Note that this does not change the statements in Proposition 6.2 and Lemma 6.3 as these estimates only concern the subproblems separately. Moreover, the estimate in Lemma 6.3 is the weaker the smaller $\ell \in \mathbb{N}$ is, since $1-(1-\eta)^a > 1-(1-\eta)^b$ for $a > b$ as $1-\eta \in (0,1)$. Together with this observation we obtain that in the case of subdomain dependent inner surrogate iterations Corollary 6.4 then holds with replacing N_{sur} by $\min_{i\in\{1,\ldots,M\}} N_{sur,i}$, i.e. by the minimal number of inner surrogate iterations over all subdomains.

Remark 6.6. The above presented algorithms and its analysis is not restricted to problem (1.4) and also holds for more general problems of the following type

$$\inf_{p\in K}\left\{\widetilde{\mathcal{D}}(p) := \frac{1}{2}\|\Lambda p - f\|_{B^{-1}}^2\right\}, \tag{6.2}$$

where $\Lambda : V \to W$ is a bounded linear operator, V, W are real Hilbert spaces, $B^{-1} : W \to W$ a positive definite self-adjoint bounded linear operator, $K \subseteq V$ a closed convex set, $f \in W$, and $\|q\|_{B^{-1}}^2 := \langle B^{-1}q, q\rangle_W$ for $q \in W$. This is under assumption that $\widetilde{\mathcal{D}}$ is coercive, ensuring the existence of a solution of (6.2) and that a partition of unity, as defined in (2.1), exists to decompose K. We emphasize, however, that the specific construction in Example 2.1 and the resulting Lemma 2.2 rely on the concrete setting $V = H_0^{\mathrm{div}}(\Omega)^m$ and a direct extension to a more general Hilbert space setting considered in this remark may not be possible.

Nonetheless, the general convergence results established in Theorem 4.2 and Corollary 6.4 apply equally to problem (6.2), yielding the same convergence rates as for (1.4). Note that while our theory holds for such general problems, it may not be optimal in every case, see, e.g. [4].

7. Subsolver Algorithms

Solution strategies for solving (1.4) are especially relevant in our decomposition setting of Algorithms 1 and 2, since we have to solve subproblems (3.2) or (6.1) of the same general form. While for domain decomposition in Algorithms 1 and 2 our Theorem 4.2 only guarantees an asymptotic energy convergence rate of $\mathcal{O}(1/n)$, $n \to \infty$, subsolvers for (1.4) may have different convergence characteristics themselves. While any solver for (1.4) would be applicable, for convenience and later comparison we shortly present as examples both the semi-implicit algorithm of Chambolle [8] and the fast iterative shrinkage-thresholding algorithm (FISTA) by Beck and Teboulle [6] in a setting suitable to us. Note that while we will formulate the subsolver algorithms for (1.4), in order to solve the actual subproblem (3.2) one just needs to accordingly substitute K with $\theta_i K$, λ with the pointwise function $\theta_i \lambda$ and $T^* g$ with f_i^n.

7.1. Semi-implicit dual multiplier method [8]

The semi-implicit Lagrange multiplier method due to Chambolle [8] solves (1.4) for the special case $B = I$. While [8] uses finite differences, we present the algorithm in a Hilbert space setting and for more general B.

As in (1.4) let $K := \{p \in V : |p|_F \leq \lambda\}$ denote the set of feasible dual variables. Similar to [8] there exists a Lagrange multiplier $\mu \in L^\infty(\Omega)$ corresponding to the constraint in K, cf. [26, Theorem 1.6], such that $p \in V$ is a solution of (1.4) if and only if

$$0 = \Lambda^* B^{-1}(\Lambda p - T^* g) + \mu p \tag{7.1}$$

with $\mu \geq 0$, $|p|_F \leq \lambda$ and $\frac{\mu}{2}(|p|_F^2 - \lambda^2) = 0$ holds. Here μp is to be understood as pointwise multiplication.

Recall that $\lambda > 0$. Observing that in a pointwise sense $\mu = 0$ implies $\xi := \Lambda^* B^{-1}(\Lambda p - T^* g) = 0$ and $\mu > 0$ implies $|p|_F = \lambda$ almost everywhere, we deduce from condition (7.1), that in either case $\mu = \frac{|\xi|_F}{\lambda}$. Thus (7.1) becomes

$$0 = \xi + \frac{|\xi|_F}{\lambda} p.$$

The semi-implicit iterative method then uses for some starting value $p^0 \in K$ and stepsize $\tau > 0$ iterates $(p^n)_{n \in \mathbb{N}_0} \subseteq V$ satisfying

$$p^{n+1} = p^n - \tau \left(\xi^n + \frac{|\xi^n|_F}{\lambda} p^{n+1} \right), \tag{7.2}$$

where $\xi^n := \Lambda^* B^{-1}(\Lambda p^n - T^* g)$, $n \in \mathbb{N}_0$. Solving (7.2) for p^{n+1} then yields Algorithm 4.

Algorithm 4 Semi-implicit dual multiplier method [8]

Require: $p^0 \in K$ and $\tau \in (0, \frac{1}{\|\Lambda^* B^{-1} \Lambda\|}]$

1: **for** $n = 0, 1, 2, \ldots$ **do**
2: $\quad \xi^n = \Lambda^* B^{-1}(\Lambda p^n - T^* g)$
3: $\quad p^{n+1} = \lambda \dfrac{p^n - \tau \xi^n}{\lambda + \tau |\xi^n|_F}$
4: **end for**

In contrary to above, where V is $H_0^{\mathrm{div}}(\Omega)$, in the subsequent theorem we require V to be a finite dimensional subspace of L^2, e.g. a suitable discretization. Note that while $\Lambda = \mathrm{div}$ from (1.4) can be defined on $L^2(\Omega)^{d \times m}$ as unbounded operator [27], it is bounded on a finite dimensional subspace $V \subseteq L^2(\Omega)^{d \times m}$.

Theorem 7.1 (cf. [8, Theorem 3.1]). *Let $V \subseteq L^2$ be finite dimensional and $p^0 \in K$. Then Algorithm 4 generates a sequence $(p^n)_{n\in\mathbb{N}_0} \subseteq K$ such that $\mathcal{D}(p^n) \to \mathcal{D}(\widehat{p})$ for $n \to \infty$, where $\widehat{p} \in K$ is a minimizer of (1.4).*

Proof. We follow along the lines of [8, Theorem 3.1]. Notice that $|p^0|_F \leq \lambda$ and thus inductively

$$|p^{n+1}|_F \leq \lambda \frac{|p^n|_F + \tau|\xi^n|_F}{\lambda + \tau|\xi^n|_F} \leq \lambda,$$

i.e. $p^n \in K$ for all $n \in \mathbb{N}$. Let $F : V \to V$ denote the iteration function of Algorithm 4, such that $p^{n+1} = F(p^n)$, $n \in \mathbb{N}_0$. Any fixed point of F or equivalently of (7.2) satisfies the stationary point condition (7.1) per construction and, since $\mathcal{D}$ is convex, will be a minimizer of (1.4).
Denote $\eta^n := \frac{1}{\tau}(p^n - p^{n+1})$ and let $\langle\cdot,\cdot\rangle_V := \langle\cdot,\cdot\rangle_{L^2}$, then we bound the energy difference as

$$\begin{aligned}
\mathcal{D}(p^n) - \mathcal{D}(p^{n+1}) &= -\frac{1}{2}|p^n - p^{n+1}|_*^2 + \langle \mathcal{D}'(p^n), p^n - p^{n+1}\rangle_{L^2} && \text{(Lemma 4.5 (i))}\\
&= -\frac{\tau^2}{2}|\eta^n|_*^2 + \tau\,\langle \xi^n, \eta^n\rangle_{L^2}\\
&= \frac{\tau}{2}(\|\eta^n\|_{L^2}^2 - \tau|\eta^n|_*^2) + \tau\left\langle \xi^n - \frac{1}{2}\eta^n, \eta^n\right\rangle_{L^2}\\
&= \frac{\tau}{2}(\|\eta^n\|_{L^2}^2 - \tau|\eta^n|_*^2) + \frac{\tau}{2}\langle \xi^n - \frac{|\xi^n|_F}{\lambda}p^{n+1}, \xi^n + \frac{|\xi^n|_F}{\lambda}p^{n+1}\rangle_{L^2} && \text{(applying (7.2))}\\
&= \frac{\tau}{2}(\|\eta^n\|_{L^2}^2 - \tau\langle \Lambda^* B^{-1}\Lambda\eta^n, \eta^n\rangle_{L^2}) + \frac{\tau}{2}\left(\|\xi^n\|_{L^2}^2 - \||\xi^n|_F\frac{p^{n+1}}{\lambda}\|_{L^2}^2\right)\\
&\geq \frac{\tau}{2}\big(1 - \tau\|\Lambda^* B^{-1}\Lambda\|\big)\|\eta^n\|_{L^2}^2 + \frac{\tau}{2}\|\xi^n\|_{L^2}^2\left(1 - \operatorname*{ess\,sup}_{x\in\Omega}\frac{|p^{n+1}(x)|_F^2}{\lambda^2}\right)\\
&\geq \frac{1}{2\tau}(1 - \tau\|\Lambda^* B^{-1}\Lambda\|)\|p^n - p^{n+1}\|_{L^2}^2.
\end{aligned}$$

We see that as long as $\tau < \|\Lambda^* B^{-1}\Lambda\|^{-1}$, the sequence $(\mathcal{D}(p^n))_{n\in\mathbb{N}_0}$ is non-increasing and thus, since it is non-negative, also convergent. The feasible set $K \subseteq V$ is closed and bounded, see [21, Lemma 3.2], and compact since V is finite dimensional. Consequently there exists a convergent subsequence $(q^n)_{n\in\mathbb{N}} \subseteq (p^n)_{n\in\mathbb{N}} \subseteq K$, $q^n \to q \in K$ and with continuity of F we get $F(q^n) \to F(q)$. Using the estimate above and the convergence of energies we see that for some $c > 0$ we have $c\|q^n - F(q^n)\|_{L^2}^2 \leq \mathcal{D}(q^n) - \mathcal{D}(F(q^n)) \to 0$ and therefore the limit needs to be a fixed point, $q = F(q)$, and thus a minimizer of (1.4). ∎

7.2. **Fast Iterative Shrinkage-Thresholding Algorithm** [6]

The fast iterative shrinkage-thresholding algorithm (FISTA) due to Beck and Teboulle [6] considers as minimization objective the more general sum of a smooth convex function and a possibly non-smooth one. We apply the algorithm to (1.4) with the non-smooth part incorporating our constraint to K and present it using our notation. For details on the history and derivation of FISTA we refer the reader to [6].

The proximal step [6, (2.6)] with $L := \frac{1}{\tau}$ and $\tau \in (0, \frac{1}{\|\Lambda^* B^{-1}\Lambda\|}]$ requires us to solve

$$\operatorname*{arg\,min}_{p\in K} \frac{1}{2\tau}\|p - \zeta\|_V^2 \tag{7.3}$$

with $\zeta := q - \tau\Lambda^* B^{-1}(\Lambda q - T^* g)$ and $q \in K$. As in (1.4) let $K := \{p \in V : |p|_F \leq \lambda\}$. There exists a Lagrange multiplier $\mu \in L^\infty(\Omega)$ corresponding to the constraint in K, cf. [26, Theorem 1.6], such that

$p \in V$ is a solution of (7.3) if and only if

$$0 = \frac{1}{\tau}(p - \zeta) + \mu p \tag{7.4}$$

with $\mu \geq 0$, $|p|_F \leq \lambda$ and $\frac{\mu}{2}(|p|_F^2 - \lambda^2) = 0$ holds pointwise almost everywhere, where again μp is to be understood as pointwise multiplication. Working in a pointwise sense $\mu = 0$ implies $0 = p - \zeta$ from (7.4) while $\mu > 0$ implies $|p|_F = \lambda$ from the complementary slackness condition. This leads to either $p = \zeta$ or $p = \lambda \frac{\zeta}{|\zeta|_F}$ and thus $p = \mathrm{proj}_K(\zeta)$, where proj_K denotes the pointwise projection to K. Accordingly [6, FISTA with constant stepsize] is given by Algorithm 5.

Algorithm 5 FISTA with constant stepsize [6]

Require: $q^0 = p^0 \in K$, $t^0 = 1$ and $\tau \in (0, \frac{1}{\|\Lambda^* B^{-1} \Lambda\|}]$

1: **for** $n = 0, 1, 2, \ldots$ **do**
2: $\quad p^{n+1} = \mathrm{proj}_K\left(q^n - \tau \Lambda^* B^{-1}(\Lambda q^n - T^* g)\right)$
3: $\quad t^{n+1} = \frac{1}{2}(1 + \sqrt{1 + 4(t^n)^2})$
4: $\quad q^{n+1} = p^{n+1} + \frac{t^n - 1}{t^{n+1}}(p^{n+1} - p^n)$
5: **end for**

Theorem 7.2 (cf. [6, Theorem 4.4 with Remark 2.1])**.** *Algorithm 5 generates a sequence* $(p^n)_{n \in \mathbb{N}_0} \subseteq K$ *respecting*

$$\mathcal{D}(p^n) - \mathcal{D}(\widehat{p}) \leq \frac{2\|\Lambda^* B^{-1} \Lambda\| \|p^0 - \widehat{p}\|_V^2}{(k+1)^2}$$

where $\widehat{p} \in K$ *is a minimizer of* (1.4)*.*

7.3. General Remarks

We want to conclude the section on subsolvers with some comments applying to both Algorithm 4 and Algorithm 5.

Remark 7.3.

(1) Theorems 7.1 and 7.2 guarantee the convergence to a minimal dual energy for both algorithms, which allows us to reconstruct the optimal primal solution due to Proposition 4.3. They do not, however, guarantee convergence of the dual iterates $(p^n)_{n \in \mathbb{N}}$ themselves.

(2) If Λ and B^{-1} are local operators, then the gradient step $\Lambda^* B^{-1}(\Lambda q^n - T^* g)$ is local as well and thus each step of the solver allows for straightforward parallelization over the domain.

(3) A more explicit bound for the maximum stepsize τ can be derived from

$$\|\Lambda^* B^{-1} \Lambda\| \leq \|\nabla\|^2 \|B^{-1}\|.$$

For finite differences as used in this paper (see Definition 8.1 below) we have [8]

$$\|\nabla_h\|_{\ell^2}^2 \leq 8,$$

where $\|\cdot\|_{\ell^2}$ denotes the operator norm induced by the standard Euclidean norm.

8. **Numerical Experiments**

Let for $a, b \in \mathbb{Z}^d$ the discrete domain be given by

$$\Omega_h := \Omega_{h,[a,b]} := \{x = (x_1, \ldots, x_d) \in \mathbb{Z}^d : a \le x \le b\} \subseteq \mathbb{Z}^d.$$

That is, Ω_h is a d-dimensional hyperrectangle.

Computer images given by an array $A \in [0,1]^{n_1 \times \cdots \times n_d}$, $n = (n_1, \ldots, n_d) \in \mathbb{N}^d$ of intensity values between 0 (black) and 1 (white) are then mapped to a discrete function $u_h : \Omega_{h,[1,n]} \to \mathbb{R}$ by defining $u_h(x) := A_x$, where A_x denotes the entry of A at position x.

Definition 8.1 (Finite Difference Operators)**.** For $u_h : \Omega_h \to \mathbb{R}^m$ and $p_h = (p_{h,1}, \ldots, p_{h,d}) : \Omega_h \to \mathbb{R}^{d \times m}$ let forward differences $\partial^+_{h,k} : \Omega_h \to \mathbb{R}^m$ and backward differences $\partial^-_{h,k} : \Omega_h \to \mathbb{R}^m$ be given by

$$\partial^+_{h,k} u_h(x) := \begin{cases} 0 & \text{if } x_k = b_k, \\ u_h(x + he^k) - u_h(x) & \text{otherwise,} \end{cases}$$

$$\partial^-_{h,k} u_h(x) := \begin{cases} u_h(x) & \text{if } x_k = a_k, \\ -u_h(x - he^k) & \text{if } x_k = b_k, \\ u_h(x) - u_h(x - he^k) & \text{otherwise,} \end{cases}$$

where $e^k \in \mathbb{N}^d$ denotes the k-th unit vector, $k = 1, \ldots, d$. The discrete gradient $\nabla_h u_h : \Omega_h \to \mathbb{R}^{d \times m}$ and discrete divergence $\operatorname{div}_h p_h : \Omega_h \to \mathbb{R}^m$ are then defined as

$$\nabla_h u_h := (\partial^+_{h,k} u_h)_{k=1}^d, \qquad \operatorname{div}_h p_h := \sum_{k=1}^d \partial^-_{h,k} p_{h,k}.$$

For a given discrete overlap $r \in \mathbb{N}$ and a desired number of domains $M \in \mathbb{N}$ we first define a discrete covering of Ω_h in dimension $d = 1$. Let $s := |b - a|$ be the diameter of Ω_h, i.e. its length. Define M approximately equal integer sublengths given recursively by

$$s_i := \left\lfloor \frac{s + (M-1)r - \sum_{j=1}^{i-1} s_j}{M - (i-1)} \right\rfloor, \qquad i = 1, \ldots, M,$$

where $\lfloor x \rfloor$ is the floor of $x \in \mathbb{R}$, i.e. the greatest integer less than or equal to x. These give rise to the subdomains

$$\Omega_{h,i} := \{a_i, a_i + 1, \ldots, a_i + s_i =: b_i\}, \quad a_i := a + \sum_{j=1}^{i-1} (s_j - r), \quad i = 1, \ldots, M$$

of diameter s_i and the partition functions $\theta_{h,i} : \Omega_h \to [0,1]$ by

$$\theta_{h,i}(x) := \begin{cases} 1 & \text{if } x \in \Omega_{h,i} \setminus (\Omega_{h,i-1} \cup \Omega_{h,i+1}), \\ \frac{x - a_i}{r} & \text{if } \in \Omega_{h,i} \cap \Omega_{h,i-1}, \\ \frac{a_i + s_i - x}{r} & \text{if } \in \Omega_{h,i} \cap \Omega_{h,i+1}, \\ 0 & \text{if } x \in \Omega_h \setminus \Omega_{h,i} \end{cases} \qquad \text{for } i = 1, \ldots, M,$$

where $\Omega_{h,0} = \Omega_{h,M+1} = \emptyset$. Compare with Figure 8.1 for a decomposition into two and three domains, respectively. The above construction in one dimension yields M discrete subdomains $\Omega_{h,i}$ and a corresponding partition of unity $\theta_{h,i}$ for a discrete domain Ω_h of any size provided M and r are chosen such that $s_i \ge 2r$.

Higher dimensions $d > 1$ are realized through a standard tensor-product formulation based on the above construction, yielding $M = \prod_{k=1}^d M_k$ subdomains ordered lexicographically with overlaps $r = (r_1, \ldots, r_d)$.

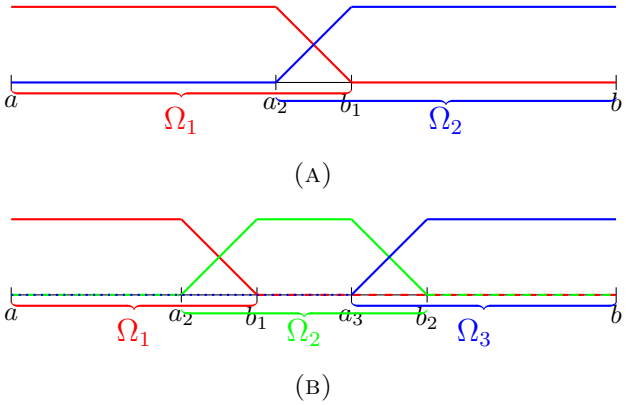


FIGURE 8.1. Overlapping decomposition of an 1d domain into (a) $M = 2$ and (b) $M = 3$ subdomains with the respective partition of unity function.

While we use M here to denote the number of subdomains according to the spatial decomposition, we make use of a chessboard-like coloring similar to [10] shown in Figure 2.1. For Algorithm 1 this allows us to use constant $\sigma = \frac{1}{M_C} = \frac{1}{2^d}$ independent of M, while for Algorithm 2 this means that the sequential order in which the subproblems are solved changes, namely to be lexicographic per color. In all our experiments, we choose $\tau = \frac{1}{8\|B^{-1}\|}$ in Algorithms 4 and 5, in accordance with Remark 7.3. When Algorithm 4 is used, either globally or as a subsolver within the domain decomposition framework, we refer to the configuration as *global*, *dd sequential*, or *dd parallel*, respectively. In contrast, when Algorithm 5 is used, we append *fista* to the label to distinguish it from the variants based on Algorithm 4.

The source code for all following numerical examples has been made available under a permissive license [20].

8.1. **Convergence**

We numerically verify the theoretical sublinear convergence properties of Algorithm 1 and Algorithm 2 due to Theorem 4.2 for three different applications, i.e. image denoising, image inpainting and estimating the optical flow.

For each application described below we compare Algorithms 4 and 5 on the global, non-decomposed problem and the decomposition Algorithms 1 and 2.

For the global algorithm we abort after 10^6 iterations, while for the decomposition algorithms with Algorithm 4 as a subalgorithm solver, we abort after 10,000 outer iterations and each subalgorithm after 100 inner iterations. For a fair comparison we denote with $k \in \mathbb{N}$ the outer iterations of Algorithms 1 and 2 and the iterations of the global algorithm inversely scaled by the number of inner iterations of the decomposition algorithms, that is

$$k = \begin{cases} n & \text{if Algorithm 1 or Algorithm 2 is used;} \\ \frac{n}{100} & \text{if Algorithm 4 is used as global algorithm.} \end{cases}$$

All three algorithms are initialized using $p^0 = 0$. For both Algorithms 1 and 2 in each outer iteration n the subalgorithm on the i-th subdomain is initialized with the current subdomain view $\theta_i p_{i-1}^n \in \theta_i K$.

In each case we downsample input images to a small size of 48×32 pixels and decompose the domain into $M = 2 \cdot 2$ subdomains with an overlap of $r_1 = r_2 = 5$ pixels in order to make a very high number of iterations timely feasible.

The three applications are realized by making use of Proposition 1.1 and setting data g, operator T and model parameters λ, β therein as follows.

Denoising. We start with a ground truth image $\widetilde{g}$ and generate an artificially noisy input $g = \widetilde{g} + \eta$, where η denotes zero-mean additive Gaussian noise with variance 0.1. Setting $T = I$, and choosing model parameters $\lambda = 0.1$, $\beta = 0$ we apply the respective algorithm to obtain the denoised output u.

Inpainting. Starting with a ground truth image $\widetilde{g}$ we artificially mask each pixel with probability $\frac{1}{2}$ by setting its value to 0 (black) to receive a corrupted input image g. Denoting by $A \subseteq \Omega$ the masked area we set $T = \mathbf{1}_{\Omega \setminus A}$ where $\mathbf{1}_{\Omega \setminus A}$ is the indicator function on $\Omega \setminus A$ while the model parameters are chosen to be $\lambda = 5 \cdot 10^{-2}$, $\beta = 1 \cdot 10^{-3}$.

Optical flow estimation. Given two greyscale images $g_0, g_1 : \Omega \to [0, 1]$ we estimate their vector-valued optical flow displacement field u by setting the difference $g = g_0 - g_1$ as input data and T using $Tu = \nabla g_1 \cdot u$. This formulation is the linear approximation of the brightness constancy constraint suitable for small displacements and may be found in [3, (5.81)]. Model parameters are set to $\lambda = 2 \cdot 10^{-3}$, $\beta = 1 \cdot 10^{-3}$. We visualize the optical flow field u and the benchmark-provided ground truth as a color-coded image following [5].

For each of the applications we denote by $\mathcal{D}_{\min}$ the minimum energy obtained by running the global Algorithm 5 for a maximum of 10^7 iterations and subtracting $\varepsilon_{\text{mach}} \approx 2.22 \cdot 10^{-16}$ to accomodate logarithmic plotting. For denoising, inpainting and optical flow estimation we determined $\mathcal{D}_{\min} \approx 456.61$, $\mathcal{D}_{\min} \approx 226.37$ and $\mathcal{D}_{\min} \approx 6.74 \cdot 10^{-2}$ respectively. The energy values over all iterations for the three algorithms and each application are plotted in Figures 8.2 to 8.4 together with respective input and output images.

Comparing parallel and sequential domain decomposition with the same subsolver, we observe in Figure 8.2 behavior similar to that in [10]. The sequential decomposition method exhibits slightly faster decrease in energy in the early iterations than the global algorithm, likely due to the effect of domain overlap. In contrast, the energy curve of the parallel averaging algorithm displays a characteristic bulge in the beginning, presumably corresponding to the linear convergence regime. Note that in constrast to the comparison in [10, Figure 7(a)] we use the largest possible values for σ for each algorithm, i.e. $\sigma = \frac{1}{4}$ for Algorithm 1 and $\sigma = 1$ for Algorithm 2. In Figures 8.3 and 8.4 the performance difference between the sequential and parallel algorithm is less visible for both inpainting and optical flow estimation.

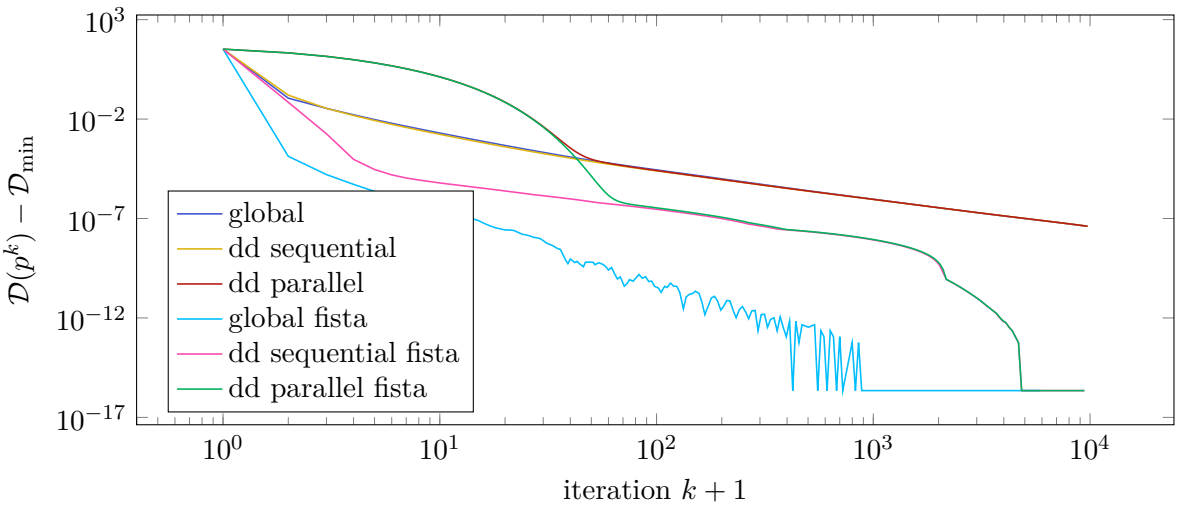
Comparing the results between the global Algorithms 4 and 5, we see unsurprisingly that the accelerated Algorithm 5 clearly exhibits a faster sublinear convergence rate than the global Algorithm 4. When used as subsolvers in a domain decomposition method, both show a similar sublinear convergence rate while Algorithm 5 seems to stay longer in the linear convergence regime. Notably this means that the total convergence rate is limited by the domain decomposition method, which in this case has a slower rate than Algorithm 5.

8.2. **Overlap**

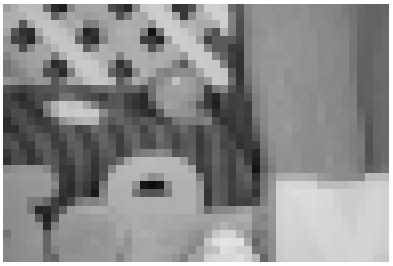
Since for overlapping domain decomposition methods the amount of overlap is known to have significant impact on the performance of the method [47], we perform experiments with varying overlap size r.

We use the same denoising setting from Section 8.1 but using the original size 584×388 of the input image on $4 = 2 \cdot 2$ subdomains running Algorithms 1 and 2 for 1,000 iterations using Algorithm 5 as subsolver with a constant 200 inner iterations. The minimal energy $\mathcal{D}_{\min}$ was obtained by running Algorithm 5 on the global problem for $2 \cdot 10^5$ iterations.

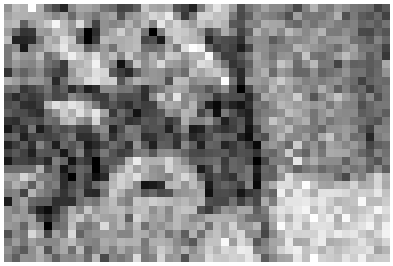
In Figure 8.5 we can see that larger overlap generally corresponds to lower energy for the same number of iterations. While the sublinear asymptotic behaviour seems similar, for Algorithm 1 the



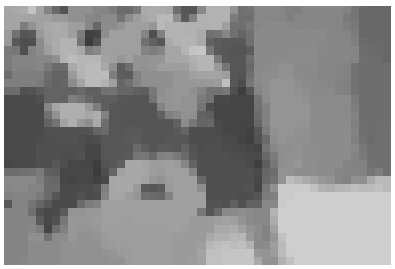
(A) Energy



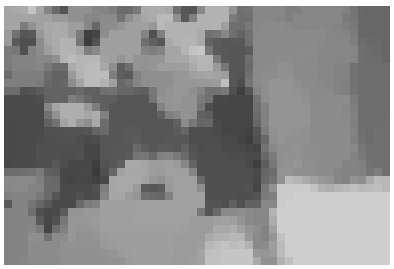
(B) Ground truth image



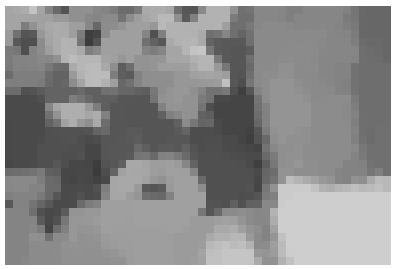
(C) Noisy input image



(D) Denoised output image, global



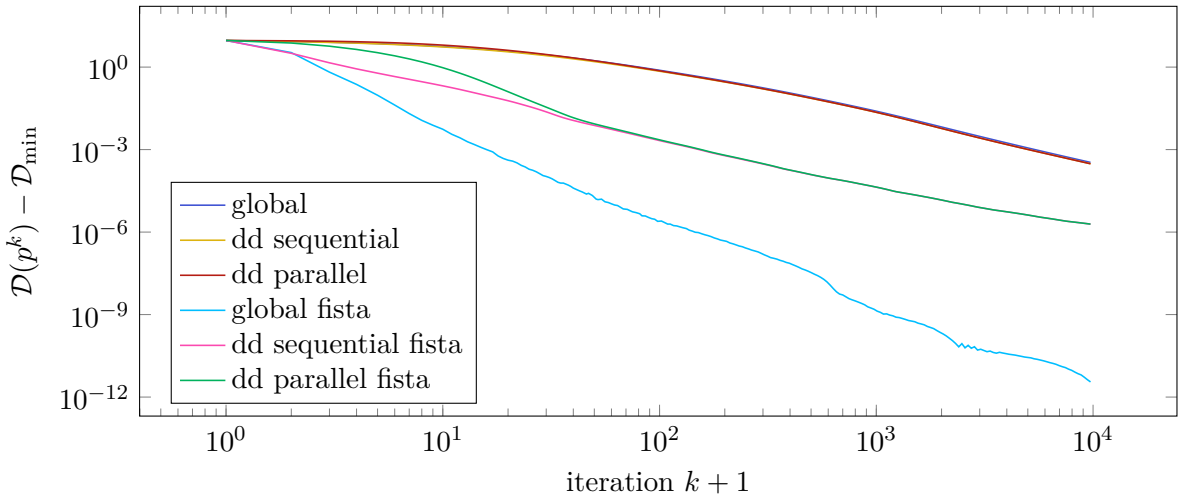
(E) Denoised output image, dd sequential



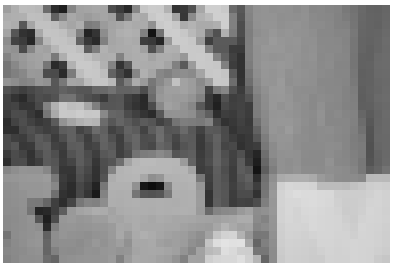
(F) Denoised output image, dd parallel

FIGURE 8.2. Denoising: convergence of energy and results

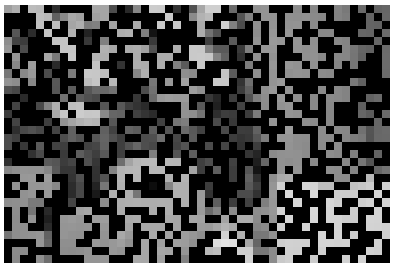
initial linear convergence regime seems to last longer for runs with larger overlap. These observations confirm similar ones made in [44] where pseudo-linear convergence was observed for larger overlaps. Note however that in our example this behaviour seems to scale well only up to an overlap of around 4 pixels, after which diminishing returns start to appear with e.g. the increase from 32 to 128 pixels of overlap not having much of an effect.



(A) Energy



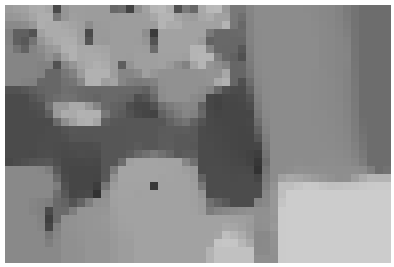
(B) Ground truth image



(C) Corrupted input image

(D) Inpainted output image, global

(E) Inpainted output image, dd sequential



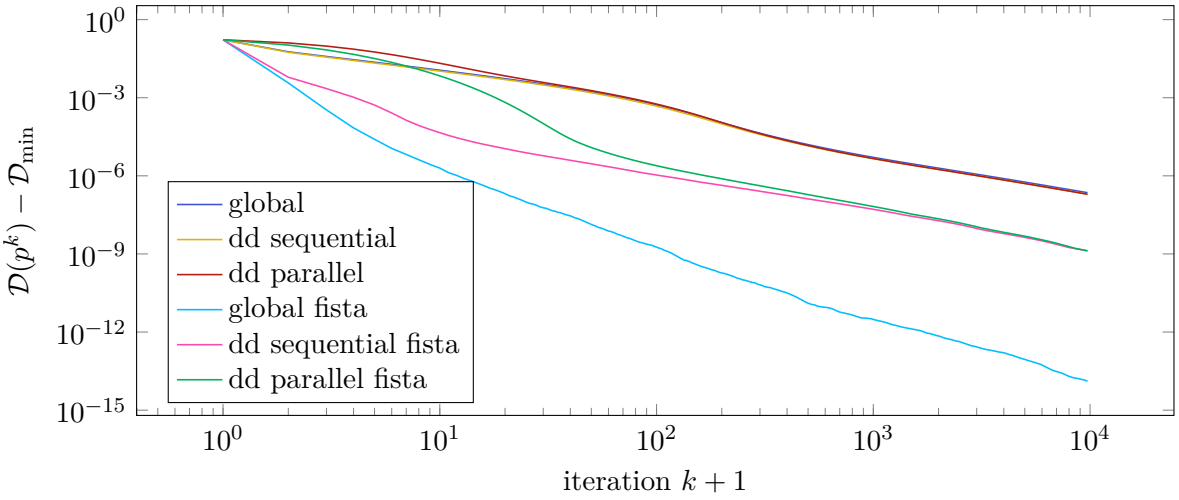
(F) Inpainted output image, dd parallel

Figure 8.3. Inpainting: convergence of energy and results

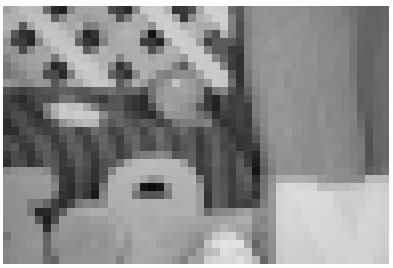
8.3. **Surrogate**

For local operators B we compare (i) nesting the surrogate iteration (Algorithm 3) within domain decomposition and (ii) nesting domain decomposition within a global surrogate iteration using Algorithm 5 as subsolver. Note that for $B = I$, $\tau \to 1$ and a single surrogate iteration both of these are identical.

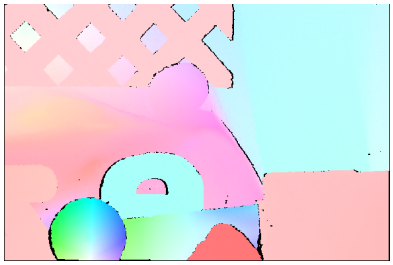
We use the optical flow problem with frames of original size 584×388 pixels and model parameters $\beta = 1 \cdot 10^{-3}$, $\lambda = 1 \cdot 10^{-2}$. The number of subdomains is $M = 4 \cdot 4$ with larger overlap $r_1 = r_2 = 50$ pixels corresponding to the larger image size. We perform for both ways of nesting 50 iterations of



(A) Energy

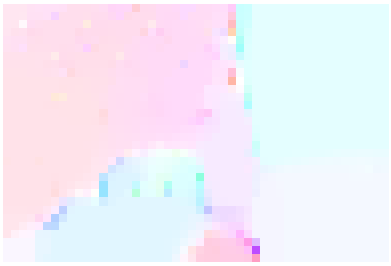


(B) First image f_0 of image sequence

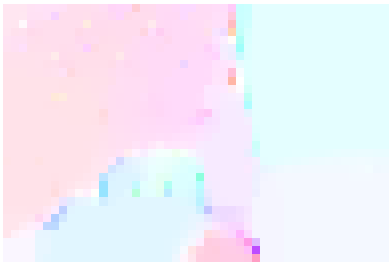


(C) Optical flow ground truth from [5] (original resolution)

(D) Computed optical flow, global



(E) Computed optical flow, dd sequential

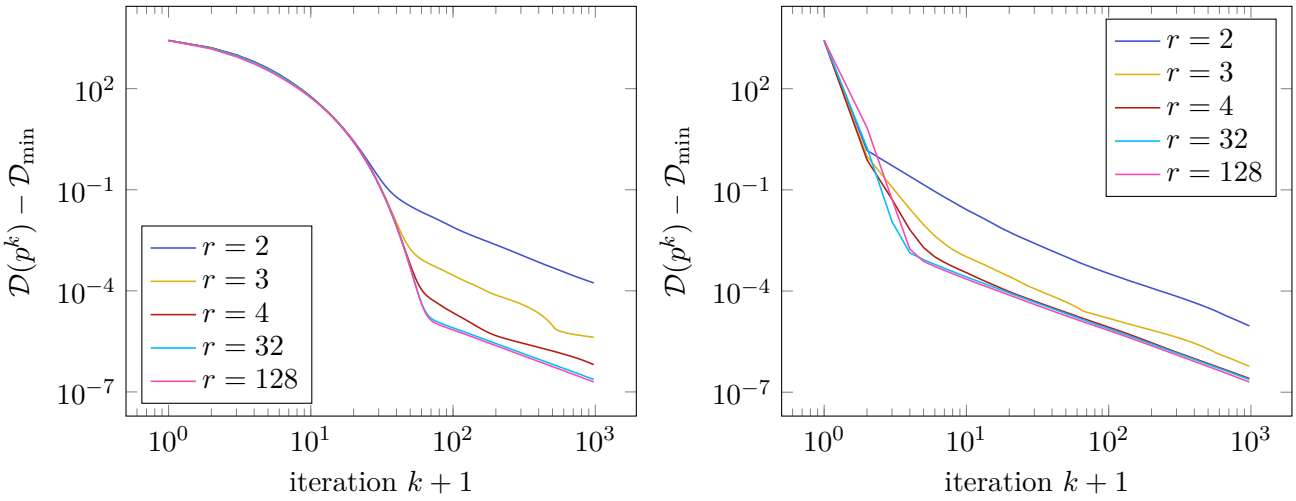


(F) Computed optical flow, dd parallel

FIGURE 8.4. Optical flow: convergence of energy and results

the inner algorithm and stop the whole algorithm after 100 outer iterations. We estimate the minimal energy $\mathcal{D}_{\min} \approx 106.88$ by running Algorithm 5 for 50,000 iterations.

Both nestings perform similarly as can be seen in Figure 8.6, while nesting the surrogate iteration within the domain decomposition has a slight edge. This can be attributed to additional evaluations of B in regions of overlap.



(A) Algorithm 1 (parallel, $\sigma = 0.25$)

(B) Algorithm 2 (sequential, $\sigma = 1$)

FIGURE 8.5. Energy development for varying overlap size r given in pixels, Algorithm 5 used as subsolver

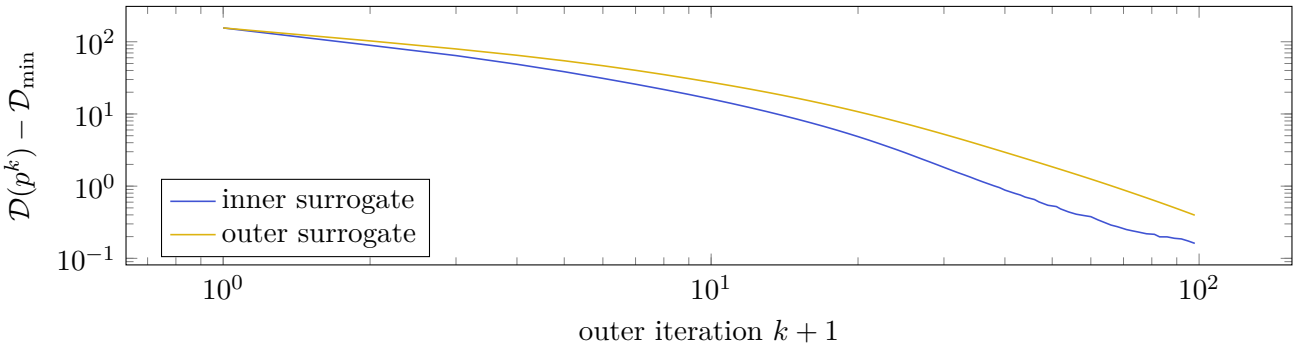


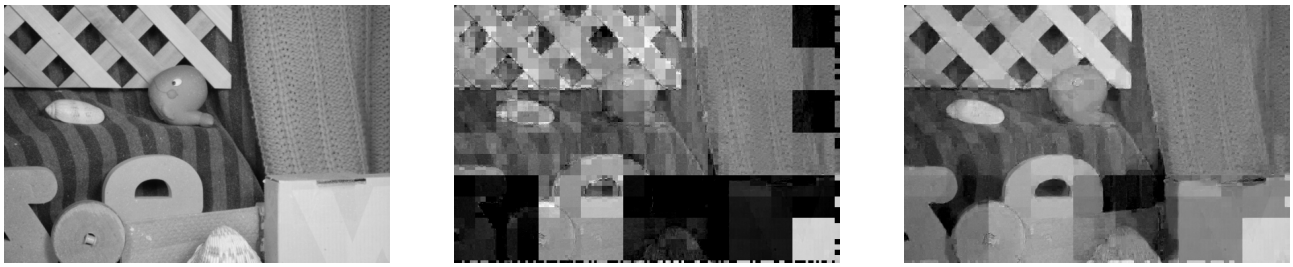
FIGURE 8.6. Comparison of outer and inner surrogate, 50 inner iterations per domain decomposition iteration, one single inner iteration per surrogate iteration, Algorithm 5 used as subsolver

8.4. **Wavelet Transformation**

To demonstrate feasibility of our method even for global operators, we aim to apply it to the reconstruction of corrupted wavelet coefficients. To that end we first define the Wavelet transform T^∞ in a way convenient to us for working with arbitrarily sized images.

Denote $\Omega_s := \Omega_{h,[1,s]}$ for $s \in \mathbb{N}_0^d$ and let $k = k(s) \in \mathbb{N}_0^d$ be such that $2k \leq s \leq 2k+1$. We define the d-dimensional n-th level discrete Haar wavelet transform $T^n : \mathbb{R}^{\Omega_s} \to \mathbb{R}^{\Omega_s}$ recursively by $T^0 := I$ and for $n \geq 1$ by

$$(T^n u)(\alpha \cdot k + x) := \begin{cases} (T^{n-1} T_0 u|_{\Omega_{2k}})(x) & \text{if } \alpha = 0,\ k \geq 1, \\ (T_\alpha u|_{\Omega_{2k}})(x) & \text{if } 0 \neq \alpha \leq 1,\ k \geq 1, \\ u(\alpha \cdot k + x) & \text{otherwise,} \end{cases}$$



(A) Ground truth g_0 (B) Corrupted image $(T^\infty)^{-1}g$ (C) Reconstruction u

FIGURE 8.7. Wavelet inpainting using Algorithm 5 as subsolver

for all $\alpha \cdot k + x \in \Omega_s$, where $u : \Omega_s \to \mathbb{R}$, $\alpha, x \in \mathbb{N}^d$, $x \leq k$ and the transformation $T_\alpha : \mathbb{R}^{\Omega_{2k}} \to \mathbb{R}^{\Omega_k}$ on the orthant indicated by $\alpha \in \{0,1\}^d$ is given by

$$(T_\alpha u)(x) := 2^{-\frac{d}{2}} \sum_{\substack{\beta \in \mathbb{N}_0^d \\ \beta \leq 1}} (-1)^{|\alpha \cdot \beta|} u\big(2(x-1)+1+\beta\big)$$

for all $x \in \mathbb{N}^d$, $x \leq k$. Since $T_\alpha : \mathbb{R}^{\Omega_{2k}} \to \mathbb{R}^{\Omega_k}$ halves the size and for $s \leq 1$ we have $T^n = I$ for any $n \in \mathbb{N}$, the operator T^n becomes idempotent for large enough n and we thus conveniently denote by $T^\infty := \lim_{n\to\infty} T^n$ the full wavelet transform.

We realize the application for wavelet inpainting by again making use of Proposition 1.1 and define data g, operator T and model parameters λ, β therein as follows. We start out with a ground truth image $g_0 \in \mathbb{R}^{\Omega_s}$ of size 584×388 pixels, i.e. $s = (584, 388)$, see Figure 8.7a, and compute artificially corrupted wavelet data $g = Tg_0 := RT^\infty g_0 \in \mathbb{R}^{\Omega_s}$ using an operator R as follows. We select a random subset $J \subseteq \Omega_s$ by choosing every element of Ω_s with probability $\frac{1}{2}$ and define for such fixed J the operator $R = R_J : \mathbb{R}^{\Omega_s} \to \mathbb{R}^{\Omega_s}$ by

$$(Ru)(x) = \begin{cases} u(x) & \text{if } x \notin J, \\ 0 & \text{if } x \in J. \end{cases}$$

For model parameters we use $\lambda = 2 \cdot 10^{-2}$ and $\beta = 1 \cdot 10^{-3}$.

We decompose the domain into $M = 4 \cdot 4$ domains with an overlap of $r_1 = r_2 = 5$ pixels and apply Algorithm 2 with Algorithm 3 as a nested subalgorithm using Algorithm 5 to solve the subproblems. We use $N_{\text{sur}} = 1$ surrogate iterations, 1,000 iterations for the innermost solver and stop the outer decomposition algorithm after just 100 iterations. In Figure 8.7 we can see the used ground truth g_0, the corrupted image visualized as a naive reconstruction $(T^\infty)^{-1}g$ of the corrupted wavelet data g and the result of our wavelet inpainting u. Even in regions where bigger chunks of the corrupted image are lost, wavelet inpainting manages to reconstruct those structures which were preserved by other wavelet coefficients.

8.5. **Scaling of parallel implementation**

Algorithms 1 and 2 allow for a parallel implementation in a domain decomposition setting. Indeed, while the subproblems of Algorithm 1 are independent and may be executed in parallel without additional consideration, Algorithm 2 can be parallelized by solving subproblems for each colored class of subdomains in parallel and for each color in sequence similar to [10].

We use $M = 4 \cdot 8$ subdomains, $r_1 = r_2 = 5$ and 1,000 inner iterations, where each subproblem is solved via Algorithm 5. All the other parameters and image data are the same as in the surrogate comparison above solving the same optical flow problem as in Section 8.3 but with the original image

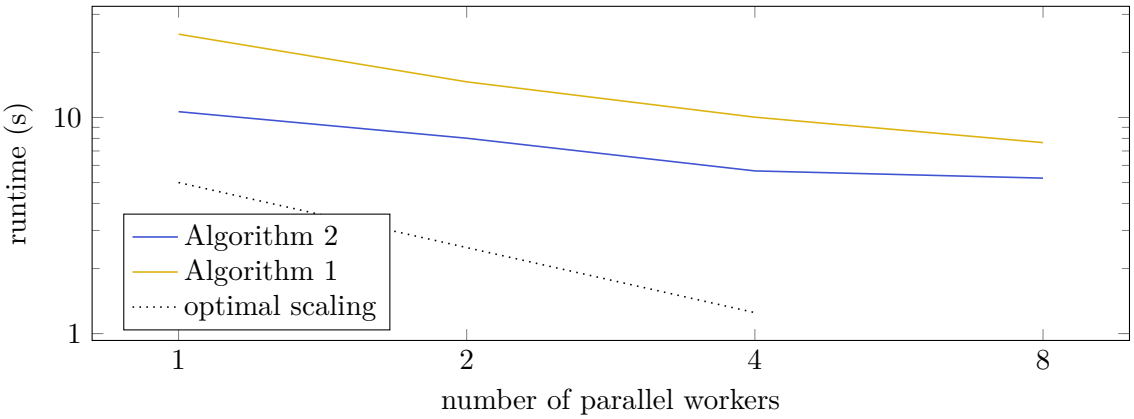


FIGURE 8.8. Time scaling behaviour for our parallel implementation of Algorithm 1 and Algorithm 2 with regard to the number of parallel workers using Algorithm 5 as subsolver

size of 584×388. This means that for each of four colors we have a maximum limit of 8 disjoint subproblems which are scheduled in parallel for Algorithm 2 while Algorithm 1 schedules all subproblems in parallel. We execute the parallel algorithm with $1, 2, 4, 8$ workers respectively on a AMD Ryzen 7 5825U CPU (8 cores) and terminate after reaching an energy of 130.

In Figure 8.8 we can see that a parallel implementation can bring about runtime savings when increasing the number of parallel workers. The runtime behaves almost inversely linear to the number of workers up to the number of available processor cores, though the constant factor is not optimal. We attribute this to the data preparation and communication steps that are carried out on a single worker and apparently do not scale well in this implementation.

9. Conclusion

We have seen that it is possible to improve the domain decomposition convergence rate results from [10] by making use of different proof techniques from alternating minimization. Since as in [10] Algorithm 2 has a slight advantage over Algorithm 1 in terms of iteration count, it suggests that there is still room for improvement of α in Theorem 4.2 in the sequential case. This observation aligns with existing sharp results for sequential methods in specific settings. In particular, in [48], for the linear case, sharp convergence results for alternating projections and subspace correction methods are established, yielding optimal convergence rates in Hilbert spaces. A way to improve convergence rates of decomposition methods is through the incorporation of acceleration techniques. In [9], in the setting of composite quadratic problems, it is shown that an accelerated alternating minimization scheme achieves an $\mathcal{O}(1/n^2)$ rate, where n denotes the iteration number. We note, however, that this rate relies on a Nesterov-type extrapolation step [40]. Without such acceleration, the convergence rate would degrade to $\mathcal{O}(1/n)$, which is consistent with our current setting.

While several accelerated domain decomposition methods have been proposed in the literature (e.g., [35, 49]), these results assume a non-overlapping decomposition and exact subproblem solutions and do not cover the setting studied in this work, i.e., overlapping decompositions and inexact subproblems. In principle, our domain decomposition methods could be accelerated in a similar manner, but the convergence theory would need to be extended to handle overlapping decompositions and inexact local solvers. Developing such a framework remains an open and promising direction for future work.

We could easily apply Algorithms 1 and 2 to a wider range of local image processing tasks, namely inpainting and optical flow estimation, while global operators could only be decomposed by means of the surrogate technique, which incurred an additional cost. When considering the total number of iterations of the inner subalgorithm, Algorithms 1 and 2 did not differ substantially in terms of convergence speed from the global one, i.e. not decomposing the problem, which suggests a minor overhead of the decomposition method. A runtime improvement is only to be expected by parallel execution of the subproblems which we managed to verify in a parallel implementation. Using the decomposition methods in a memory-constrained computing environment is expected to be possible.

References

[1] Luigi Ambrosio, Nicola Fusco, and Diego Pallara. *Functions of Bounded Variation and Free Discontinuity Problems*. Oxford Mathematical Monographs. Clarendon Press, 2000.

[2] Hedy Attouch, Giuseppe Buttazzo, and Gérard Michaille. *Variational Analysis in Sobolev and BV Spaces*. MOS-SIAM Series on Optimization. Mathematical Optimization Society; Society for Industrial and Applied Mathematics, second edition, 2014.

[3] Gilles Aubert and Pierre Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer, second edition, 2006.

[4] Lori Badea. Convergence rate of a Schwarz multilevel method for the constrained minimization of non-quadratic functionals. *SIAM J. Numer. Anal.*, 44(2):449–477, 2006.

[5] Simon Baker, Daniel Scharstein, James P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A Database and Evaluation Methodology for Optical Flow. *Int. J. Comput. Vis.*, 92(1):1–31, 2011.

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

[7] Jakub W. Both. On the rate of convergence of alternating minimization for non-smooth non-strongly convex optimization in Banach spaces. *Optim. Lett.*, 16(2):729–743, 2022.

[8] Antonin Chambolle. An Algorithm for Total Variation Minimization and Applications. *J. Math. Imaging Vis.*, 20(1-2):89–97, 2004.

[9] Antonin Chambolle and Thomas Pock. A remark on accelerated block coordinate descent for computing the proximity operators of a sum of convex functions. *SMAI J. Comput. Math.*, 1:29–54, 2015.

[10] Huibin Chang, Xue-Cheng Tai, Li-Lian Wang, and Danping Yang. Convergence Rate of Overlapping Domain Decomposition Methods for the Rudin–Osher–Fatemi Model Based on a Dual Formulation. *SIAM J. Imaging Sci.*, 8(1):564–591, 2015.

[11] Patrick L. Combettes and Valérie R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4):1168–1200, 2005.

[12] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Commun. Pure Appl. Math.*, 57(11):1413–1457, 2004.

[13] Ingrid Daubechies, Gerd Teschke, and Luminita Vese. Iteratively solving linear inverse problems under general convex constraints. *Inverse Probl. Imaging*, 1(1):29–46, 2007.

[14] Victorita Dolean, Pierre Jolivet, and Frédéric Nataf. *An Introduction to Domain Decomposition Methods: Algorithms, Theory, and Parallel Implementation*. Society for Industrial and Applied Mathematics, 2015.

[15] Joseph C. Dunn. Rates of convergence for conditional gradient algorithms near singular and nonsingular extremals. *SIAM J. Control Optim.*, 17(2):187–211, 1979.

[16] Ivar Ekeland and Roger Témam. *Convex Analysis and Variational Problems*, volume 28 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics, english edition, 1999. Translated from the French.

[17] Massimo Fornasier, Yunho Kim, Andreas Langer, and Carola-Bibiane Schönlieb. Wavelet Decomposition Method for L_2/TV-Image Deblurring. *SIAM J. Imaging Sci.*, 5(3):857–885, 2012.

[18] Massimo Fornasier, Andreas Langer, and Carola-Bibiane Schönlieb. A convergent overlapping domain decomposition method for total variation minimization. *Numer. Math.*, 116(4):645–685, 2010.

[19] Massimo Fornasier and Carola-Bibiane Schönlieb. Subspace correction methods for total variation and l_1-minimization. *SIAM J. Numer. Anal.*, 47(5):3397–3428, 2009.

[20] Stephan Hilb. DualTVDD: Dual total variation decomposition algorithm and related tools, 2021. `https://gitlab.mathematik.uni-stuttgart.de/stephan.hilb/DualTVDD.jl`.

[21] Stephan Hilb, Andreas Langer, and Martin Alkämper. A Primal-Dual Finite Element Method for Scalar and Vectorial Total Variation Minimization. *J. Sci. Comput.*, 96(1): article no. 24 (33 pages), 2023.

[22] Michael Hintermüller and Karl Kunisch. Total bounded variation regularization as a bilaterally constrained optimization problem. *SIAM J. Appl. Math.*, 64(4):1311–1333, 2004.

[23] Michael Hintermüller and Andreas Langer. Subspace correction methods for a class of nonsmooth and nonadditive convex variational problems with mixed L^1/L^2 data-fidelity in image processing. *SIAM J. Imaging Sci.*, 6(4):2134–2173, 2013.

[24] Michael Hintermüller and Andreas Langer. Surrogate Functional Based Subspace Correction Methods for Image Processing. In *Domain Decomposition Methods in Science and Engineering XXI*, pages 829–837. Springer, 2014.

[25] Michael Hintermüller and Andreas Langer. Non-overlapping domain decomposition methods for dual total variation based image denoising. *J. Sci. Comput.*, 62(2):456–481, 2015.

[26] Kazufumi Ito and Karl Kunisch. *Lagrange Multiplier Approach to Variational Problems and Applications*. Advances in Design and Control. Society for Industrial and Applied Mathematics, 2008.

[27] Mikael Kurula and Hans Zwart. The duality between the gradient and divergence operators on bounded Lipschitz domains, 2012. University of Twente.

[28] Andreas Langer. Domain Decomposition for Non-smooth (in Particular TV) Minimization. In *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging: Mathematical Imaging and Vision*, pages 1–47. Springer, 2021.

[29] Andreas Langer and Fernando Gaspoz. Overlapping Domain Decomposition Methods for Total Variation Denoising. *SIAM J. Numer. Anal.*, 57(3):1411–1444, 2019.

[30] Andreas Langer, Stanley Osher, and Carola-Bibiane Schönlieb. Bregmanized domain decomposition for image restoration. *J. Sci. Comput.*, 54(2-3):549–576, 2013.

[31] Chang-Ock Lee and Changmin Nam. Primal Domain Decomposition Methods for the Total Variation Minimization, Based on Dual Decomposition. *SIAM J. Sci. Comput.*, 39(2):B403–B423, 2017.

[32] Chang-Ock Lee, Changmin Nam, and Jongho Park. Domain Decomposition Methods Using Dual Conversion for the Total Variation Minimization with L^1 Fidelity Term. *J. Sci. Comput.*, 78(2):951–970, 2019.

[33] Chang-Ock Lee, Eun-Hee Park, and Jongho Park. A finite element approach for the dual Rudin–Osher–Fatemi model and its nonoverlapping domain decomposition methods. *SIAM J. Sci. Comput.*, 41(2):B205–B228, 2019.

[34] Chang-Ock Lee and Jongho Park. Fast Nonoverlapping Block Jacobi Method for the Dual Rudin–Osher–Fatemi Model. *SIAM J. Imaging Sci.*, 12(4):2009–2034, 2019.

[35] Chang-Ock Lee and Jongho Park. A finite element nonoverlapping domain decomposition method with Lagrange multipliers for the dual total variation minimizations. *J. Sci. Comput.*, 81(3):2331–2355, 2019.

[36] Chang-Ock Lee and Jongho Park. Recent advances in domain decomposition methods for total variation minimization. *J. Korean Soc. Ind. Appl. Math.*, 24(2):161–197, 2020.

[37] Xue Li, Zhenwei Zhang, Huibin Chang, and Yuping Duan. Accelerated Non-Overlapping Domain Decomposition Method for Total Variation Minimization. *Numer. Math., Theory Methods Appl.*, 14(4):1017–1041, 2021.

[38] Julien Mairal. Optimization with first-order surrogate functions. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 783–791. PMLR, 2013.

[39] Reinhard Nabben. Comparisons between multiplicative and additive Schwarz iterations in domain decomposition methods. *Numer. Math.*, 95:145–162, 2003.

[40] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. *Sov. Math., Dokl.*, 27(2):372–376, 1983.

[41] Jongho Park. Additive Schwarz methods for convex optimization as gradient methods. *SIAM J. Numer. Anal.*, 58(3):1495–1530, 2020.

[42] Jongho Park. An overlapping domain decomposition framework without dual formulation for variational imaging problems. *Adv. Comput. Math.*, 46(4): article no. 57 (29 pages), 2020.

[43] Jongho Park. Accelerated additive Schwarz methods for convex optimization with adaptive restart. *J. Sci. Comput.*, 89(3): article no. 58 (20 pages), 2021.

[44] Jongho Park. Pseudo-linear convergence of an additive Schwarz method for dual total variation minimization. *Electron. Trans. Numer. Anal.*, 54:176–197, 2021.

[45] Alfio Quarteroni and Alberto Valli. *Domain Decomposition Methods for Partial Differential Equations*. Clarendon Press, 1999.

[46] Xue-Cheng Tai. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. *Numer. Math.*, 93(4):755–786, 2003.

[47] Andrea Toselli and Olof B. Widlund. *Domain Decomposition Methods: Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, 2005.

[48] Jinchao Xu and Ludmil Zikatanov. The method of alternating projections and the method of subspace corrections in Hilbert space. *J. Am. Math. Soc.*, 15(3):573–597, 2002.

[49] Zhenwei Zhang, Huibin Chang, and Yuping Duan. Fast Non-overlapping Domain Decomposition Methods for Continuous Multi-phase Labeling Problem. *J. Sci. Comput.*, 97(3): article no. 67 (29 pages), 2023.