Choose your path wisely: gradient descent in a Bregman distance framework*

Martin Benning†, Marta M. Betcke‡, Matthias J. Ehrhardt§, and Carola-Bibiane Schönlieb¶

Abstract. We propose an extension of a special form of gradient descent — in the literature known as linearised Bregman iteration — to a larger class of non-convex functions. We replace the classical (squared) two norm metric in the gradient descent setting with a generalised Bregman distance, based on a proper, convex and lower semi-continuous function. The algorithm's global convergence is proven for functions that satisfy the Kurdyka-Łojasiewicz property. Examples illustrate that features of different scale are being introduced throughout the iteration, transitioning from coarse to fine. This coarse-to-fine approach with respect to scale allows to recover solutions of non-convex optimisation problems that are superior to those obtained with conventional gradient descent, or even projected and proximal gradient descent. The effectiveness of the linearised Bregman iteration in combination with early stopping is illustrated for the applications of parallel magnetic resonance imaging, blind deconvolution as well as image classification with neural networks.

Key words. Non-convex Optimisation, Non-smooth Optimisation, Gradient Descent, Bregman Iteration, Linearised Bregman Iteration, Parallel MRI, Blind Deconvolution, Deep Learning

AMS subject classifications. 49M37, 65K05, 65K10, 90C26, 90C30

1. Introduction.

Non-convex optimisation methods are indispensable mathematical tools for a large variety of applications [62]. For differentiable objectives, first-order methods such as gradient descent have proven to be useful tools in all kinds of scenarios. Throughout the last decade, however, there has been an increasing interest in first-order methods for non-convex and non-smooth objectives. These methods range from forward-backward, respectively proximal-type, schemes [2, 3, 4, 18, 19], over linearised proximal schemes [80, 16, 81, 61], to inertial methods [63, 68], primal-dual algorithms [78, 52, 57, 12], scaled gradient projection methods [69] and non-smooth Gauß-Newton extensions [35, 64].

In this paper, we follow a different approach of incorporating non-smoothness into first-order methods for non-convex problems. We present a direct generalisation of gradient descent, first introduced in [10], where the usual squared two-norm metric that penalises the gap of two subsequent iterates is being replaced by a potentially non-smooth distance term. This distance term is given in form of a generalised Bregman distance [20, 22, 66], where the underlying function is proper, lower semi-continuous and convex, but not necessarily smooth. If the underlying function is a Legendre function (see [73, Section 26] and [7]), the proposed gener-

*Submitted to the editors DATE.

Funding: This work was funded by the Leverhulme Trust Early Career Fellowship 'Learning from mistakes: a supervised feedback-loop for imaging applications', the Isaac Newton Trust, the Engineering and Physical Sciences Research Council (EPSRC) 'EP/K009745/1', the Leverhulme Trust project 'Breaking the non-convexity barrier', the EPSRC grant 'EP/M00483X/1', the EPSRC centre 'EP/N014588/1', the Cantab Capital Institute for the Mathematics of Information and CHiPS (Horizon 2020 RISE project grant).

†School of Mathematical Sciences, Queen Mary University of London, UK (m.benning@qmul.ac.uk).

‡Department of Computer Science, University College London, UK (m.betcke@ucl.ac.uk).

§Institute for Mathematical Innovation, University of Bath, UK (m.ehrhardt@bath.ac.uk).

¶Department of Applied Mathematics and Theoretical Physics, University of Cambridge, UK (cbs31@cam.ac.uk).

alisation basically coincides with the recently proposed non-convex extension of the Bregman proximal gradient method [17]. In the more general case, the proposed method is a generalisation of the so-called linearised Bregman iteration [33, 83, 25, 24] to non-convex data fidelities.

Motivated by inverse scale space methods (cf. [21, 22, 66]), the use of non-smooth Bregman distances for the penalisation of the iterates gap allows to control the scale of features present in the individual iterates. Replacing the squared two-norm, for instance, with a squared two-norm plus the Bregman distance w.r.t. a one-norm leads to very sparse initial iterates, with iterates becoming more dense throughout the course of the iteration. This control of scale, i.e. the slow evolution from iterates with coarse structures to iterates with fine structures, can help tp overcome unwanted minima of a non-convex objective, as we are going to demonstrate with an example in Section 2. This is in stark contrast to many of the non-smooth, non-convex first-order approaches mentioned above, where the methods are often initialised with random inputs that become more regular throughout the iteration.

Our main contributions of this paper are the generalisation of the linearised Bregman iteration to non-convex functions, a detailed convergence analysis of the proposed method as well as the presentation of numerical results that demonstrate that the use of coarse-to-fine scale space approaches in the context of non-convex optimisation can lead to superior solutions.

The outline of the paper is as follows. Based on the non-convex problem of blind deconvolution, we first give a motivation in Section 2 of why a coarse-to-fine approach in terms of scale can indeed lead to superior solutions of non-convex optimisation problems. We then recall key concepts of convex and non-convex analysis that are needed throughout the paper in Section A. Subsequently, we define the extension of the linearised Bregman iteration for non-convex functions in Section 3. Then, motivated by the informal convergence recipe of Bolte et al. [16, Section 3.2] we show a global convergence result in Section 4, which concludes the theoretical part. We conclude with the modelling of the applications of parallel Magnetic Resonance Imaging (MRI), blind deconvolution and image classification in Section 5, followed by corresponding numerical results in Section 6 as well as conclusions and outlook in Section 7.

2. Motivation.

We want to motivate the use of the linearised Bregman iteration for non-convex optimisation problems with the example of blind deconvolution. In blind (image) deconvolution the goal is to recover an unknown image u from a blurred and usually noisy image f. Assuming that the degradation is the same for each pixel, the problem of blind deconvolution can be modelled as the minimisation of the energy

$$E_1(u,h) := \underbrace{\frac{1}{2}\|u * h - f\|_2^2}_{=:F(u,h)} + \chi_C(h)\,, \tag{2.1}$$

with respect to the arguments $u \in \mathbb{R}^n$ and $h \in \mathbb{R}^r$. Here $*$ denotes a discrete convolution operator, and χ_C, is the characteristic function

$$\chi_C(h) := \begin{cases} 0 & h \in C \\ \infty & h \notin C \end{cases},$$

(a) Original image $\hat{u}$ (b) f and $\hat{h}$ (c) Projected gradient descent

(d) $\alpha = 10^{-3}$ (e) $\alpha = 10^{-4}$

Figure 2.1. ***Standard approaches for blind deconvolution.*** *Figure 2.1a shows the image $\hat{u}$ of Pixel the Gambian pouched rat, courtesy of Monique Boddington. Figure 2.1b shows a motion-blurred version f of that same image; the corresponding convolution kernel $\hat{h}$ is depicted in the bottom left corner. Figure 2.1c visualises the reconstruction of the image and the convolution kernel obtained with the projected gradient descent method (2.2). In Figure 2.1d we see the result of gradient descent method (2.4) for $\alpha = 10^{-3}$, whereas Figure 2.1e shows the result of (2.4) for the choice $\alpha = 10^{-4}$.*

defined over the simplex constraint set

$$C := \left\{ h \in \mathbb{R}^r \;\middle|\; \sum_{j=1}^{r} h_j = 1,\ h_j \geq 0,\ \forall j \in \{1, \dots, r\} \right\} .$$

Even with data f in the range of the non-linear convolution operator, i.e. $f = \hat{u} * \hat{h}$ for some $\hat{u} \in \mathbb{R}^n$ with $\hat{h} \in C$, it is usually still fairly challenging to recover $\hat{u}$ and $\hat{h}$ as solutions of (2.1). A possible reason for this could be that (2.1) is an invex function on $\mathbb{R}^n \times C$, where every stationary point is already a global minimum. If we simply try to recover $\hat{u}$ and $\hat{h}$ via projected gradient descent, we usually require an initial point in the neighbourhood of $(\hat{u}, \hat{h})$ in order to converge to that point. We want to illustrate this with a concrete example. Assume we are given an image $\hat{u}$ and a convolution kernel $\hat{h}$ as depicted in Figure 2.1, and $f = \hat{u} * \hat{h}$ is as shown in Figure 2.1b. Minimising (2.1) via projected gradient descent leads to the following procedure:

$$u^{k+1} = u^k - \tau^k \, \partial_u F(u^k, h^k) \,, \tag{2.2a}$$

$$h^{k+1} = \mathrm{proj}_C \left(h^k - \tau^k \, \partial_h F(u^k, h^k) \right) , \tag{2.2b}$$

where proj_C denotes the projection onto the convex set C. If we initialise with $u^0 = (0, \dots, 0)^T$ and $h^0 = (1, \dots, 1)^T / r$, set $\tau^0 = 1$, update τ^k via backtracking to ensure a monotonic decrease of the energy E_1, and iterate (2.2) for 3500 iterations, we obtain the reconstructions visualised in Figure 2.1c. Even without any noise present in the data f, the algorithm converges to a solution very different from $\hat{u}$ and $\hat{h}$. This is not necessarily surprising as we do not impose any regularity on the image. We can try to overcome this issue by modifying (2.1) as follows:

$$\begin{aligned} E_2(u,h) &:= F(u,h) + \chi_C(h) + \alpha \mathrm{TV}(u)\,, \\ &= E_1(u,h) + \alpha \mathrm{TV}(u)\,. \end{aligned} \tag{2.3}$$

Here TV denotes the discretised total variation, i.e.

$$\mathrm{TV}(u) := \| |\nabla u| \|_1\,,$$

where $\nabla : \mathbb{R}^n \to \mathbb{R}^{2n}$ is a (forward) finite difference discretisation of the gradient operator, $|\cdot|$ the Euclidean vector norm and $\|\cdot\|_1$ the one-norm, and α is a positive scalar. The minimisation of (2.3) can easily be carried out by the proximal gradient descent method, also known as forward-backward splitting [54], which is a minor modification of the projected gradient method [41, 42, 13] to more general proximal mappings. In the context of minimising (2.3), the proximal gradient method reads as

$$u^{k+1} = (I + \alpha \partial \mathrm{TV})^{-1}(u^k - \tau^k\, \partial_u F(u^k, h^k))\,, \tag{2.4a}$$

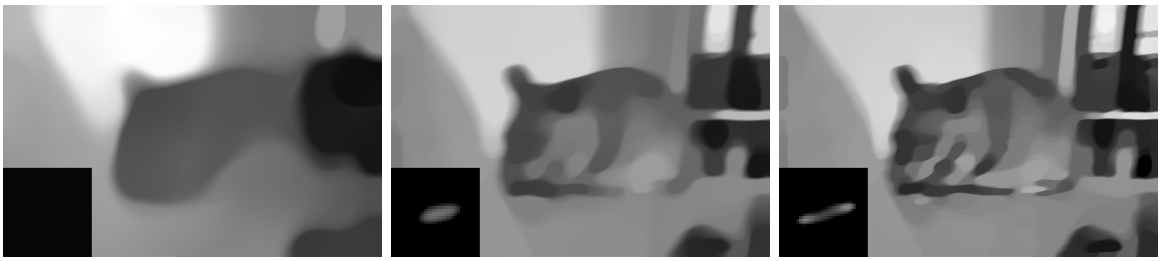
$$h^{k+1} = \mathrm{proj}_C\left(h^k - \tau^k\, \partial_h F(u^k, h^k)\right)\,, \tag{2.4b}$$

where $(I + \alpha \partial \mathrm{TV})^{-1}$ denotes the proximal mapping [58, 59] with respect to the total variation, i.e.

$$(I + \alpha \partial \mathrm{TV})^{-1}(z) := \operatorname*{arg\,min}_{u \in \mathbb{R}^n} \left\{ \frac{1}{2}\|u - z\|_2^2 + \alpha \mathrm{TV}(u) \right\}\,. \tag{2.5}$$

It is straight-forward to solve (2.5) for a given argument with numerical methods such as the (accelerated) primal-dual hybrid gradient method (cf. [84, 67, 37, 28, 29]) up to sufficient numerical accuracy. If we then evaluate 3000 iterations of (2.4) for $\alpha \in \{10^{-3}, 10^{-4}\}$ with the same initial values that we used for the projected gradient method, we obtain the results visualised in Figure 2.1. We observe that for the larger choice of $\alpha = 10^{-3}$ we obtain a better reconstruction of the convolution kernel, but at the cost of a reconstructed image that is very cartoon-like. Reducing the parameter α to $\alpha = 10^{-4}$ reduces the impact of the total variation regularisation; however, the reconstructed image then remains fairly blurry and the reconstructed convolution kernel is closer to a Dirac delta.

The reason for this is that the total variation-based model (2.3) is basically not suitable for deconvolution tasks. Blurred images generally have a smaller total variation compared to their sharp counterparts, hence it is easier to minimise the energy in (2.3) by recovering a kernel close to a Dirac delta and a smoothed version of the blurry image in order to reduce the total variation.



(a) 1st iterate (b) 10th iterate (c) 50th iterate

(d) 500th iterate (e) 1500th iterate (f) 3000th iterate

Figure 2.2. ***Proposed approach for blind deconvolution.*** *Figure 2.2 shows several iterates of the linearised Bregman iteration (2.6) for the choice $\alpha = 0.05$. The strong initial effect of the total variation regularisation enables the algorithm to converge to a solution close to $\hat{u}$ and $\hat{h}$.*

We therefore want to use an alternative approach that is different to the two approaches presented above. We do observe from the proximal gradient example that a larger regularisation parameter seems to work better for a more accurate reconstruction of the convolution kernel (at the cost of a rather cartoon-like image). The explanation for this is that image features at a relatively coarse scale have to be adjusted to minimise the data fit, forcing the convolution kernel to correct for this. It therefore seems reasonable to find a minimiser of (2.1) with a scale-space approach, changing from coarse to fine scales over the course of the iteration. Specifically, we propose to use a variant of the linearised Bregman iteration adopted to minimising non-convex problems such as the minimisation of the function E_1 as defined in (2.1). For the choice of E_1 in (2.1), this method reads as

$$u^{k+1} = \operatorname*{arg\,min}_{u \in \mathbb{R}^n} \left\{ \frac{1}{2}\|u - u^k\|^2 + \tau^k \left(\alpha D_{\mathrm{TV}}^{q^k}(u, u^k) + \langle \partial_u F(u^k, h^k), u \rangle \right) \right\}, \tag{2.6a}$$

$$q^{k+1} = q^k - \frac{1}{\tau^k \alpha} \left(u^k - u^{k+1} - \tau^k \, \partial_u F(u^k, h^k) \right), \tag{2.6b}$$

$$h^{k+1} = \mathrm{proj}_C \left(h^k - \tau^k \, \partial_h F(u^k, h^k) \right). \tag{2.6c}$$

Here $q^k \in \partial \mathrm{TV}(u^k)$ denotes a subgradient of TV at u^k, $\alpha \geq 0$ is a scalar and $D_{\mathrm{TV}}^{q^k}(u^{k+1}, u^k)$ is the generalised Bregman distance [20] with respect to the total variation, i.e.

$$D_{\mathrm{TV}}^{q^k}(u^{k+1}, u^k) = \mathrm{TV}(u^{k+1}) - \mathrm{TV}(u^k) - \langle q^k, u^{k+1} - u^k \rangle,$$

Algorithm 3.1 Generalised linearised Bregman iteration for minimising E

Initialise $\{\tau^k\}_{k\in\mathbb{N}}$, u^0 and $p^0 \in \partial J(u^0)$
for $k = 0, 1, \ldots$ **do**
  Compute $u^{k+1} = \arg\min_{u\in\mathbb{R}^n} \left\{ \tau^k \langle u - u^k, \nabla E(u^k)\rangle + D_J^{p^k}(u, u^k) \right\}$
  Compute $p^{k+1} = p^k - \tau^k \nabla E(u^k)$
end for

for a subgradient $q^k \in \partial \mathrm{TV}(u^k)$. Note that (2.6) reduces to the projected gradient method (2.2) for the choice $\alpha = 0$.

Replacing the total variation semi-norm in (2.4) with its Bregman distance yields an iterative scale-space method that changes the influence of the total variation regularisation throughout the course of the iteration. With a larger parameter α, the initial iterates have a very low total variation and contain only coarse features. Throughout the iteration, features of finer and finer scale are introduced. We have visualised several iterates of (2.6) for the choice $\alpha = 0.05$ in Figure 2.2 to demonstrate this phenomenon.

We observe that this modification of projected gradient descent enables us to converge to minimisers of E_1 as defined in (2.1) that are fairly close to the original choices of $\hat{u}$ and $\hat{h}$. Hence, the choice of Bregman distance strongly affects the outcome of the iteration procedure and can be used to guide the iterates towards more desirable outcomes.

Obviously real data is never in the range of the forward model, and in that case we do not want to converge to a minimiser of E_1. However, we can still apply the linearised Bregman iteration in combination with early stopping in order to produce superior results compared to projected or proximal gradient descent, which we will further demonstrate in Section 5 and Section 6. Prior to this, we provide a comprehensive convergence analysis of the linearised Bregman iteration in the Sections 3 and 4.

3. Linearised Bregman iteration for non-convex problems.

We are interested in the minimisation of functions $E \in \mathcal{S}_L$, where $\mathcal{S}_L$ is defined in Definition A.8 in the appendix. We want to emphasise that the function E does not necessarily have to be convex. In order for the minimisation of E to make sense, we have to introduce some additional assumptions for this function first. From now on we assume $E \in \Psi_L$, with Ψ_L being defined as

$$\Psi_L := \left\{ E \in \mathcal{S}_L \;\middle|\; \begin{array}{l} E \text{ has bounded level sets} \\ E \text{ is bounded from below} \end{array} \right\}.$$

We further recall the definition of the set of critical points of E, i.e.

$$\mathrm{crit}(E) := \{ u \in \mathrm{dom}(E) \,|\, \nabla E(u) = 0 \}\,. \tag{3.1}$$

The requirements on E ensure that sequences $\{u^k\}_{k\in\mathbb{N}}$ are already bounded if the sequences $\{E(u^k)\}_{k\in\mathbb{N}}$ are bounded, that an infimum exists and that the set of critical points is non-empty.

We want to minimise E iteratively in a way that allows us to follow solution paths of different regularity. This regularity will be induced by an additional function $J \in \Gamma_0$, where

Γ_0 is defined in the appendix. Precisely, we approach the minimisation of E via the linearised Bregman iteration

$$u^{k+1} = \operatorname*{arg\,min}_{u\in\mathbb{R}^n} \left\{ \tau^k \langle \nabla E(u^k), u - u^k \rangle + D_J^{p^k}(u, u^k) \right\}, \tag{3.2a}$$

$$p^{k+1} = p^k - \tau^k \nabla E(u^k), \tag{3.2b}$$

for $k \in \mathbb{N}$, a sequence of positive parameters $\{\tau^k\}_{k\in\mathbb{N}}$ and initial values u^0 and p^0 with $p^0 \in \partial J(u^0)$. Here ∂J denotes the subdifferential; we refer to the appendix for its definition. Note that (3.2b) is simply the optimality condition of (3.2a). If J is differentiable, ∂J is single-valued and we do not have to compute (3.2b) as we do not need to pick a specific element from the set. However, if ∂J is multivalued, (3.2) guarantees $p^{k+1} \in \partial J(u^{k+1})$ for all $k \in \mathbb{N}$. This general form of linearised Bregman iteration for the minimisation of non-convex functions is summed up in Algorithm 3.1.

Remark 1. *For $J(u) = \frac{1}{2}\|u\|^2$, (3.2) (and therefore also Algorithm 3.1) reduces to classical gradient descent. Hence, the linearised Bregman iteration is indeed a generalisation of gradient descent.*

Based on what has become known as the Bregman iteration [27, 76, 36, 46, 65], the linearised Bregman iteration has initially been proposed in [33] for the computation of sparse solutions of underdetermined linear systems of equations. It has been extensively studied in this context (cf. [83, 25, 24]) and also in the context of the minimisation of more general convex functions (see [82]). It is also closely linked to (linearised variants of) the alternating direction method of multipliers (ADMM) [39], as well as generalisations to non-quadratic Bregman distances [79]. It has further been analysed in the context of non-linear inverse problems in [5]. In [10], the linearised Bregman iteration has been studied in the context of minimising general smooth but non-convex functions. Algorithm 3.1 allows us to control the scale of the iterates, depending on the choice of J. Note that we can also reformulate (3.2a) as follows:

$$u^{k+1} = \operatorname*{arg\,min}_{u\in\mathbb{R}^n} \left\{ \tau^k \left\langle \nabla E(u^k) - \frac{1}{\tau^k} p^k, u - u^k \right\rangle + J(u) \right\}. \tag{3.3}$$

In order to ensure that a solution of Update (3.3) (respectively (3.2a)) exists, we choose J such that $J(u) + \tau^k \langle u^*, u\rangle$ is coercive for all $u^* \in \mathbb{R}^n$. In particular, we choose J to be of the form $J_k := \frac{1}{2}\|\cdot\|^2 + \tau^k R$, where $R \in \Gamma_0$. For this choice the iterates (3.2) read as

$$\begin{aligned} u^{k+1} &= \operatorname*{arg\,min}_{u\in\mathbb{R}^n} \left\{ \tau^k \left(\langle \nabla E(u^k), u - u^k \rangle + D_R^{q^k}(u, u^k) \right) + \frac{1}{2}\|u - u^k\|^2 \right\}, \\ &= \left(I + \tau^k \partial R \right)^{-1} \left(u^k + \tau^k \left(q^k - \nabla E(u^k) \right) \right), \end{aligned} \tag{3.4a}$$

$$q^{k+1} = q^k - \frac{1}{\tau^k} \left(u^{k+1} - u^k + \tau^k \nabla E(u^k) \right), \tag{3.4b}$$

for $q^k \in \partial R(u^k)$. Note that (3.4b) can be written as

$$q^{k+1} = q^0 - \sum_{n=0}^{k} \left[\frac{1}{\tau^n} (u^{n+1} - u^n) \right] - \sum_{n=0}^{k} \nabla E(u^n), \tag{3.5}$$

Algorithm 3.2 Specialised linearised Bregman iteration for minimising E

Initialise $\{\tau^k\}_{k\in\mathbb{N}}$, u^0 and $q^0 \in \partial R(u^0)$
for $k = 0, 1, \ldots$ **do**
  Get $u^{k+1} = \left(I + \tau^k \partial R\right)^{-1} \left(u^k + \tau^k \left(q^k - \nabla E(u^k)\right)\right)$
  Compute $q^{k+1} = q^k - \frac{1}{\tau^k}\left(u^{k+1} - u^k + \tau^k \nabla E(u^k)\right)$
end for

and hence, for constant stepsize $\tau^k = \tau$ (3.4a) simplifies to

$$u^{k+1} = (I + \tau\partial R)^{-1}\left(u^0 + \tau q^0 - \tau \sum_{n=0}^{k} \nabla E(u^n)\right). \tag{3.6}$$

Equations (3.4) are summarised in Algorithm 3.2. Note that both Algorithm 3.2 and Equation (3.6) demonstrate that this specialised linearised Bregman iteration is indeed different to proximal gradient descent, for which one iterate reads $u^{k+1} = (I + \tau\partial R)^{-1}\left(u^k - \tau\nabla E(u^k)\right)$. Instead, from Equation (3.4a) we observe that one computes a subgradient descent step in the direction of the subgradient of $E - R$, followed by an application of the proximal step with respect to R.

In the following we prove decrease properties and a global convergence result for Algorithm 3.2.

4. A global convergence result for Algorithm 3.2. The convergence analysis is inspired by the global convergence recipe of [16]. It is an extension to a class of non-smooth surrogate functions for which a tailored convergence analysis is presented that utilises the convexity of R. We begin our analysis of Algorithm 3.2 by showing a sufficient decrease property of the surrogate function and a subgradient bound by the (primal) iterates gap. In order to do so, we first define the following surrogate function for E.

Definition 4.1 (Surrogate objective). *Assume $E \in \Psi_L$ and $R \in \Gamma_0$. Then we define a surrogate function $F : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ as*

$$F(x, y) := E(x) + R(x) + R^*(y) - \langle x, y\rangle. \tag{4.1}$$

Here R^* denotes the convex conjugate of R as defined in Definition A.3 in the appendix.

Note that based on Remark 6 in the appendix, the surrogate function (4.1) satisfies

$$F(x, y) := E(x) + D_R^y(x, z)\,,$$

for any $z \in \partial R^*(y)$, which implies $F(x, y) \geq E(x)$ for all $x, y \in \mathbb{R}^n$. Before we continue, we want to introduce the concise notation $s^k := (u^k, q^{k-1})$ for all $k \in \mathbb{N}$, such that $F(s^k) = F(u^k, q^{k-1})$. With the following lemma we prove a sufficient decrease property of the surrogate energy (4.1) for subsequent iterates.

Lemma 4.2 (Sufficient decrease property). *Assume $E \in \Psi_L$ and $R \in \Gamma_0$. Further, suppose that the stepsize τ^k satisfies the condition*

$$0 < \tau^k \leq \frac{2}{L + 2\rho_1}\,, \tag{4.2}$$

for some $\rho_1 > 0$ and all $k \in \mathbb{N}$. Then the iterates of Algorithm 3.2 satisfy the descent estimate

$$F(s^{k+1}) + \rho_1 \|u^{k+1} - u^k\|^2 \leq F(s^k)\,, \tag{4.3}$$

for $s^k := (u^k, q^{k-1})$ and F as defined in (4.1). In addition, we observe

$$\lim_{k\to\infty} \|u^{k+1} - u^k\|^2 = 0 \quad \text{as well as} \quad \lim_{k\to\infty} D_R^{symm}(u^{k+1}, u^k) = 0\,. \tag{4.4}$$

Proof. First of all, we compute

$$\tau^k \left(\nabla E(u^k) + q^{k+1} - q^k\right) + u^{k+1} - u^k = 0$$

as the optimality condition of (3.4a), which is also the rearranged update formula (3.4b) as mentioned earlier (for $q^{k+1} \in \partial R(u^{k+1})$). Taking the inner product with $u^{k+1} - u^k$ therefore yields

$$-\langle \nabla E(u^k), u^{k+1} - u^k\rangle = \frac{1}{\tau^k}\|u^{k+1} - u^k\|^2 + D_R^{\mathrm{symm}}(u^{k+1}, u^k)\,. \tag{4.5}$$

Due to the Lipschitz-continuity of the gradient of E we can use (A.4) from the appendix and further estimate

$$E(u^{k+1}) \leq E(u^k) + \langle \nabla E(u^k), u^{k+1} - u^k\rangle + \frac{L}{2}\|u^{k+1} - u^k\|^2\,.$$

Together with (4.5) and the stepsize bound (4.2) we therefore obtain the estimate

$$E(u^{k+1}) + D_R^{\mathrm{symm}}(u^{k+1}, u^k) + \rho_1\|u^{k+1} - u^k\|^2 \leq E(u^k)\,. \tag{4.6}$$

Adding $D_R^{q^{k-1}}(u^k, u^{k-1})$ to both sides of the inequality then allows us to conclude

$$\begin{aligned} &F(s^{k+1}) + D_R^{q^{k+1}}(u^k, u^{k+1}) + D_R^{q^{k-1}}(u^k, u^{k-1}) + \rho_1\|u^{k+1} - u^k\|^2 \\ &\leq F(s^k)\,. \end{aligned}$$

Due to the non-negativity of $D_R^{q^{k+1}}(u^k, u^{k+1})$ and $D_R^{q^{k-1}}(u^k, u^{k-1})$, we have verified (4.3). Moreover, summing up (4.6) over $k = 0, \ldots, N$ yields

$$\begin{aligned} \sum_{k=0}^{N} \left[\rho_1\|u^{k+1} - u^k\|^2 + D_R^{\mathrm{symm}}(u^{k+1}, u^k)\right] &\leq \sum_{k=0}^{N} E(u^k) - E(u^{k+1})\,, \\ &= E(u^0) - E(u^{N+1})\,, \\ &\leq E(u^0) - \inf_u E(u) < \infty\,. \end{aligned}$$

Taking the limit $N \to \infty$ therefore implies

$$\sum_{k=0}^{\infty} \left[\rho_1\|u^{k+1} - u^k\|^2 + D_R^{\mathrm{symm}}(u^{k+1}, u^k)\right] < \infty\,,$$

and thus (4.4), due to $\rho_1 > 0$. ∎

Remark 2. *As Lemma 4.2 implies the monotonic decrease $F(s^{k+1}) \leq F(s^k)$, we already know that the sequence $\{F(s^k)\}_{k\in\mathbb{N}}$ is bounded from above. It is also bounded from below, since $F(s^k) \geq E(u^k) \geq \inf_u E(u) > -\infty$, due to $E \in \Psi_L$.*

It is worth mentioning that the name sufficient decrease can be misleading in the context of Algorithm 3.2 as it is not unusual for specific choices of R that the function value of E does not change for several iterations.

Our next result is a bound for the subgradients of the surrogate energy at the iterates computed with Algorithm 3.2. Note that the subdifferential of the surrogate objective reads as

$$\partial F(x,y) = \left\{ \begin{pmatrix} \nabla E(x) + z_1 - y \\ z_2 - x \end{pmatrix} \,\middle|\, z_1 \in \partial R(x), z_2 \in \partial R^*(y) \right\},$$

which can for example be deduced from [74]. With $q^{k+1} \in \partial R(u^{k+1})$, and the fact that $q^k \in \partial R(u^k)$ is equivalent to $u^k \in \partial R^*(q^k)$ (Lemma A.4 in the appendix), we know that

$$r^{k+1} := \begin{pmatrix} \nabla E(u^{k+1}) + q^{k+1} - q^k \\ u^k - u^{k+1} \end{pmatrix} \in \partial F(u^{k+1}, q^k) = \partial F(s^{k+1}). \tag{4.7}$$

Subsequently, we want to show that the norm of this sequence of subgradients $\{r^k\}_{k\in\mathbb{N}}$ is bounded by the iterates gap of the primal variable.

Lemma 4.3 (A subgradient lower bound for the iterates gap). *Let the same assumptions hold true as in Lemma 4.2 and $\tau^k \geq \tau^{min} := \inf_k \tau^k > 0$. Then the iterates of Algorithm (3.2) satisfy*

$$\|r^k\| \leq \rho_2 \|u^k - u^{k-1}\|, \tag{4.8}$$

for $r^k \in \partial F(s^k)$ as defined in (4.7), $s^k := (u^k, q^{k-1})$, $\rho_2 := \left(1 + L + 1/\tau^{min}\right)$ and $k \in \mathbb{N}$.

Proof. From (4.7) we know

$$\|r^k\| \leq \|\nabla E(u^k) + q^k - q^{k-1}\| + \|u^k - u^{k-1}\|.$$

Together with (3.4b) we therefore estimate

$$\begin{aligned} \|r^k\| &\leq \left\|\nabla E(u^k) + q^k - q^{k-1}\right\| + \|u^k - u^{k-1}\| \\ &= \left\|\nabla E(u^k) - \nabla E(u^{k-1}) + \frac{1}{\tau^{k-1}}\left(u^{k-1} - u^k\right)\right\| + \|u^k - u^{k-1}\|, \\ &\leq \left(1 + L + \frac{1}{\tau^{\min}}\right)\|u^k - u^{k-1}\| = \rho_2 \|u^k - u^{k-1}\|, \end{aligned}$$

where we have made use of the Lipschitz-continuity of the gradient of E. ∎

Remark 3. *We want to point out that the Lipschitz-continuity of ∇E is not necessary if $R \equiv 0$. In that case it is easy to see that we can obtain the estimate*

$$\|\nabla E(u^k)\| \leq \frac{1}{\tau^{min}} \|u^{k+1} - u^k\|$$

instead of (4.8) *(see also [10]), without the use of Lipschitz-continuity. For the sufficient decrease Theorem 4.2 it is already enough to choose* τ^k *such that* $G := \frac{1}{2}\|\cdot\|^2 - \tau^k E$ *is convex for all arguments and all* $k \in \mathbb{N}$*. This observation has already been made and exploited in [6, 10, 17]. We also want to emphasise that the requirement of Lipschitz continuity can potentially be relaxed if backtracking strategies are incorporated into Algorithm 3.2.*

To conclude our convergence analysis we prove global convergence of Algorithm 3.2 with the help of the Kurdyka-Łojasiewicz (KL) property as defined in the appendix in Definition A.11. In order to apply the KL property, we have to verify some properties of the set of limit points. Let $\{s^k\}_{k\in\mathbb{N}} = \{(u^k, q^{k-1})\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 3.2 from starting points u^0 and q^0 with $q^0 \in \partial R(u^0)$. The set of limit points is defined as

$$\omega(s^0) := \left\{ \overline{s} = (\overline{u}, \overline{q}) \in \mathbb{R}^n \times \mathbb{R}^n \;\middle|\; \text{there exists an increasing sequence of integers } \{k_j\}_{j\in\mathbb{N}} \text{ such that } \lim_{j\to\infty} u^{k_j} = \overline{u} \text{ and } \lim_{j\to\infty} q^{k_j} = \overline{q} \right\}.$$

Before we continue, we want to emphasise that the current assumptions on E and R are not sufficient in order to guarantee convergence of the dual variable, which we want to demonstrate with a simple counter example.

Remark 4. *Let* $E(u) = (u+1)^2/2$*, and* $R(u) = \chi_{\geq 0}(u)$ *with*

$$\chi_{\geq 0}(u) := \begin{cases} 0 & u \geq 0 \\ \infty & u < 0 \end{cases}.$$

It is obvious that $E \in \Psi_1$ *and that the only critical point of* E *is* $\hat{u} = -1$*. However, Algorithm 3.2 can never converge to that point but will converge to* $\overline{u} = 0$ *due to the choice of* R*. This can be seen for instance for the choices* $u^0 > 0$*,* $q^0 = 0$ *and* $\tau^k = 1$*. Then the subsequent iterates are* $u^k = 0$ *and* $q^k = u^0 - k$*, thus,* $u^k \to 0$ *and* $q^k \to -\infty$*.*

For convex, quadratic fidelity terms (such as E in the example above) it is sufficient to satisfy a source condition of the form $\partial R(\hat{u}) \neq \emptyset$ (which in Remark 4 is clearly violated) in order to guarantee boundedness of the subgradients, see for instance [38]. For general, non-convex terms E it is not straight forward to adapt the concept of source conditions, which is why we are going to assume local boundedness of the subgradients instead.

Definition 4.4 (Locally bounded subgradients). *We say that* R *has* locally bounded subgradients *if for every compact set* $U \subset \mathbb{R}^n$ *there exists a constant* $C \in (0, \infty)$ *such that for all* $v \in U$ *and all* $q \in \partial R(v)$ *we have* $\|q\| \leq C$*.*

Boundedness is not a very restrictive requirement as it is for instance satisfied for the large class of Lipschitz-continuous functions.

Proposition 4.5. *Let* $R \in \Gamma_0$ *be a (globally) Lipschitz continuous function in the sense of Definition A.7 in the appendix. Then* R *has locally bounded subgradients.*

Proof. From the convexity of R we observe

$$\langle q, h\rangle \leq |R(v+h) - R(v)| \leq L\|h\|\,,$$

for any $U \subset \mathbb{R}^n$ and any $h, v \in U$ with $v + h \in U$ and $q \in \partial R(v)$. Taking the supremum over h with $\|h\| \leq 1$ shows $\|q\| \leq L$, which proves the assertion. ∎

Remark 5. *Note that every continuously differentiable function is already locally Lipschitz-continuous, and therefore has locally bounded gradients according to Proposition 4.5.*

Before we show global convergence of Algorithm 3.2 to a critical point of E, we need to verify that the surrogate function converges to E on $\omega(s^0)$, that $\omega(s^0)$ is a non-empty, compact and connected set and that its primal limiting points form a subset of the set of critical points of E. The following lemma guarantees that for a sequence converging to a limit point we also know that the surrogate objective converges to the objective evaluated at this limit point.

Lemma 4.6. *Suppose $E \in \Psi_L$, $R \in \Gamma_0$, and let $\overline{s} \in \omega(s^0)$. Then we already know*

$$\lim_{k\to\infty} F(s^k) = F(\overline{s}) = E(\overline{u})\,. \tag{4.9}$$

Proof. Since $\overline{s}$ is a limit point of $\{s^k\}_{k\in\mathbb{N}}$ we know that there exists a subsequence $\{s^{k_j}\}_{j\in\mathbb{N}}$ with $\lim_{j\to\infty} s^{k_j} = \overline{s}$. Hence, we immediately obtain

$$\lim_{j\to\infty} F(s^{k_j}) = \lim_{j\to\infty} \left\{ E(u^{k_j}) + D_R^{q^{k_j-1}}(u^{k_j}, u^{k_j-1}) \right\} = E(\overline{u})\,,$$

due to the continuity of E and $\lim_{j\to\infty} D_R^{q^{k_j-1}}(u^{k_j}, u^{k_j-1}) = 0$ as a result of Lemma 4.2. Since $\{F(s^k)\}_{k\in\mathbb{N}}$ is also monotonically decreasing and bounded from below according to Remark 2, we can further conclude (4.9) as a consequence of the monotone convergence theorem. ∎

In addition to Lemma 4.6, the following lemma states that $\omega(s^0)$ is a non-empty, compact and connected set, and that the objective F is constant on that set.

Lemma 4.7 ([16, Lemma 5]). *Suppose $E \in \Psi_L$ and that $R \in \Gamma_0$ has locally bounded subgradients. Then the set $\omega(s^0)$ is a non-empty, compact and connected set, the surrogate objective F is constant on $\omega(s^0)$ and we have $\lim_{k\to\infty} \operatorname{dist}(s^k, \omega(s^0)) = 0$.*

We can further verify that the set of primal limiting points is a subset of the set of critical points of the energy E.

Lemma 4.8. *Suppose $E \in \Psi_L$, and that $R \in \Gamma_0$ has locally bounded subgradients. Then we have $\overline{u} \in \operatorname{crit}(E)$ for every $\overline{s} = (\overline{u}, \overline{q}) \in \omega(s^0)$.*

Proof. We prove this assertion by contradiction to the boundedness of the subgradients. Let $\overline{s} := (\overline{u}, \overline{q}) \in \omega(s^0)$, which means $\lim_{k\to\infty} u^k = \overline{u}$. Assume that $\nabla E(\overline{u}) \neq 0$ and let $c := \|\nabla E(\overline{u})\| > 0$. It follows from the subgradient update (3.5) and the reverse triangle inequality $\|a + \sum_i a_i\| \geq \|a\| - \sum_i \|a_i\|$ that

$$\|q^k\| \geq \left\| \sum_{n=0}^{k-1} \nabla E(\overline{u}) \right\| - \|q^0\| - \sum_{n=0}^{k-1} \left[\frac{1}{\tau^n}\|u^{n+1} - u^n\| + \|\nabla E(u^n) - \nabla E(\overline{u})\| \right]\,.$$

As $u^k \to \overline{u}$, there exists $K \in \mathbb{N}$ such that for all $n \geq K$ the bounds $\|u^n - \overline{u}\| \leq c\tau^{\min}/8$ and $\|\nabla E(u^n) - \nabla E(\overline{u})\| \leq c/4$ hold. Thus, we have for all $n \geq K$ that

$$1/\tau^n \|u^{n+1} - u^n\| + \|\nabla E(u^n) - \nabla E(\overline{u})\| \leq c/2,$$

and therefore

$$\sum_{n=0}^{k-1} \left[\frac{1}{\tau^n} \|u^{n+1} - u^n\| + \|\nabla E(u^n) - \nabla E(\overline{u})\| \right]$$
$$\leq \sum_{n=K}^{k-1} \left[\frac{1}{\tau^n} \|u^{n+1} - u^n\| + \|\nabla E(u^n) - \nabla E(\overline{u})\| \right] + \text{const} \leq kc/2 + \text{const},$$

for all $k \in \mathbb{N}$, with a constant independent of k. Combining these two estimates yields

$$\|q^k\| \geq \left\| \sum_{n=0}^{k-1} \nabla E(\overline{u}) \right\| - kc/2 + \text{const} = kc/2 + \text{const}\,.$$

Hence, we observe $\lim_{k\to\infty} \|q^k\| = \infty$, which is a contradiction to the boundedness of $\{q^k\}$. Thus, $\nabla E(\overline{u}) = 0$, which means $\overline{u} \in \operatorname{crit}(E)$. ∎

Now we have all the necessary ingredients to show the following global convergence result for Algorithm 3.2.

Theorem 4.9 (Finite length property). *Suppose that F is a KL function in the sense of Definition A.11. Further, assume $R \in \Gamma_0$ with locally bounded subgradients. Let $\{s^k\}_{k\in\mathbb{N}} = \{(u^k, q^{k-1})\}_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 3.2. Then the sequence $\{u^k\}_{k\in\mathbb{N}}$ has finite length, i.e.*

$$\sum_{k=0}^{\infty} \|u^{k+1} - u^k\| < \infty\,. \tag{4.10}$$

Proof. We follow the steps of the proof of [16, Theorem 1] but with non-trivial modifications.

The sequence $\{u^k\}_{k\in\mathbb{N}}$ is bounded, which follows from the assumption $E \in \Psi_L$ and the monotonic decrease. Thus, we know that there exists a convergent subsequence $\{u^{k_j}\}_{j\in\mathbb{N}}$ and $\overline{u} \in \mathbb{R}^n$ with

$$\lim_{j\to\infty} u^{k_j} = \overline{u}\,.$$

As a consequence of Lemma 4.6 we further know that $\lim_{k\to\infty} F(s^k) = F(\overline{s}) = E(\overline{u})$. If there exists an index $l \in \mathbb{N}$ with $F(s^l) = E(\overline{u})$ the results follow trivially. If there does not exist such an index, we observe that for any $\eta > 0$ there exists an index k_1 such that

$$E(\overline{u}) < F(s^k) < E(\overline{u}) + \eta$$

for all $k > k_1$. In addition, for any $\varepsilon > 0$ there exists an index k_2 with

$$\operatorname{dist}(s^k, \omega(s^0)) < \varepsilon$$

for all $k > k_2$, due to Lemma 4.7. Hence, if we choose $l := \max(k_1, k_2)$, we know that u^k is in the set (A.5) for all $k > l$ according to Lemma A.12 in the appendix.

By Lemma 4.7, $\omega(u^0)$ satisfies all the assumptions of Lemma A.12 and we have

$$1 \leq \varphi'(F(s^k) - E(\overline{u}))\ \operatorname{dist}(0, \partial F(s^k)) \tag{4.11}$$

for all $k > l$. This inequality makes sense due to $F(s^k) > E(\overline{u})$ for all k.
From the concavity of φ we know that

$$\varphi'(x) \leq \frac{\varphi(x) - \varphi(y)}{x - y}$$

holds for all $x, y \in [0, \eta), x > y$, which we will use for the specific choices of $x = F(w^k) - E(\overline{u})$ and $y = F(s^{k+1}) - E(\overline{u})$. Combining the latter with Lemma 4.2 and abbreviating

$$\varphi^k := \varphi(F(s^k) - E(\overline{u}))$$

yields

$$\varphi'(F(s^k) - E(\overline{u})) \leq \frac{\varphi^k - \varphi^{k+1}}{F(s^k) - F(s^{k+1})} \leq \frac{\varphi^k - \varphi^{k+1}}{\rho_1 \|u^{k+1} - u^k\|^2}\,. \tag{4.12}$$

Inserting (4.12) and the subgradient bound (4.8) into the KL inequality (4.11) leads to

$$\|u^{k+1} - u^k\|^2 \leq \frac{\rho_2}{\rho_1}(\varphi^k - \varphi^{k+1})\|u^k - u^{k-1}\|\,.$$

Taking the square root, multiplying by 2 and using Young's inequality of the form $2\sqrt{ab} \leq a + b$ then yields

$$2\|u^{k+1} - u^k\| \leq \frac{\rho_2}{\rho_1}(\varphi^k - \varphi^{k+1}) + \|u^k - u^{k-1}\|\,.$$

Subtracting $\|u^{k+1} - u^k\|$ and summing from $k = l, \ldots, N$ leads to

$$\begin{aligned}\sum_{k=l}^{N} \|u^{k+1} - u^k\| &\leq \frac{\rho_2}{\rho_1}(\varphi^l - \varphi^{N+1}) + \|u^l - u^{l-1}\| - \|u^{N+1} - u^N\| \\ &\leq \frac{\rho_2}{\rho_1}\varphi^l + \|u^l - u^{l-1}\| < \infty\,,\end{aligned}$$

and hence, we obtain the finite length property by taking the limit $N \to \infty$. ∎

Corollary 4.10 (Convergence). *Under the same assumptions as Theorem 4.9, the sequence $\{u^k\}_{k\in\mathbb{N}}$ converges to a critical point of E.*

Proof. As in the proof of [16, Theorem 1 (ii)], the finite length property Theorem 4.9 implies $\sum_{k=l}^{N} \|u^{k+1} - u^k\| \to 0$ for $N \to \infty$. Thus, for any $s \geq r \geq l$ we have

$$\|u^s - u^r\| = \left\| \sum_{k=r}^{s-1} u^{k+1} - u^k \right\| \leq \sum_{k=r}^{s-1} \|u^{k+1} - u^k\| \leq \sum_{k=l}^{\infty} \|u^{k+1} - u^k\|\,.$$

This shows that $\{u^k\}_{k\in\mathbb{N}}$ is a Cauchy sequence and, thus, is convergent. According to Lemma 4.8 its limit is a critical point of E. ∎

4.1. Global convergence in the absence of locally bounded subgradients. In the previous section we have made the assumption that the subgradients of R have to be locally bounded in order to guarantee convergence of the primal iterates to a critical point of E. In Remark 4 we have seen an example for which the subgradients of R diverge, but the primal iterates still converge, just not to a critical point of E. This leaves us with two open questions: 1) could we prove convergence of the primal iterates without boundedness of the dual iterates and 2) would the limit (if it exists) be a critical point of some other energy? It might be possible to answer the first question by slightly modifying Definition A.11 and Lemma A.12 in the appendix, as well as Lemma 4.7 to accommodate the fact that the surrogate function is also constant on the set of limiting points that only depends on the primal variable (which we denote by $\omega(u^0)$ for convenience). A potential modification of (A.5) in Lemma 4.7 could for instance be

$$\{u, q \in \mathbb{R}^n \mid \operatorname{dist}(u, \omega(u^0)) < \varepsilon\} \cap \{u, q \in \mathbb{R}^n \mid E(\overline{u}) \leq F(u, q) \leq E(\overline{u}) + \eta\},$$

where $\overline{u} \in \omega(u^0)$. Note that this modification would not affect the finite length proof of Theorem 4.9 and therefore would still imply global convergence, but not necessarily to a critical point of E. Remark 4 leaves room for speculation whether an answer to the second question is that the primal iterates converge to a critical point of $E + \chi_{\operatorname{dom}(R)}$, where $\chi_{\operatorname{dom}(R)}$ denotes the characteristic function over the effective domain of R. Proving this, however, is beyond the scope of this paper.

4.2. Limitations of the convergence analysis and possible remedies. The convergence analysis presented in this paper relies on the fact that the function E satisfies $E \in \mathcal{S}_L$, which is often restrictive for practical applications. Even simple functions such as the blind deconvolution data fidelity term from Section 2 are not globally L-smooth. Remedies are the use of an alternating version of Algorithm 3.2 in the spirit of [16] and to make use of local smoothness of the functions with fixed variables. For two variables u_1 and u_2, such a scheme is of the form

$$\begin{aligned}
u_1^{k+1} &= \operatorname*{arg\,min}_{u_1 \in \mathbb{R}^n} \left\{ \tau_1^k \langle \nabla_1 E(u_1^k, u_2^k), u_1 - u_1^k \rangle + D_{J_1}^{p_1^k}(u_1, u_1^k) \right\} \\
p_1^{k+1} &= p_1^k - \tau_1^k \nabla_1 E(u_1^k, u_2^k), \\
u_2^{k+1} &= \operatorname*{arg\,min}_{u_2 \in \mathbb{R}^n} \left\{ \tau_2^k \langle \nabla_2 E(u_1^{k+1}, u_2^k), u_2 - u_2^k \rangle + D_{J_2}^{p_2^k}(u_2, u_2^k) \right\} \\
p_2^{k+1} &= p_2^k - \tau_2^k \nabla_2 E(u_1^{k+1}, u_2^k),
\end{aligned}$$

assuming a separable structure of $J(u_1, u_2) = J_1(u_1) + J_2(u_2)$. Here ∇_1 and ∇_2 refer to the partial gradients of E with respect to u_1 and u_2, and $p_1^k \in \partial J_1(u_1^k)$ and $p_2^k \in \partial J_2(u_2^k)$ are subgradients in the subdifferential of J_1 and J_2, respectively. The analysis of such a scheme should be relatively straight-forward, but is beyond the scope of this work.

Another limitation in terms of convergence analysis that becomes obvious from the motivating example in Section 2 is the use of characteristic functions. If we incorporate them in the function R, we run into the issues outlined in Section 4.1. If we add them to the objective function E, we lose the continuity and differentiability. A remedy for the blind deconvolution example (and many similar examples) in Section 2 is that for the convolution kernel the

additional Bregman function R is simply zero, so that the algorithm merely has to perform a proximal point step in the direction of the convolution kernel. The convergence analysis in such a setting is straight-forward, but we did not include it in order not to complicate notation. Alternatively, one could replace the characteristic function with its Moreau–Yosida envelope.

This concludes the theoretical analysis of Algorithm 3.2. In the following two sections we are going to discuss three applications, their mathematical modelling in the context of Algorithm 3.2 and their numerical results.

5. Applications.

We demonstrate the capabilities of the linearised Bregman iteration by using it to approximately minimise several non-convex minimisation problems. We say approximately, as we do not exactly minimise the corresponding objective functions, but rather compute iteratively regularised solutions to the associated inverse problems via early stopping of the iteration.

5.1. Parallel Magnetic Resonance Imaging.

In (standard) Magnetic Resonance Imaging (MRI) the goal is to recover the spin-proton density from sub-sampled Fourier measurements that were obtained with a single radio-frequency (RF) coil. In parallel MRI, multiple RF coils are used for taking measurements, thus allowing to recover the spin-proton density from more measurements compared to the standard case. This, however, comes at the cost of having to model the sensitivities of the individual RF coils w.r.t. the measured material. We basically follow the mathematical modelling of [70, 77] and describe the recovery of the spin-proton density and the RF coil sensitivities as the minimisation of the following energy function:

$$E(u, b_1, \ldots, b_s) := \frac{1}{2}\sum_{j=1}^{s} \|\mathcal{S}(\mathcal{F}((K(u, b_1, \ldots, b_s))_j)) - f_j\|_2^2 + \frac{\epsilon}{2}\left(\|u\|^2 + \sum_{j=1}^{s}\|b_j\|^2\right). \tag{5.1}$$

Here $\mathcal{F} \in \mathbb{C}^{n\times n}$ is the (discrete) Fourier transform, $\mathcal{S} \in \{0,1\}^{m\times n}$ is a sub-sampling operator, K is the non-linear operator $K(u, b_1, \ldots, b_s) = (ub_1, ub_2, \ldots, ub_s)^T$, u denotes the spin-proton density, $b_1, b_2, \ldots, b_s$ the s coil sensitivities, $f_1, \ldots, f_s$ the corresponding sub-sampled k-space data and $\epsilon > 0$ is a scalar parameter that ensures bounded level-sets of E. Since $\mathbb{C}$ has the same topology as $\mathbb{R} \times \mathbb{R}$, we can formally treat all variables as variables in $\mathbb{R}^{2n}$. Note that E as defined in (5.1) is not globally L-smooth, which is why we also assume that we choose parameters and initial values such that our sequence $\{u^k\}_{k\in\mathbb{N}}$ of primal variables generated by Algorithm 3.2 satisfies

$$\|\nabla E(u^k) - \nabla E(u^{k-1})\|_2 \leq L^k \|u^k - u^{k-1}\|_2 \,,$$

for a sequence $\{L^k\}_{k\in\mathbb{N}}$ of positive constants. Hence, $E \in \Psi_{L^k}$, which means that E is (locally) L^k-smooth, respectively ∇E is (locally) L^k-Lipschitz-continuous in the sense of Definition A.7. Furthermore, we assume that the sequence $\{L^k\}_{k\in\mathbb{N}}$ is bounded from above, i.e. $L^k \leq L$ for all $k \in \mathbb{N}$, and consequently $E \in \Psi_L$. It is not necessarily straight-forward to prove existence of L a-priori, but it is relatively easy to validate it a-posteriori. Note that, alternatively, one could use an alternating version of Algorithm 3.2 as discussed in Section 4.2.

The inverse problem of parallel MRI has been subject in numerous research publications [71, 48, 12]. We follow a different methodology here and apply Algorithm 3.2 to approximately minimise (5.1) with the following configuration. We choose the function R to be of the form

$$R(u, b_1, \dots, b_s) = R_1(u) + \sum_{j=1}^{s} R_2(b_j)\,,$$

with

$$R_1(u) = \alpha_0 \mathrm{TV}(u) = \alpha_0 \| |\nabla u| \|_1$$

and

$$R_2(b_j) = \alpha_j \sum_{l=1}^{n} w_l \, |(C\, b_j)_l|\,, \qquad \forall j \in \{1, \dots, s\}\,.$$

Here ∇ denotes a discrete finite forward difference approximation of the gradient, $|\cdot|$ is the Euclidean vector norm, C denotes the discrete two-dimensional cosine transform, $\{w_l\}_{l\in\{1,\dots,n\}}$ is a set of weighting-coefficients and $\alpha_0, \dots, \alpha_{s+1}$ are positive scaling parameters. Note that all functions are chosen to be semi-algebraic, and semi-algebraic functions and their additive compositions are KL functions (see [2, 3, 4]). Iterating Algorithm 3.2 for too long may lead to unstable minimisers of (5.1) in case the k-space data $f_1, \dots, f_s$ are noisy, which is why we are going to apply Morozov's discrepancy principle [60] as a stopping criterion to stop the iteration early (see also [65, 40, 56], and [75, 5, 45] in the context of nonlinear inverse problems), i.e. we stop the iteration as soon as

$$E(u, b_1, \dots, b_s) \leq \eta \tag{5.2}$$

is satisfied, for some $\eta > 0$. Usually η depends on the variance of the normal-distributed noise.

5.2. Blind deconvolution. Blind deconvolution is extensively discussed in the literature, e.g. [49, 30, 26] and the references therein, with several approaches for which the convergence proofs also rely on the KL inequality [15, 72, 32]. We follow the same setting as in Section 2 (with additional regularisation as in (5.1) in order to guarantee bounded level-sets) and make the assumptions that the blur-free image u has low total variation and that the kernel h satisfies a simplex constraint, i.e. all entries are non-negative and sum up to one. The assumption of low total variation can for instance be motivated by [31], but as as we have seen in Section 2, minimising E with some additional total variation regularisation does often not lead to visually satisfactory results. We therefore apply Algorithm 3.2 with $R : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$ defined as

$$R(u, h) = \alpha \mathrm{TV}(u)\,,$$

for $\alpha \geq 0$. All functions are semi-algebraic, and we make the same local smoothness assumption as in Section 5.1. In case of noisy data, we will proceed as in Section 5.1 and stop the iteration via the discrepancy principle.

5.3. Classification. The last application that we want to discuss is the classification of images. Given a set $D \in \mathbb{R}^{s\times r}$ of r training images (with s pixel each) in column vector form, we want to train a neural network to classify those images. We do so by learning the parameters $(A_1, \ldots, A_l)$ of the l-layer neural network

$$\rho(x) := \rho_1(A_1\rho_2(A_2 \ldots \rho_l(A_l x)) \ldots)$$

in a supervised fashion. Here the parameters $A_j \in \mathbb{R}^{m_j\times n_j}$ are matrices of different size, and the functions $\{\rho_j\}_{j=1}^l$ are so-called activity functions of the neural net. Typical choices for activity functions are max- and min-functions, also known as rectifier. However, due to their non-differentiability it is common to approximate them with either the pointwise smooth-max-function, i.e.

$$\rho_j(x, c, \beta) := \frac{x\exp(\beta x) + c\exp(\beta c)}{\exp(\beta x) + \exp(\beta c)},$$

for $x \in \mathbb{R}$ and constants $\beta, \in \mathbb{R}$, or the soft-max-function, i.e.

$$\rho_j(x)_i = \frac{\exp(x_i)}{\sum_{l=1}^m \exp(x_l)},$$

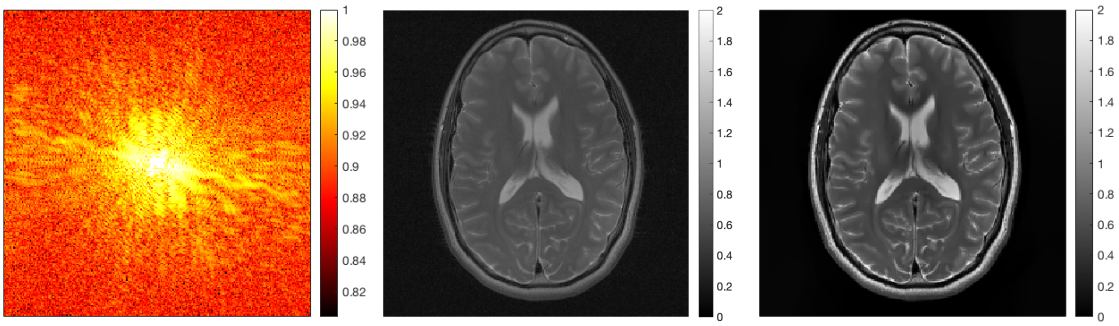
for $x \in \mathbb{R}^m$. The latter has the advantage that the function output automatically satisfies the simplex constraint.

Note that if each function $\rho_j(A_j x)$ is chosen to be semi-algebraic, the composition ρ is also semi-algebraic, see [1, Proposition 2.2.10]. If we choose $\rho_j(y) := \min(1, \max(0, y))$ for all $j \in \{1, \ldots, l\}$ for instance, we can then show that also ρ is semi-algebraic.
Defining a nonlinear operator $K(A_1, A_2, \ldots, A_l) := \rho_1(A_1\rho_2(A_2 \ldots \rho_l(A_l D)) \ldots)$ for a given matrix D and a given label matrix $Y \in \mathbb{R}^{m_1\times r}$, we aim to minimise

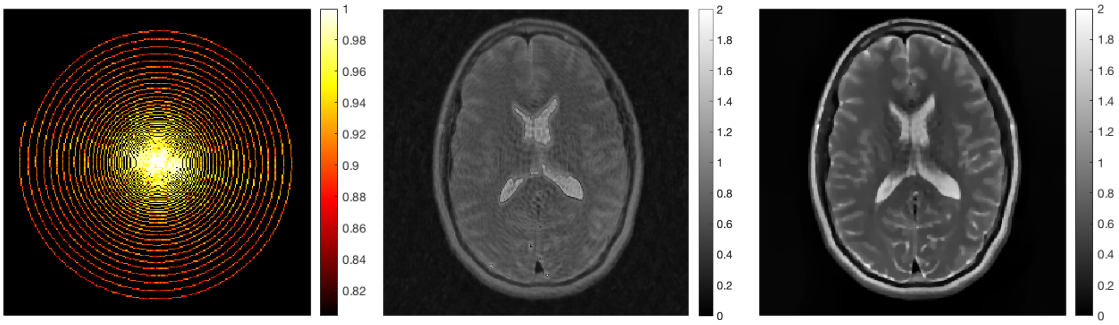
$$E(A_1, A_2, \ldots, A_l) := \mathcal{D}(K(A_1, A_2, \ldots, A_l), Y) + \frac{\epsilon}{2}\sum_{j=1}^{l} \|A_j\|_{\text{Fro}}^2, \tag{5.3}$$

where $\mathcal{D} : \mathbb{R}^{m_1\times r} \times \mathbb{R}^{m_1\times r} \to \mathbb{R}$ denotes a function that measures the distance between its arguments in some sense. Our choice for $\mathcal{D}$ is simply the squared Frobenius norm $\mathcal{D}(X, Y) = \frac{1}{2}\|X - Y\|_{\text{Fro}}^2$ but other choices are possible. As mentioned earlier, the whole objective E can be made a KL function, if for instance $\mathcal{D}$ and ρ are chosen to be semi-algebraic, as their composition will also be semi-algebraic.

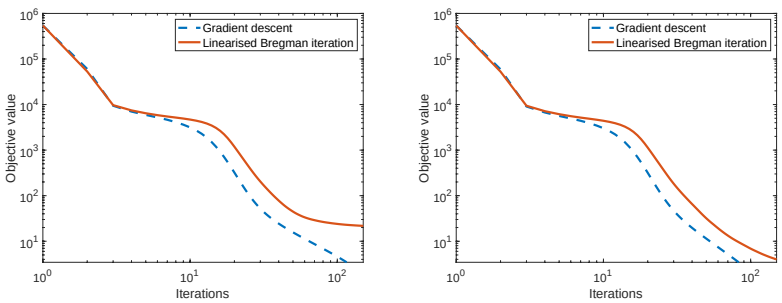
As in the previous sections, we aim to minimise (5.3) with Algorithm 3.2 and make the same local smoothness assumption as before. This time we choose $R(A_1, \ldots, A_l) = \sum_{j=1}^l \alpha_j\|A_j\|_*$. Here $\{\alpha_j\}_j^l$ is a set of positive scaling parameters, and $\|X\|_* := \sum_{i=1}^{\text{rank}(X)} \sigma_i$ is the one norm of the singular values $\{\sigma_i\}_{i=1}^{\text{rank}(X)}$ of the argument X, also known as the nuclear norm. The rationale behind this choice for R is that we can create iterates where the ranks of the individual matrices are steadily increasing. This way we control the number of effective parameters and do not fit all parameters right from the start.



(a) Fully sampled (b) Recon. from 5.1a (c) Recon. from 5.1a

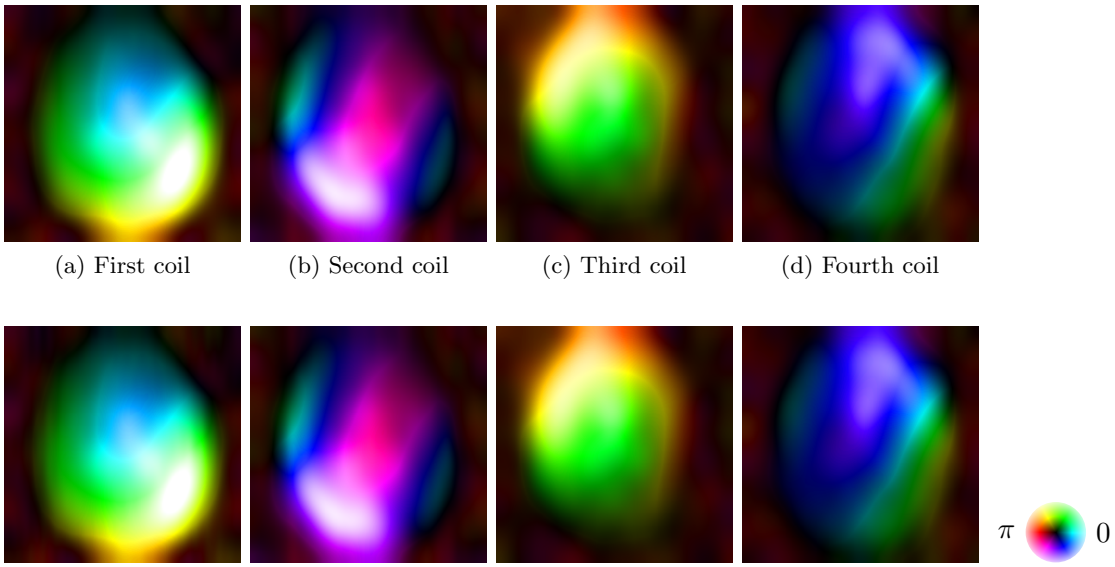


(d) Subsampled (e) Recon. from 5.1d (f) Recon. from 5.1d



(g) Convergence plot from 5.1a (h) Convergence plot from 5.1d

Figure 5.1. *Figure 5.1a shows a log-plot of the modulus of the fully sampled k-space data of the first coil taken from [48]. Figure 5.1b shows the reconstruction of the spin proton density from the data visualised in Figure 5.1a via gradient descent, whereas Figure 5.1c shows the reconstruction of the spin proton density from the same data but via Algorithm 3.2. In Figure 5.1d we see roughly 25 % of the k-space data visualised in Figure 5.1a, sampled on a spiral on a cartesian grid [11]. Figure 5.1e shows the reconstruction of the spin proton density from this subsampled k-space data with gradient descent, while Figure 5.1f shows the reconstruction of the spin proton density from the same data but with Algorithm 3.2. Figure 5.1g and Figure 5.1h are showing the convergence plots in terms of energy decrease per iteration for the reconstructions that are obtained from the k-space data shown in Figure 5.1a, respectively in Figure 5.1d.*



(a) First coil (b) Second coil (c) Third coil (d) Fourth coil

(e) First coil (f) Second coil (g) Third coil (h) Fourth coil

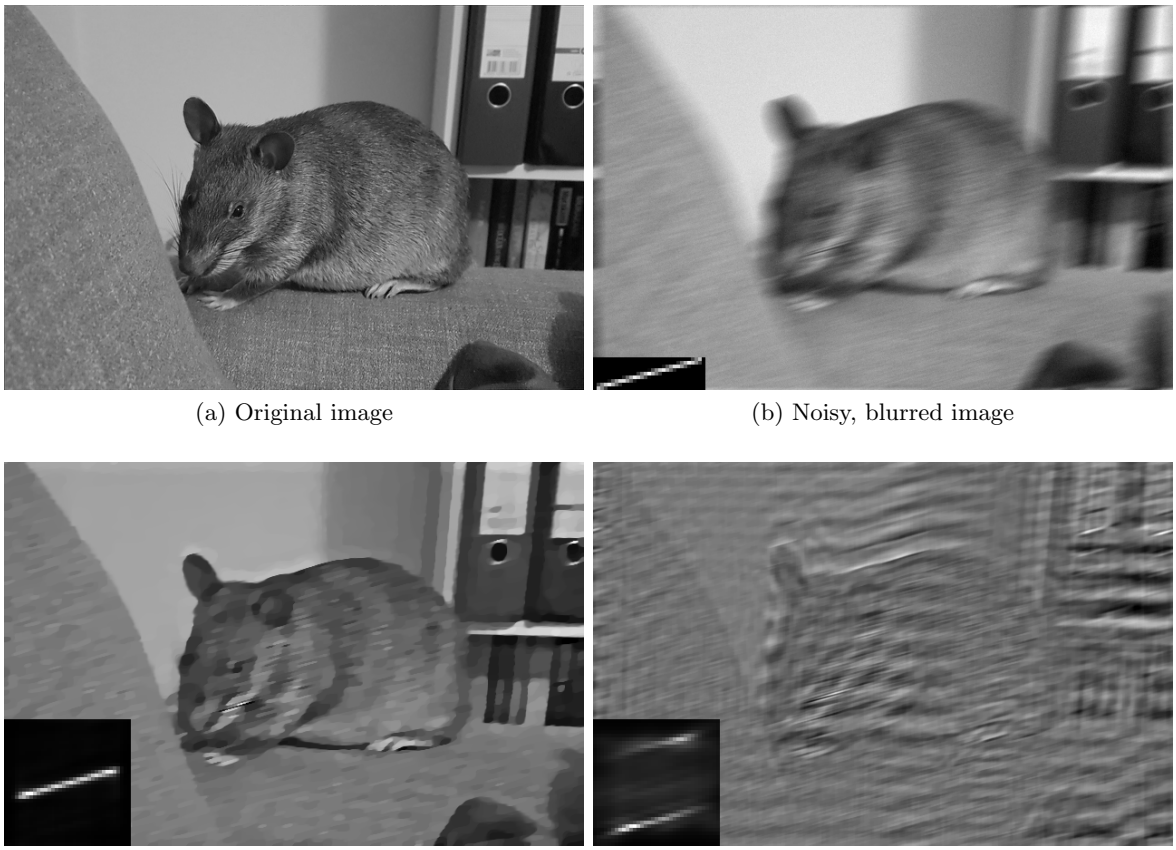
Figure 5.2. *Figure 5.2a - 5.2d show the reconstructions of the coil sensitivities from the fully sampled data. Figure 5.2e - 5.2h show the reconstructions of the same quantities from the sub-sampled data.*

6. Numerical Results. We demonstrate the particular properties and idiosyncrasies of Algorithm 3.2 by computing several numerical solutions to the problems described in Section 5. All results have been computed with MATLAB R2017b. The code for the following examples is available at https://doi.org/10.17863/CAM.16931 and can be used under the Creative Commons Attribution (CC BY) license once the article is accepted for publication.

Notably, all regularisation parameters that ensure boundedness of the level-sets are set to the smallest possible value (ϵ = machine accuracy) in practice. Since we do not use explicit Lipschitz constants, we employ a naïve backtracking strategy for the variable stepsize $\{\tau^k\}_{k\in\mathbb{N}}$. We start with an initial stepsize $\tau^0 > 0$ and check after each iteration whether $E(u^{k+1}) \leq E(u^k) + \varepsilon$ is satisfied. Here, $\varepsilon > 0$ is a small constant that accounts for numerical rounding errors that may cause $E(u^{k+1}) > E(u^k)$ when $E(u^{k+1}) \approx E(u^k)$. If the decrease is satisfied, we set $\tau^{k+1} = \tau^k$; otherwise we set $\tau^{k+1} = (3\tau^k)/4$ and backtrack again until we get a decrease. We want to emphasise that more sophisticated backtracking approaches can be used; we found, however, that the naïve strategy that we use already works well for the computational results shown in the following subsections.

6.1. Parallel MRI. We compute parallel MRI reconstructions from real k-space data. We use data from a T2-weighted TSE scan of a transaxial slice of a brain acquired with a four-channel head-coil in [47]. A reconstruction from fully sampled data is taken as a ground truth. The spiral sub-sampling is simulated by point-wise multiplication of the k-space data with the spiral pattern visualised in Figure 5.1d. We initialise with $u^0 = 2 \times \mathbf{1}^{65536\times 1}$ and $b_j^0 = \mathbf{1}^{65536\times 1}$ for $j \in \{1, \ldots, 4\}$, and compute a $q^0 \in \partial R(u^0)$.

With the parameters $\alpha_j = 1$ for all $j \in \{0, \ldots, s\}$, $\tau^0 = 1/2$, $w_1 = w_2 = w_{\sqrt{n}+1} =$



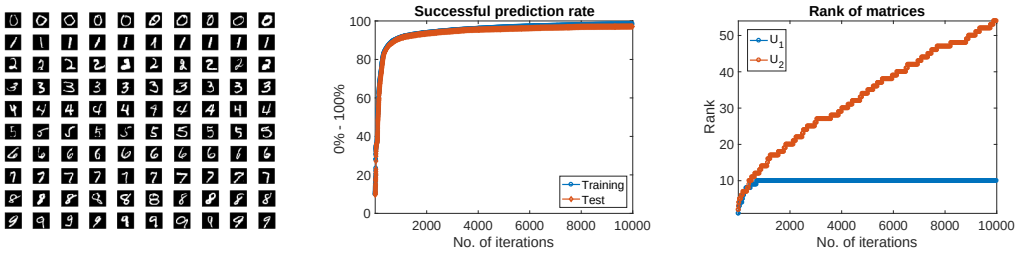
(a) Original image (b) Noisy, blurred image

(c) $\alpha = 0.1$ (d) $\alpha = 10^{-3}$

Figure 6.1. *Figure 6.1a shows an image of Pixel the Gambian pouched rat. Figure 6.1b shows a motion-blurred version of that image, together with some added normal distributed noise. The corresponding convolution kernel is depicted in the bottom left corner. Figure 6.1c visualises the reconstruction of the image and the convolution kernel with Algorithm 3.2 for the choice $\alpha = 10^{-1}$. Figure 6.1d show the reconstructions of the same quantities for the choice $\alpha = 10^{-3}$. We clearly see that a larger choice of α results in a regular solution, whereas a smaller α will mimic traditional gradient descent with almost no additional regularity of the reconstruction.*

$w_{\sqrt{n}+2} = 10^{-6}$ and $w_l = 5$ for $l \in \{1, \dots, n\} \setminus \{1, 2, \sqrt{n}+1, \sqrt{n}+2\}$, and $\eta = 3.45$ we obtain the spin proton density reconstruction visualised in Figure 5.1c, as well as the coil sensitivity reconstructions in Figure 5.2a - 5.2d. In Figure 5.1f and Figure 5.2e - 5.2h we show the results of the reconstructions from sub-sampled data using the sub-sampling scheme in Figure 5.1d.

6.2. Blind deconvolution. To simulate blurring of a gray-scale image $f_{\text{orig}} \in \mathbb{R}^{424\times640}$ we subtract its mean, normalise it and subsequently blur f_{orig} with a motion-blur filter $h \in \mathbb{R}^{9\times31}$. The filter was obtained with the MATLAB©-command `fspecial('motion', 30, 15)`, and we assume periodic boundary conditions for the blurring process. Subsequently we add normally distributed noise with mean zero and standard deviation $\sigma = 10^{-4}$ to obtain a blurry and noisy image f with ground truth f_{orig}. Both f_{orig} and f, as well as h are visualised in



(a) MNIST [51] (b) Prediction (c) Rank

Figure 6.2. *Figure 6.2a shows ten randomly chosen images of each digit from the MNIST training data [51]. Figure 6.2b shows the successful prediction rate of the classifier throughout the iteration both for the training and the test data. Figure 6.2c shows the rank of the two matrices U_1 and U_2 that are reconstructed. It becomes evident that the rank is monotonically increasing throughout the course of the iteration, allowing the model to fit only a reduced no. of effective parameters at a time.*

Figure 6.1.

We use f as our input image for Algorithm 3.2. We initialise Algorithm 3.2 with $u^0 = 0$ and $q^0 = 0$. We choose $h_0 = 1/(r^2) \times \mathbf{1}_{r\times r}$ for $r = 35$ to ensure that h_0 satisfies the simplex constraint. We set $\tau^0 = 2$ and pick $\alpha \in \{10^{-1}, 10^{-3}\}$. We then iterate Algorithm 3.2 until the discrepancy principle is violated for $\eta = (1.2\sigma^2)/(2\sqrt{424 \times 640})$. The inner total variation sub-problem is solved with the primal-dual hybrid gradient method [84, 67, 37, 28, 29]. The results are visualised in Figure 6.1.

6.3. Classification. We test the proposed framework for the classification of images of hand-written digits. We use the well-known MNIST dataset [51] as the basis for our classification. Ten example images of each class are visualised in Figure 6.2a. We pick 50000 images from the training dataset to create our training data matrix D, and use the remaining 10000 for cross validation. We model our classifier as a two-level neural network as described in Section 5.3. We choose the original rectifier activation functions for the networks' architecture, in order to ensure that the composition is semi-algebraic and that the KL condition is satisfied. We overcome the non-differentiability by setting the derivatives to zero at the non-differentiable points. This is consistent with the smooth-max approximation of the rectifier for $\beta \to \infty$. We choose E to be the squared Frobenius norm and set the scaling parameters to $\alpha_1 = \alpha_2 = 0.2$. The stepsize τ^0 is initialised with $\tau^0 = 10^{-3}$. Subsequently, we run Algorithm 3.2 for 10000 iterations. The prediction results of the classifier and the rank of the trained matrices are visualised in Figure 6.2.

7. Conclusions & Outlook. We have presented a generalisation of gradient descent that allows the incorporation of non-smooth Bregman distances, and therefore can also be seen as an extension of the linearised Bregman iteration to non-convex functions. We have shown that the proposed method satisfies a sufficient decrease property and that the computed subgradients are bounded by the gap of the primal iterates. We have proven a global convergence result, where the limit is guaranteed to be a critical point of the energy if the subgradients are locally also bounded. The numerical experiments suggest that the proposed method to-

gether with early stopping can be designed to obtain solutions superior to those attained with conventional variational regularisation methods.

There are several open questions and natural directions that can be explored from here. One could extend the method to more general proximal mappings, as demonstrated in an earlier preprint. One could also study a linearised block coordinate variant of the proposed method, which would be similar in analysis to [80, 16]. In the wake of [63, 68], a generalisation of the proposed method could include inertial terms (or even multi-step inertial terms as in [53]), or Nesterov acceleration as in [43]. Both approaches seem intuitive for accelerating the method. Another direction that can be explored is the direction of non-smooth quasi-Newton extensions similar to [9]. Motivated by applications in deep learning, one could also follow up on incremental or stochastic variants of the proposed algorithm (cf. [44, 34, 14]). As we have used early stopping in our practical experiments, an interesting open question is whether the linearised Bregman iteration is a regularisation method, and if so, in what sense. This has been partially addressed in [5], but under more restrictive assumptions. Following diagonal iterative regularisation approaches, an interesting open question is also if the concept of [40] can be combined with the linearised Bregman iteration for non-convex problems.

REFERENCES

[1] A. Aizenbud and D. Gourevitch, *Schwartz functions on nash manifolds*, International Mathematics Research Notices, 2008 (2008).

[2] H. Attouch and J. Bolte, *On the convergence of the proximal algorithm for nonsmooth functions involving analytic features*, Mathematical Programming, 116 (2009), pp. 5–16.

[3] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, *Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-łojasiewicz inequality*, Mathematics of Operations Research, 35 (2010), pp. 438–457.

[4] H. Attouch, J. Bolte, and B. F. Svaiter, *Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods*, Mathematical Programming, 137 (2013), pp. 91–129.

[5] M. Bachmayr and M. Burger, *Iterative total variation methods for nonlinear inverse problems*, Inverse Problems, 25 (2009), p. 26, https://doi.org/10.1088/0266-5611/25/10/105004.

[6] H. H. Bauschke, J. Bolte, and M. Teboulle, *A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications*, Mathematics of Operations Research, 42 (2016), pp. 330–348.

[7] H. H. Bauschke, J. M. Borwein, and P. L. Combettes, *Essential smoothness, essential strict convexity, and Legendre functions in Banach spaces*, Communications in Contemporary Mathematics, 3 (2001), pp. 615–647.

[8] H. H. Bauschke, P. L. Combettes, et al., *Convex analysis and monotone operator theory in Hilbert spaces*, vol. 408, Springer, 2011.

[9] S. Becker and J. Fadili, *A quasi-newton proximal splitting method*, in Advances in Neural Information Processing Systems, 2012, pp. 2618–2626.

[10] M. Benning, M. M. Betcke, M. J. Ehrhardt, and C.-B. Schönlieb, *Gradient descent in a generalised Bregman distance framework*, in Geometric Numerical Integration and its Applications, G. R. W. Quispel, P. Bader, D. I. McLaren, and D. Tagami, eds., vol. 74, MI Lecture Notes series of Kyushu University, April 2017, pp. 40–45, http://www.imi.kyushu-u.ac.jp/eng/files/imipublishattachment/file/math_58ec341a238fe.pdf.

[11] M. Benning, L. Gladden, D. Holland, C.-B. Schönlieb, and T. Valkonen, *Phase reconstruction from velocity-encoded MRI measurements–a survey of sparsity-promoting variational approaches*, Journal of Magnetic Resonance, 238 (2014), pp. 26–43.

[12] M. Benning, F. Knoll, C.-B. Schönlieb, and T. Valkonen, *Preconditioned ADMM with nonlinear operator constraint*, in IFIP Conference on System Modeling and Optimization, Springer, 2015, pp. 117–126.
[13] D. Bertsekas, *On the Goldstein-Levitin-Polyak gradient projection method*, IEEE Transactions on automatic control, 21 (1976), pp. 174–184.
[14] D. P. Bertsekas, *Incremental gradient, subgradient, and proximal methods for convex optimization: A survey*, in Optimization for Machine Learning, S. Sra, S. and Nowozin, S. and Wright, ed., MIT Press, 2011, pp. 85–120.
[15] J. Bolte, P. L. Combettes, and J.-C. Pesquet, *Alternating proximal algorithm for blind image recovery*, in 2010 IEEE International Conference on Image Processing, IEEE, 2010, pp. 1673–1676.
[16] J. Bolte, S. Sabach, and M. Teboulle, *Proximal alternating linearized minimization for nonconvex and nonsmooth problems*, Mathematical Programming, 146 (2014), pp. 459–494.
[17] J. Bolte, S. Sabach, M. Teboulle, and Y. Vaisbourd, *First order methods beyond convexity and Lipschitz gradient continuity with applications to quadratic inverse problems*, arXiv preprint arXiv:1706.06461, (2017).
[18] S. Bonettini, I. Loris, F. Porta, and M. Prato, *Variable metric inexact line-search based methods for nonsmooth optimization*, SIAM Journal on Optimization, 26 (2016), pp. 891–921, https://doi.org/http://dx.doi.org/10.1137/15M1019325, http://arxiv.org/abs/1506.00385.
[19] S. Bonettini, I. Loris, F. Porta, M. Prato, and S. Rebegoldi, *On the convergence of a linesearch based proximal-gradient method for nonconvex optimization*, Inverse Problems, (2017), https://doi.org/http://dx.doi.org/10.1088/1361-6420/aa5bfd, https://arxiv.org/abs/1605.03791. Accepted.
[20] L. M. Bregman, *The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming*, USSR computational mathematics and mathematical physics, 7 (1967), pp. 200–217.
[21] M. Burger, G. Gilboa, S. Osher, J. Xu, et al., *Nonlinear inverse scale space methods*, Communications in Mathematical Sciences, 4 (2006), pp. 179–212.
[22] M. Burger, M. Möller, M. Benning, and S. Osher, *An adaptive inverse scale space method for compressed sensing*, Mathematics of Computation, 82 (2013), pp. 269–299.
[23] M. Burger, E. Resmerita, and L. He, *Error estimation for Bregman iterations and inverse scale space methods in image restoration*, Computing, 81 (2007), pp. 109–135.
[24] J.-F. Cai, S. Osher, and Z. Shen, *Convergence of the linearized Bregman iteration for ℓ^1-norm minimization*, Mathematics of Computation, 78 (2009), pp. 2127–2136.
[25] J.-F. Cai, S. Osher, and Z. Shen, *Linearized Bregman iterations for compressed sensing*, Mathematics of Computation, 78 (2009), pp. 1515–1536.
[26] P. Campisi and K. Egiazarian, *Blind image deconvolution: theory and applications*, CRC press, 2016.
[27] Y. Censor and S. A. Zenios, *Proximal minimization algorithm with d-functions*, Journal of Optimization Theory and Applications, 73 (1992), pp. 451–464.
[28] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of mathematical imaging and vision, 40 (2011), pp. 120–145.
[29] A. Chambolle and T. Pock, *An introduction to continuous optimization for imaging*, Acta Numerica, 25 (2016), pp. 161–319.
[30] T. F. Chan and J. Shen, *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*, Other titles in applied mathematics, Society for Industrial and Applied Mathematics, 2005.
[31] T. F. Chan and C.-K. Wong, *Total variation blind deconvolution*, IEEE Transactions on Image Processing, 7 (1998), pp. 370–375.
[32] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, *A block coordinate variable metric forward–backward algorithm*, Journal of Global Optimization, 66 (2016), pp. 457–485.
[33] J. Darbon and S. Osher, *Fast discrete optimization for sparse approximations and deconvolutions*, (2007).
[34] A. Defazio, F. Bach, and S. Lacoste-Julien, *Saga: A fast incremental gradient method with support for non-strongly convex composite objectives*, Nips, (2014), pp. 1–12, https://arxiv.org/abs/arXiv:1407.0202v2.
[35] D. Drusvyatskiy, A. D. Ioffe, and A. S. Lewis, *Nonsmooth optimization using Taylor-like models: error bounds, convergence, and termination criteria*, arXiv preprint arXiv:1610.03446, (2016).

[36] J. Eckstein, *Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming*, Mathematics of Operations Research, 18 (1993), pp. 202–226.
[37] E. Esser, X. Zhang, and T. F. Chan, *A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 1015–1046.
[38] K. Frick and O. Scherzer, *Regularization of ill-posed linear equations by the non-stationary augmented lagrangian method*, The Journal of Integral Equations and Applications, (2010), pp. 217–257.
[39] D. Gabay, *Chapter ix applications of the method of multipliers to variational inequalities*, in Studies in mathematics and its applications, vol. 15, Elsevier, 1983, pp. 299–331.
[40] G. Garrigos, L. Rosasco, and S. Villa, *Iterative regularization via dual diagonal descent*, arXiv preprint arXiv:1610.02170, (2016).
[41] A. A. Goldstein, *Convex programming in Hilbert space*, Bulletin of the American Mathematical Society, 70 (1964), pp. 709–710.
[42] A. A. Goldstein, *Constructive real analysis*, tech. report, Washington Univ. Seattle Dept. of Mathematics, 1967.
[43] B. Huang, S. Ma, and D. Goldfarb, *Accelerated linearized bregman method*, Journal of Scientific Computing, 54 (2013), pp. 428–453.
[44] R. Johnson and T. Zhang, *Accelerating Stochastic Gradient Descent using Predictive Variance Reduction*, Nips, 1 (2013), pp. 315–323.
[45] B. Kaltenbacher, F. Schöpfer, and T. Schuster, *Iterative methods for nonlinear ill-posed problems in banach spaces: convergence and applications to parameter identification problems*, Inverse Problems, 25 (2009), p. 065003.
[46] K. C. Kiwiel, *Proximal minimization methods with generalized Bregman functions*, SIAM journal on control and optimization, 35 (1997), pp. 1142–1168.
[47] F. Knoll, K. Bredies, T. Pock, and R. Stollberger, *MRI raw data: T2 weighted TSE scan of a healthy volunteer (4 channel head coil)*, Dec. 2010, https://doi.org/10.5281/zenodo.800525, https://doi.org/10.5281/zenodo.800525.
[48] F. Knoll, C. Clason, K. Bredies, M. Uecker, and R. Stollberger, *Parallel imaging with nonlinear reconstruction using variational penalties*, Magnetic Resonance in Medicine, 67 (2012), pp. 34–41.
[49] D. Kundur and D. Hatzinakos, *Blind image deconvolution*, IEEE Signal Processing Magazine, 13 (1996), p. 43, https://doi.org/10.1109/79.489268.
[50] K. Kurdyka, *On gradients of functions definable in o-minimal structures*, in Annales de l'institut Fourier, vol. 48, Chartres: L'Institut, 1950-, 1998, pp. 769–784.
[51] Y. LeCun, C. Cortes, and C. J. Burges, *MNIST handwritten digit database*, AT&T Labs [Online], 2 (2010), http://yann.lecun.com/exdb/mnist.
[52] G. Li and T. K. Pong, *Global convergence of splitting methods for nonconvex composite optimization*, SIAM Journal on Optimization, 25 (2015), pp. 2434–2460.
[53] J. Liang, J. Fadili, and G. Peyré, *A multi-step inertial forward-backward splitting method for non-convex optimization*, in Advances in Neural Information Processing Systems, 2016, pp. 4035–4043.
[54] P.-L. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM Journal on Numerical Analysis, 16 (1979), pp. 964–979.
[55] S. Lojasiewicz, *Une propriété topologique des sous-ensembles analytiques réels*, Les équations aux dérivées partielles, 117 (1963), pp. 87–89.
[56] S. Matet, L. Rosasco, S. Villa, and B. L. Vu, *Don't relax: early stopping for convex regularization*, arXiv preprint arXiv:1707.05422, (2017).
[57] M. Moeller, M. Benning, C. Schönlieb, and D. Cremers, *Variational depth from focus reconstruction*, IEEE Transactions on Image Processing, 24 (2015), pp. 5369–5378.
[58] J.-J. Moreau, *Décomposition orthogonale d'un espace hilbertien selon deux cônes mutuellement polaires*, CR Acad. Sci. Paris, 225 (1962), pp. 238–240.
[59] J.-J. Moreau, *Proximité et dualité dans un espace hilbertien*, Bull. Soc. Math. France, 93 (1965), pp. 273–299.
[60] V. A. Morozov, *Methods for solving incorrectly posed problems*, Springer Science & Business Media, 2012.
[61] M. Nikolova and P. Tan, *Alternating proximal gradient descent for nonconvex regularised problems*

with multiconvex coupling terms, (2017).
[62] J. Nocedal and S. J. Wright, *Numerical optimization 2nd*, 2006.
[63] P. Ochs, Y. Chen, T. Brox, and T. Pock, *ipiano: Inertial proximal algorithm for nonconvex optimization*, SIAM Journal on Imaging Sciences, 7 (2014), pp. 1388–1419.
[64] P. Ochs, J. Fadili, and T. Brox, *Non-smooth non-convex Bregman minimization: Unification and new algorithms*, arXiv preprint arXiv:1707.02278, (2017).
[65] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, *An iterative regularization method for total variation-based image restoration*, Multiscale Modeling & Simulation, 4 (2005), pp. 460–489.
[66] S. Osher, F. Ruan, J. Xiong, Y. Yao, and W. Yin, *Sparse recovery via differential inclusions*, Applied and Computational Harmonic Analysis, 41 (2016), pp. 436–469.
[67] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, *An algorithm for minimizing the Mumford-Shah functional*, in Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1133–1140.
[68] T. Pock and S. Sabach, *Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems*, SIAM Journal on Imaging Sciences, 9 (2016), pp. 1756–1787.
[69] M. Prato, S. Bonettini, I. Loris, F. Porta, and S. Rebegoldi, *On the constrained minimization of smooth Kurdyka-Łojasiewicz functions with the scaled gradient projection method*, Journal of Physics: Conference Series, 756 (2016), p. 012001, https://doi.org/http://dx.doi.org/10.1088/1742-6596/756/1/012001.
[70] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, *SENSE: Sensitivity Encoding for Fast MRI*, Magnetic Resonance in Medicine, 42 (1999), pp. 952–62.
[71] S. Ramani and J. A. Fessler, *Parallel MR image reconstruction using augmented lagrangian methods*, IEEE Transactions on Medical Imaging, 30 (2011), pp. 694–706.
[72] A. Repetti, M. Q. Pham, L. Duval, E. Chouzenoux, and J.-C. Pesquet, *Euclid in a taxicab: Sparse blind deconvolution with smoothed ell$_1$-ell$_2$ regularization*, IEEE Signal Processing Letters, 22 (2014), pp. 539–543.
[73] R. T. Rockafellar, *Convex Analysis*, vol. 28, Princeton University Press, 1970.
[74] R. T. Rockafellar and R. J.-B. Wets, *Variational analysis*, vol. 317, Springer Science & Business Media, 2009.
[75] F. Schöpfer, A. K. Louis, and T. Schuster, *Nonlinear iterative methods for linear ill-posed problems in Banach spaces*, Inverse Problems, 22 (2006), p. 311.
[76] M. Teboulle, *Entropic proximal mappings with applications to nonlinear programming*, Mathematics of Operations Research, 17 (1992), pp. 670–690.
[77] M. Uecker, T. Hohage, K. T. Block, and J. Frahm, *Image reconstruction by regularized nonlinear inversion-joint estimation of coil sensitivities and image content*, Magnetic Resonance in Medicine, 60 (2008), pp. 674–682.
[78] T. Valkonen, *A primal–dual hybrid gradient method for nonlinear operators with applications to MRI*, Inverse Problems, 30 (2014), p. 055012.
[79] H. Wang and A. Banerjee, *Bregman alternating direction method of multipliers*, in Advances in Neural Information Processing Systems, 2014, pp. 2816–2824.
[80] Y. Xu and W. Yin, *A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion*, SIAM Journal on imaging sciences, 6 (2013), pp. 1758–1789.
[81] Y. Xu and W. Yin, *A globally convergent algorithm for nonconvex optimization based on block coordinate update*, Journal of Scientific Computing, (2017), pp. 1–35.
[82] W. Yin, *Analysis and generalizations of the linearized Bregman method*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 856–877.
[83] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, *Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing*, SIAM Journal on Imaging sciences, 1 (2008), pp. 143–168.
[84] M. Zhu and T. Chan, *An efficient primal-dual hybrid gradient algorithm for total variation image restoration*, UCLA CAM Report, 34 (2008).

Appendix A. Mathematical preliminaries.

We briefly summarise several concepts of convex and non-convex analysis that are of importance for the main part of this paper. De-

tailed informations about these concepts can be found in various textbooks, such as [73, 8]. We frequently use functions that are proper, lower semi-continuous and convex, and therefore define the following set of functions:

$$\Gamma_0 := \{J : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\} \,|\, J \text{ is proper, lower semi-continuous and convex}\} \,.$$

Here proper means that the effective domain of J is not empty. The effective domain of J is defined as follows.

Definition A.1 (Effective domain). *The* effective domain *of a function* $J : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ *is defined as*

$$\mathrm{dom}(J) := \{u \in \mathbb{R}^n \,|\, J(u) < \infty\} \,.$$

Convex and proper functions are not necessarily differentiable, but subdifferentiable. We therefore want to recall the definition of subgradients and the subdifferential of a convex function.

Definition A.2 (Subdifferential). *Let* $J \in \Gamma_0$. *The function* J *is called* subdifferentiable *at* $u \in \mathbb{R}^n$, *if there exists an element* $p \in \mathbb{R}^n$ *such that*

$$J(v) \geq J(u) + \langle p, v - u \rangle$$

holds, for all $v \in \mathbb{R}^n$. *Furthermore, we call* p *a* subgradient *at position* u. *The collection of all subgradients at position* u, *i.e.*

$$\partial J(u) := \{p \in \mathbb{R}^n \,|\, J(v) \geq J(u) + \langle p, v - u \rangle \,, \forall v \in \mathbb{R}^n\} \,,$$

is called subdifferential *of* J *at* u.

Another useful concept that we want to recall is the concept of Fenchel-, respectively convex-conjugates.

Definition A.3 (Convex conjugate). *Let* $J \in \Gamma_0$. *Then its* convex conjugate $J^* : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ *is defined as*

$$J^*(p) := \sup_{u \in \mathbb{R}^n} \{\langle u, p \rangle - J(u)\} \,,$$

for all $p \in \mathbb{R}^n$.

Amongst others, subgradients of convex conjugates satisfy the following two useful properties.

Lemma A.4. *Let* $J \in \Gamma_0$, *and* J^* *denote the convex conjugate of* J. *Then for all arguments* $u \in \mathbb{R}^n$ *with corresponding subgradients* $p \in \partial J(u)$ *we know*

- $\langle u, p \rangle = J(u) + J^*(p)$,
- $p \in \partial J(u)$ *is equivalent to* $u \in \partial J^*(p)$.

Bregman distances, introduced by Lev Bregman in 1967 (see [20]), play a vital role in the definition as well as in the convergence analysis of the linearised Bregman iteration for non-convex functions. We recall its generalised variant for subdifferentiable functions [46].

Definition A.5 (Bregman distance). *Let $J \in \Gamma_0$. Then the* generalised Bregman distance *for a particular subgradient $q \in \partial J(v)$ is defined as*

$$D_J^q(u,v) := J(u) - J(v) - \langle q, u - v\rangle\,, \tag{A.1}$$

for $v \in \mathrm{dom}(J)$ and all $u \in \mathbb{R}^n$.

Remark 6. *Based on Lemma A.4 we can rewrite (A.1) as follows:*

$$D_J^q(u,v) = J(u) + J^*(q) - \langle u, q\rangle\,. \tag{A.2}$$

Noticeable, the Bregman distance does not depend on v anymore, and could therefore be defined as a function of u and q only, $D_J(u,q)$, via (A.2) instead.

Bregman distances are not symmetric in general; however, they satisfy a dual symmetry $D_J^q(u,v) = D_{J^*}^u(q,p)$ for arguments $u \in \mathbb{R}^{\mathrm{n}}$, $v \in \mathrm{dom}(J)$ and subgradients $p \in \partial J(u)$ and $q \in \partial J(v)$. Symmetry can nevertheless be achieved by simply adding two Bregman distances with interchanged arguments. The name symmetric Bregman distance goes back to [23].

Definition A.6 (Symmetric Bregman distance). *Let $J \in \Gamma_0$. Then the* symmetric generalised Bregman distance $D_J^{symm}(u,v)$ *is defined as*

$$D_J^{symm}(u,v) := D_J^q(u,v) + D_J^p(v,u) = \langle p - q, u - v\rangle\,,$$

for $u, v \in \mathrm{dom}(J)$ with $p \in \partial J(u)$ and $q \in \partial J(v)$.

Another concept that we exploit is Lipschitz-continuity of the gradient of a function. For general operators, Lipschitz-continuity is defined as follows.

Definition A.7 (Lipschitz-continuity). *An operator $F : U \subset \mathbb{R}^{\mathrm{n}} \to \mathbb{R}^m$ is said to be* (globally) Lipschitz-continuous *if there exists a constant $L \geq 0$ such that*

$$\|F(u) - F(v)\| \leq L\|u - v\| \tag{A.3}$$

is satisfied for all $u, v \in U$.

Due to the importance of Lipschitz-continuous gradients, we define the following class of continuously differentiable functions with Lipschitz-continuous gradient:

Definition A.8 (Smoothness). *A function $J : U \subset \mathbb{R}^n \to \mathbb{R}$ is called* L-smooth *if it is differentiable and its gradient $\nabla J : U \to \mathbb{R}^{\mathrm{n}}$ is Lipschitz-continuous with Lipschitz constant L. The set of all L-smooth functions is therefore denoted by $\mathcal{S}_L$ with*

$$\mathcal{S}_L := \left\{ J : U \to \mathbb{R} \;\middle|\; \begin{array}{c} J \text{ is continuously differentiable} \\ \nabla J \text{ is } L\text{-Lipschitz-continuous} \end{array} \right\}.$$

Note that it is a well-known fact that L-smooth functions satisfy the Lipschitz estimate

$$J(u) \leq J(v) + \langle \nabla J(v), u - v\rangle + \frac{L}{2}\|u - v\|_2^2\,, \tag{A.4}$$

for all $u, v \in U$. Note that if $U = \mathbb{R}^n$ then J is already globally L-smooth and this estimate is true for all arguments $u, v \in \mathbb{R}^n$.
In the following we recall the definition of the proximal mapping.

Definition A.9 (Proximal mapping [58, 59]). *We define the* proximal mapping *as the operator* $(I + \partial J)^{-1} : \mathbb{R}^{\mathrm{n}} \to \mathrm{dom}(J)$ *with*

$$(I + \partial J)^{-1}(f) := \underset{u \in \mathrm{dom}(J)}{\arg\min} \left\{ \frac{1}{2}\|u - f\|^2 + J(u) \right\},$$

for all arguments $f \in \mathbb{R}^{\mathrm{n}}$.

To conclude this section, we want to recall the Kurdyka-Łojasiewicz (KL) property [55, 50]. For the definition of the KL property we need to define a distance between sub-sets and elements of $\mathbb{R}^n$ first.

Definition A.10. *Let* $\Omega \subset \mathbb{R}^n$ *and* $u \in \mathbb{R}^n$. *We define the distance from* Ω *to* u *as*

$$\mathrm{dist}(u, \Omega) := \begin{cases} \inf\{\|v - u\| \,|\, v \in \Omega\} & \Omega \neq \emptyset \\ \infty & \Omega = \emptyset \end{cases}.$$

The definition of the KL property based on the distance measure defined in Definition A.10 reads as follows.

Definition A.11 (Kurdyka-Łojasierwicz property). *A function* J *is said to have the* Kurdyka-Łojasierwicz (KL) property *at* $\overline{u} \in dom(\partial J) := \{u \in \mathbb{R}^n \,|\, \partial J(u) \neq \emptyset\}$ *if there exists a constant* $\eta \in (0, \infty]$, *a neighbourhood* Θ *of* $\overline{u}$ *and a function* $\varphi : [0, \eta) \to \mathbb{R}_{>0}$, *which is a concave function that is continuous at 0 and satisfies* $\varphi(0) = 0$, $\varphi \in C^1((0, \eta))$ *and* $\varphi'(s) > 0$ *for all* $s \in (0, \eta)$, *such that for all* $u \in \Theta \cap \{u \in \mathbb{R}^n \,|\, J(\overline{u}) < J(u) < J(\overline{u}) + \eta\}$ *the inequality*

$$\varphi'\left(J(u) - J(\overline{u})\right) \mathrm{dist}(0, \partial J(u)) \geq 1 \tag{KL}$$

holds.
If J *satisfies the KL property at each point of* $\mathrm{dom}(\partial J)$, J *is called a* KL function.

We conclude the appendix by recalling one important result from [16] that is necessary for successfully carrying out the convergence proof in the main part of the paper.

Lemma A.12 (Uniformised KL property [16, Lemma 6]). *Let* Ω *be a compact set, and suppose that* J *is a function that is constant on* Ω *and that satisfies* (KL) *at each point in* Ω. *Then there exist* $\varepsilon > 0$, $\eta > 0$ *and* $\varphi \in C^1((0, \eta))$ *that satisfy the same conditions as in Definition A.11, such that for all* $\overline{u} \in \Omega$ *and all* u *in*

$$\{u \in \mathbb{R}^n \,|\, \mathrm{dist}(u, \Omega) < \varepsilon\} \cap \{u \in \mathbb{R}^n \,|\, J(\overline{u}) < J(u) < J(\overline{u}) + \eta\} \tag{A.5}$$

condition (KL) *is satisfied.*