Contents

1 Proximal Bundle Method for Nonconvex Functions with Inexact Information

This section focuses on the proximal bundle method presented by Hare et al. in [4]. The idea is to extend the basic bundle algorithm for nonconvex functions with both inexact function and subgradient information. The key idea of the algorithm is the one already developed by Hare and Sagastizábal in [3]: When dealing with nonconvex functions a very critiicalcal difference to the convex case is that the linearization errors are not necessarily nonnegative any more. To tackle this problem the errors are manipulated to enforce nonnegativity. In this case this is done my modeling not the objective function directly but a convexified version of it.

1.1 Derivation of the Method

Throughout this section we consider the optimization problem

$$\min_x f(x) \quad \text{s.t.} \quad x \in X. \tag{1.1}$$

The objective function $f : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz and (subdifferentially) regular. $X \subseteq \mathbb{R}^n$ is assumed to be a convex compact set.

Definition 1.1 [14, Theorem 7.25] $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ is called *subdifferentially regular* at $\bar{x}$ if $f(\bar{x})$ is finite and the epigraph

$$epi(f) := \{(x, \alpha) \in \mathbb{R}^n \times \mathbb{R} | \alpha \geq f(x)\}$$

is Clarke regular at $\bar{x}, f(\bar{x})$.

1.1.1 Inexactness

It is assumed that both the function value as well as one element of the subdifferential can be provided in an inexact form. For the function value inexactness is defined straight forwardly: If

$$\|f_x - f(x)\| \leq \sigma_x$$

then f_x approximates the value $f(x)$ within σ_x. This is slightly different from the definition in (**??**). In the convex case it follows from (**??**) that $\bar{\sigma} \geq \sigma_x \geq -\theta_x \geq -\bar{\theta}$ and

therefore $f_x \in [f(x) - \bar{\theta}, f(x) + \bar{\sigma}]$.

As the 'normal' ε-subdifferential is not defined for nonconvex functions we adopt the notation used in [11] and interpret inexactness in the following way: $g \in \mathbb{R}^n$ approximates a subgradient of $\partial f(x)$ within $\theta \geq 0$ if

$$g \in \partial f(x) + B_\theta(0) := \partial_{[\theta]} f(x)$$

where $\partial f(x)$ is the Clarke subdifferential of f.

The given definition of the inexactness can be motivated by the relation

$$g \in \partial_{[\theta]} f(x) \Leftrightarrow g \in \partial(f + \theta\| \cdot -x\|)(x)$$

noticed in [15]. It means that the approximation of the subgradient of $f(x)$ is an exact subgradient of a small perturbation of f at x. $\partial_{[\varepsilon]} f(x)$ is also known as the Fréchet ε-subdifferential of $f(x)$.

Remark: For convex objective functions this approximate subdifferential does *not* equal the usual convex ε-subdifferential. The two can however be related via

$$\partial_\theta f(x) \subset \partial_{[\theta']} f(x)$$

for a suitable θ'. Generally an explicit relation between θ and θ' is hard to find [11].

Like in the paper it is assumed that the errors are bounded although the bound does not have to be known:

$$|\sigma_j| \leq \bar{\sigma}, \bar{\sigma} > 0 \quad \text{and} \quad 0 \leq \theta_j \leq \bar{\theta} \quad \forall j \in J^k.$$

For ease of notation we write from now on f_j instead of f_{x^j} for the approximation of the function value at the j'th iterate in the bundle J. The approximation at the k'th stability center reads $\hat{f}_k$.

1.1.2 Nonconvexity

A main issue both nonconvexity and inexactness entail is that the linearization errors e_j^k are not necessarily nonnegative any more. So based on the results in [17] not the

objective function but a convexified version of it is modeled as the objective function of the subproblem.

As already pointed out in **??** the bundle subproblem can be formulated by means of the prox-operator (**??**).

The key idea is to use the relation

$$prox_{T=\frac{1}{\eta}+t,f}(x) = prox_{t,f+\eta/2\|\cdot-x\|^2}(x).$$

This means, that the proximal point of the function f for parameter $T = \frac{1}{\eta} + t$, $\eta, t > 0$ is the same as the one of the convexified function

$$\tilde{f}(y) = f(y) + \frac{\eta}{2}\|y - x\|^2 \tag{1.2}$$

with respect to the parameter t [3]. η is therefore called the *convexification parameter* and t the *prox-parameter*.

The main difference of the method in [4] to the basic bundle method is that the function that is modeled by the cutting plane model s no longer the original objective function f but the convexified version $\tilde{f}$. This results in the following changes:

In addition to downshifting the linear functions forming the model they have a tilted slope. This is because instead of subgradients of the original objective f subgradients of the function $\tilde{f}$ are taken. We call them *augmented subgradients*. At the iterate x^j it is given by

$$s_j^k = g^j + \eta_k \left(x^j - \hat{x}^k\right).$$

Downshifting is done in a way that keeps the linearization error nonnegative. The *augmented linearization error* is therefore defined as

$$0 \leq c_j^k := e_j^k + b_j^k, \quad \text{with} \quad \begin{cases} e_j^k := \hat{f}_k - f_j - \left\langle g^j, \hat{x}^k - x^j \right\rangle \\ b_j^k := \frac{\eta_k}{2}\|x^j - \hat{x}^k\|^2 \end{cases}$$

and

$$\eta_k \geq \max\left\{\max_{j\in J_k, x^j\neq\hat{x}^k} \frac{-2e_j^k}{\|x^j-\hat{x}^k\|^2}, 0\right\} + \gamma.$$

The parameter $\gamma \geq 0$ is a safeguarding parameter to keep the calculations numerically stable.

The new model function can therefore be written as

$$M_k(\hat{x}^k + d) := \hat{f}_k + \max_{j\in J_k}\left\{\left\langle s_j^k, d\right\rangle - c_j^k\right\}.$$

At the proximal center $\hat{x}^k$ holds $M_k(\hat{x}^k))\hat{f}_k$ for all k by the fact that then $d = 0$ and $c_j^k = 0$.

1.1.3 Aggregate Objects

The definition of the *augmented aggregate subgradient* S^k, *error* C_k and *linearization* A_k follows straightforwardly:

$$S^k := \sum_{j\in J_k} \alpha_j^k s_j^k, \tag{1.3}$$

$$C_k := \sum_{j\in J_k} \alpha_j^k c_j^k \tag{1.4}$$

$$A_k(\hat{x}^k + d) := M_k(x^{k+1}) + \left\langle S^k, d - d^k\right\rangle. \tag{1.5}$$

Just as the model decrease

$$\delta^k := C_k + t_k\|S^k + \nu^k\|^2 = C_k + \frac{1}{t_k}\|d^k\|^2, \tag{1.6}$$

which contains the normal vector

$$\nu^k \in \partial \mathtt{i}_X(x^{k+1}) \tag{1.7}$$

of the constraint set X.

The second formulation in (1.6) follows from the relation $d^k = -t_k(S^k + \nu^k)$.

By the same argumentation as for (**??**) the KKT conditions also reveal another useful characterization of the augmented aggregate linearization error:

$$C_k = \hat{f}_k - M_k(x^{k+1}) + \left\langle S^k, d^k \right\rangle \tag{1.8}$$

As the model function M_k is convex even for nonconvex objective functions it is still minorized by the aggregate linearization. It holds

$$A_K(\hat{x}^k + d) \leq M_k(\hat{x}^k + d) \quad \forall d \in \mathbb{R}^n. \tag{1.9}$$

The update of t_k can be done in the same way described in (**??**) and (**??**) for the basic bundle method. Similarly the methods to update the bundle index set J^k stay valid. The update conditions (**??**) and (**??**) for the model are now written with respect to the augmented aggregate linearization and the approximate function value $\hat{f}_{k+1}$.

$$M_{k+1}(\hat{x}^k + d) \geq \hat{f}_{k+1} - c_{k+1}^{k+1} + \left\langle s^{k+1}, d \right\rangle \quad \forall d \in \mathbb{R}^n \tag{1.10}$$

$$M_{k+1}(\hat{x}^k + d) \geq A_k(\hat{x}^k + d) \quad \forall d \in \mathbb{R}^n. \tag{1.11}$$

A bundle algorithm that deals with nonconvexity and inexact function and subgradient information can now be stated.

Algorithm 1.1: Nonconvex Proximal Bundle Method with Inexact Information

---

Select parameters $m \in (0,1), \gamma > 0$ and a stopping tolerance $\texttt{tol} \geq 0$.
Choose a starting point $x^1 \in \mathbb{R}^n$ and compute f_1 and g^1. Set the initial index set $J_1 := \{1\}$ and the initial prox-center to $\hat{x}^1 := x^1$, $\hat{f}_1 = f_1$ and select $t_1 > 0$.
For $k = 1, 2, 3, \ldots$

1. Calculate
$$d^k = \arg\min_{d \in \mathbb{R}^n} \left\{ M_k(\hat{x}^k + d) + \mathbb{I}_X(\hat{x}^k + d) + \frac{1}{2t_k}\|d\|^2 \right\}.$$

2. Set
$$\begin{aligned} G^k &= \sum_{j \in J_k} \alpha_j^k s_j^k \\ C_k &= \sum_{j \in J_k} \alpha_j^k c_j^k, \\ \delta_k &= C_k + \frac{1}{t_k}\|d^k\|^2. \end{aligned}$$

If $\delta_k \leq \texttt{tol} \rightarrow$ STOP.

3. Set $x^{k+1} = \hat{x}^k + d^k$.
4. Compute f^{k+1}, g^{k+1}.
   If
   $$f^{k+1} \leq \hat{f}^k - m\delta_k \quad \rightarrow \text{ serious step}$$
   Set $\hat{x}^{k+1} = x^{k+1}$, $\hat{f}^{k+1} = f^{k+1}$ and select $t_{k+1} > 0$.
   Otherwise $\rightarrow$ nullstep
   Set $\hat{x}^{k+1} = \hat{x}^k$, $\hat{f}^{k+1} = f^{k+1}$ and choose $0 < t_{k+1} \leq t_k$.
5. Select new bundle index set J_{k+1}, calculate
   $$\eta_k = \max\left\{\max_{j \in J_{k+1}, x^j \neq \hat{x}^{k+1}} \frac{-2e_j^k}{|x^j - \hat{x}^{k+1}|^2}, 0\right\} + \gamma$$
   and update the model M_k.

1.2 On Different Convergence Results

In terms of usability of the described algorithm it is interesting to see if stronger convergence results are possible if additional assumptions are put on the objective function. This is investigated in the following section.

1.2.1 The Constraint Set

The constraint set X ensures the boundedness of the sequence $\{\hat{x}^k\}$. This is not necessary if the objective function is assumed to have bounded level sets $\{x \in \mathbb{R}^n | f(x) \leq f(\hat{x}^1)\}$, an assumption commonly used when optimizing nonconvex functions. As the objective function is assumed to be continuous bounded level sets are compact. Additionally the descent test ensures that $f(\hat{x}^{k+1}) \leq f(\hat{x}^k)$ for all k. The proof holds therefore in the same way as with the set X.

Another possibility is to bound the step sizes t_k also from above. Then the sequence $\{\hat{x}^k\}$ stays bounded and the proof still holds. In [18] another stopping criterion is supposed that ensures convergence even for unbounded sequences $\{\hat{x}^k\}$. Is this also possible in my case or only for convex???

1.2.2 Exact Information and Vanishing Errors

As the presented algorithm was originally designed for nonconvex objective functions where function values as well as subgradients are available in an exact manner, all convergence results stay the same with the error bounds $\bar{\sigma} = \bar{\theta} = 0$. As already indicated previously this is the case because inexactness can be seen as a kind of nonconvexity and no additional concepts had to be added to the method when generalizing it to the inexact setting.

If we additionally require the objective function to be lower-$\mathcal{C}^2$ it can be proven that the sequence $\{\eta_k\}$ is bounded [3]. This is not possible in the case of inexact information even for convex objective functions.

For asymptotically vanishing errors, meaning $\lim_{k\to\infty} \sigma_k = 0$ and $\lim_{k\to\infty} \theta_k = 0$ the convergence theory holds equally well with error bounds $\bar{\sigma} = \bar{\theta} = 0$ in [4, Lemma 5]. Still it is difficult if not impossible to show that the sequence $\{\eta_k\}$ is bounded without further assumptions. Under the assumption that f is lower-$\mathcal{C}^2$ and some continuity bounds on the errors

$$\frac{|\sigma_j - \hat{\sigma}_k|}{\|x^j - \hat{x}^k\|^2} \leq L_\sigma, \qquad \frac{\theta_j}{\|x^j - \hat{x}^k\|} \leq L_\theta \quad \forall k \text{ and } \forall j \in J_k$$

boundedness of the sequence $\{\eta_k\}$ can be shown. The question remains however if those assumptions are possible to be assured in practice.

remark on η_k? how does it behave in my applications???

1.2.3 Convex Objective Functions

An obvious gain when working with convex objective functions is that the approximate stationarity condition of [4, Lemma 5 (iii)] is now an approximate optimality condition. If one takes the error definitions (**??**) and (**??**) that are available in the convex case and assumes $X = \mathbb{R}^n$ statement (22) in [4] therefore means that

$$0 \in \partial_{\bar{\sigma}+\bar{\theta}} f(\bar{x}).$$

Thus $\bar{x}$ is $(\bar{\sigma} + \bar{\theta})$-optimal.

This follows from the definition of S^k in (1.3) and local Lipschitz continuity of the ε-subdifferential [14, Proposition 12.68].

beweis für $\bar{\sigma}$-optimalität

bounded t_k instead of D? better????

To conclude this section we can say: At the moment there exist two fundamentally different approaches to tackle inexactness in various bundle methods depending on if the method is developed for convex or nonconvex objective functions. In the nonconvex case inexactness is only considered in the paper by Hare, Sagastizàbal and Sodolov [4] presented above and Noll [11]. In these cases the inexactness can be seen as an additional nonconvexity. In practice this means that the algorithm can be taken from the nonconvex case with no or only minor changes. This includes that all results of the exact case remain true as soon as function and subgradient are evaluated in an exact way. In case of convex objective functions with inexact information stronger convergence results are possible. However to be able to exploit convexity in order to achieve those results the algorithms look different from those designed for nonconvex objective functions and are generally not able to deal with such functions.

2 Variable Metric Bundle Method

A way to extend the proximal bundle method is to use an arbitrary metric $\frac{1}{2}\langle d, W_k d\rangle$ with a symmetric and positive definite matrix W_k instead of the Euclidean metric for the stabilization term $\frac{1}{2t_k}\|d\|^2$. Methods doing so are called *variable metric bundle methods*. This section combines the method of Hare et al. presented in section 1 with the second order model function used by Noll in [11] to a metric bundle method suitable for nonconvex functions with noise.

The section starts by explaining the ideas from [11] used to extend the method presented above. It then gives an explicit strategy how to update the metric during the steps of the algorithm and concludes with a convergence proof for the developed method.

Throughout this section we still consider the optimization problem (1.1). We also keep the names and definitions of the objects used in section 1.

2.1 Main Ingredients to the Method

As already mentioned in section **??** the stabilization term can be interpreted in many different ways. In the context of this section we can understand it as a pretty rough approximation of the curvature of the objective function. Of course bundle methods are designed to work with non differentiable objectives so it cannot be expected that

the function provides any kind of curvature. However, if it has regions where there is curvature, this information can be used to speed up convergence.

2.1.1 Variable Metric Bundle Methods

Variable metric bundle methods use an approach that can be motivated by the thoughts stated above. Instead of using the Euclidean norm for the stabilization term $\frac{1}{2}\|d\|^2$ the metric is derived from a symmetric and positive definite matrix W_k. As the name of the method suggests, this matrix can vary over the iterations of the algorithm. The subproblem in the k'th iteration therefore reads

$$\min_{\hat{x}^k+d\in\mathbb{R}^n} M_k(\hat{x}^k+d) + \mathtt{i}_X(\hat{x}^k+d) + \frac{1}{2}\langle d, W_k d\rangle .$$

As explained in [5] like (**??**) this is a Moreau-Yosida regularization of the objective function (on the constraint set), so this subproblem is still strictly convex and has a unique solution. It is however harder to solve especially if the matrices W_k are no diagonal matrices [8]. In the unconstrained case or for a very simple constraint set the subproblem can be solved by calculating a quasi Newton step. Such a method is presented by Lemaréchal and Sagastizábal in [6] for convex functions. Lukšan and Vlček use an algorithm in those lines in [16] which is adapted to a limited memory setting by Haarala et al. in [2].

A challenging question is how to update the matrices W_k. It is important that the updating strategy preserves positive definiteness of the matrices and that the matrices stay bounded. The updates that are used most often are the symmetric rank 1 formula (SR1 update) and the BFGS (Broyden-Fletcher-Goldfarb-Shanno) update. These updates make it possible to assure the required conditions with only little extra effort even in the nonconvex case. Concrete instances of the updates are given in [16] and [5].

2.1.2 Noll's Second Order Model

In [12] Noll et al. present a proximal bundle method for nonconvex objective functions. An important ingredient to the method is that not the objective function itself is approximated in the subproblem but a quadratic model of it:

$$\Phi(x,\hat{x}) = \phi(x,\hat{x}) + \frac{1}{2}\langle x-\hat{x}, Q(\hat{x})(x-\hat{x})\rangle \tag{2.1}$$

The first order model $\phi(\cdot,\hat{x})$ is convex and possibly nonsmooth. The second order part

$\frac{1}{2}\langle \cdot - \hat{x}, Q(\hat{x})(\cdot - \hat{x})\rangle$ is quadratic but not necessarily convex.

As the first order part of this model is convex it can be approximated by a cutting plane model just like the objective function in usual convex bundle methods. The subproblem emerging from this approach is

$$\min_{\hat{x}^k+d} m(\hat{x}^k + d) + \frac{1}{2}\left\langle d, Q(\hat{x}^k)d\right\rangle + \frac{1}{2t_k}\|d\|^2$$

where m_k is the cutting plane model (**??**) for the nonsmooth function ϕ.

The matrix $Q(\hat{x})$ itself does not have to be positive definite. In fact the only conditions put on this matrix are that it is symmetric and that all eigenvalues are bounded. We adopt the notation in [11] and write

$$Q(\hat{x}^k) := Q_k = Q_k^\top \quad \text{and} \quad -q\mathbb{I} \prec Q_k \prec q\mathbb{I} \text{ for } q > 0.$$

The notation $A \prec B$ with $A, B \in \mathbb{R}^{n\times n}$ means that the matrix $(B-A)$ is positive definite.

As the matrix Q_k is symmetric it can also be pulled into the stabilization term. The k'th bundle subproblem then is

$$\min_{\hat{x}^k+d\in X} M_k(\hat{x}^k + d) + \frac{1}{2}\left\langle d, \left(Q_k + \frac{1}{t_k}\mathbb{I}\right) d\right\rangle. \tag{2.2}$$

If $W_k = Q_k + \frac{1}{t_k}\mathbb{I}$ is positive definite, this is a variable metric subproblem.

The decomposition of the stabilization term into a curvature approximation and a proximal term makes is easier to reach two goals at the same time:

One the one hand, curvature of the objective can be approximated only under the conditions of the boundedness and symmetry of Q_k. No positive definiteness has to be ensured for convergence. On the other hand the proximal term can be used in the trust region inspired way to make a line search obsolete. As already mentioned in section **??** this is an advantage especially when working with inexact functions where a line search is not useable.

comment on line search and curve search in [5, 6, 16]?

2.1.3 The Descent Measure

Due to the different formulation of the subproblem (2.2) the descent measure δ_k has to be adapted in the varible metric bundle method. In the same way as for (**??**) from the optimality condition

$$0 \in \partial M_k(x^{k+1}) + \partial \mathtt{i}_D(x^{k+1}) + \left(Q_k + \frac{1}{t_k}\mathbb{I}\right) d^k$$

follows that

$$S^k + \nu^k = -\left(Q_k + \frac{1}{t_k}\mathbb{I}\right) d^k, \tag{2.3}$$

S^k and ν^k being the augmented aggregate subgradient and outer normal defined in (1.3) and (1.7) respectively.

From this the model decrease (1.6) can be recovered using (1.5), (1.8) and (2.3):

$$\begin{aligned}
\delta_k &= \hat{f}_k - M_k(x^{k+1}) - \left\langle \nu^k, d^k \right\rangle \\
&= \hat{f}_k - A_k(x^{k+1}) - \left\langle \nu^k, d^k \right\rangle \\
&= C_k - \left\langle S^k + \nu^k, d^k \right\rangle \\
&= C_k + \left\langle d^k, \left(Q_k + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle .
\end{aligned} \tag{2.4}$$

The new δ_k is used in the same way as in algorithm 1.1 for the descent test and stopping conditions.

Because the changes in the algorithm concern only the stabilization and the decrease measure δ_k all other relations that were obtained for the different parts of the model M_k in section 1 are still valid.

2.2 The Variable Metric Bundle Algorithm

The variable b=metric bundle algorithm can now be stated as a varaiation of algorithm 1.1.

same form as Hare algorithm (nullstep)
add Q_k calculation

Algorithm 2.1: Nonconvex Variable Metric Bundle Method with Inexact Information

Select parameters $m \in (0,1), \gamma > 0$ and a stopping tolerance $\texttt{tol} \geq 0$.
Choose a starting point $x^1 \in \mathbb{R}^n$ and compute f_1 and g^1. Set the initial metric matrix $Q_1 = \mathbb{I}$, the initial index set $J_1 := \{1\}$ and the initial prox-center to $\hat{x}^1 := x^1$, $\hat{f}_1 = f_1$ and select $t_1 > 0$.
For $k = 1, 2, 3, \ldots$

1. Calculate

$$d^k = \arg\min_{d \in \mathbb{R}^n} \left\{ M_k(\hat{x}^k + d) + \mathbb{I}_X(\hat{x}^k + d) + \frac{1}{2} \left\langle d, \left(Q_k + \frac{1}{t_k}\mathbb{I} \right) d \right\rangle \right\}.$$

2. Set

$$\begin{aligned} G^k &= \sum_{j \in J_k} \alpha_j^k s_j^k, \\ C_k &= \sum_{j \in J_k} \alpha_j^k c_j^k, \\ \delta_k &= C_k + \left\langle d^k, \left(Q_k + \frac{1}{t_k}\mathbb{I} \right) d^k \right\rangle. \end{aligned}$$

   If $\delta_k \leq \texttt{tol} \rightarrow$ STOP.

3. Set $x^{k+1} = \hat{x}^k + d^k$.

4. Compute f^{k+1}, g^{k+1}.
   If

$$f^{k+1} \leq \hat{f}^k - m\delta_k \quad \rightarrow \text{ serious step}$$

   Set $\hat{x}^{k+1} = x^{k+1}, \hat{f}^{k+1} = f^{k+1}$ and select $t_{k+1} > 0$.
   Calculate Q_k ..., adjust t_k if necessary
   Otherwise $\rightarrow$ nullstep
   Set $\hat{x}^{k+1} = \hat{x}^k, \hat{f}^{k+1} = f^{k+1}$ and choose $0 < t_{k+1} \leq t_k$.

5. Select new bundle index set J_{k+1}, keeping all active elements. Calculate

$$\eta_k = \max \left\{ \max_{j \in J_{k+1}, x^j \neq \hat{x}^{k+1}} \frac{-2e_j^k}{|x^j - \hat{x}^{k+1}|^2}, 0 \right\} + \gamma$$

   and update the model M^k.

2.3 Convergence Analysis

In this section the convergence properties of the new method are analyzed. We do this the same way it is done by Hare et al. in [4].

In the paper all convergence properties are first stated in [4, Lemma 5]. It is then shown that all sequences generated by the method meet the requirements of this lemma which we repeat here for convenience.

Lemma 2.1 ([4, Lemma 5]) *Suppose that the cardinality of the set* $\{j \in J^k | \alpha_j^k > 0\}$ *is uniformly bounded in* k.

(i) If $C^k \to 0$ *as* $k \to \infty$, *then*

$$\sum_{j \in J^k} \alpha_j^k \|x^j - \hat{x}^k\| \to 0 \text{ as } k \to \infty.$$

(ii) If additionally for some subset $K \subset \{1, 2, \dots\}$,

$$\hat{x}^k \to \bar{x}, S^k \to \bar{S} \text{ as } K \ni k \to \infty, \text{ with } \{\eta_k | k \in K\} \text{ bounded,}$$

then we also have

$$\bar{S} \in \partial f(\bar{x}) + B_{\bar{\theta}}(0).$$

(iii) If in addition $S^k + \nu^k \to 0$ *as* $K \in k \to \infty$, *then* $\bar{x}$ *satisfies the approximate stationarity condition*

$$0 \in (\partial f(\bar{x}) + \partial \mathtt{i}_X(\bar{x})) + B_{\bar{\theta}}(0). \tag{2.5}$$

(iv) Finally if f *is also lower-*$\mathcal{C}^1$, *then for each* $\varepsilon > 0$ *there exists* $\rho > 0$ *such that*

$$f(y) \geq f(\bar{x}) - (\bar{\theta} + \varepsilon)\|y - \bar{x}\| - 2\bar{\sigma}, \quad \text{for all } y \in X \cup B_\rho(\bar{x}). \tag{2.6}$$

As neither the stabilization nor the descent test is involved in the proof of Lemma 2.1 it is the same as in [4].

We prove now that also the variable metric version of the algorithm fulfills all requirements of Lemma 2.1. The proof is divided into two parts. The first case covers the case of infinitely many serious steps, the second one considers infinitely many null steps.

For both proofs the following lemma is needed:

Lemma 2.2 *For a symmetric matrix* $A \in \mathbb{R}^{n \times n}$, *a vector* $d \in \mathbb{R}^n$ *and* $\xi > 0$ *the following*

result holds:

$$A \prec \xi\mathbb{I} \Rightarrow Ad < \xi d.$$

The second inequality is meant componentwise.

Proof: As the matrix A is real and symmetric it is orthogonally diagonalizeable. There exist eigenvalues $\lambda_i \in \mathbb{R}, i = \{1, ..., n\}$ and corresponding eigenvectors $v^i \in \mathbb{R}^n, i = \{1, ..., n\}$ that satisfy the equations

$$Av^i = \lambda_i v^i \quad i = \{1, ..., n\}.$$

The eigenvectors v^i generate a basis for $\mathbb{R}^n$ so any vector $d \in \mathbb{R}^n$ can be written as

$$d = \sum_i \alpha_i v^i$$

for $\alpha_i \in \mathbb{R}^n, i = \{1, ..., n\}$.

This yields

$$Ad = A \sum_i \alpha_i v^i = \sum_i \alpha_i \lambda_i v^i. \tag{2.7}$$

Plugging the assumption $A \prec \xi\mathbb{I}$ which is equivalent to $\max_i \lambda_i < \xi$ into (2.7) we get relation (2.3) by

$$Ad < \xi \sum_i \alpha_i v^i = \xi d.$$

□

Theorem 2.3 (c.f.[4, Theorem 6]) *Let the algorithm generate and infinite number of serious steps. Then $\delta_k \to 0$ as $k \to \infty$.*
Let the sequence $\{\eta_k\}$ be bounded. If $\liminf_{k\to\infty} t_k > 0$ then as $k \to \infty$ we have $C_k \to 0$, and for every accumulation point $\bar{x}$ of $\{\hat{x}^k\}$ there exists $\bar{S}$ such that $S^k \to \bar{S}$ and $S^k + \nu^k \to 0$.
In particular if the cardinality of $\{j \in J^k | \alpha_j^k > 0\}$ is uniformly bounded in k then the conclusions of Lemma 2.1 hold.

The proof is very similar to the one stated in [4] but minor changes have to be made due to the different formulation of the nominal decrease δ_k.

Proof: At each serious step we have

$$\hat{f}_{k+1} \leq \hat{f}_k - m\delta_k \tag{2.8}$$

where $m,\ \delta_k > 0$. From this follows that the sequence $\{\hat{f}_k\}$ is nonincreasing. Since $\{\hat{x}^k\}$ lird on yhr compact set X and f is continuous the sequence $f(\hat{x}^k)$ is bounded. With $|\sigma_k| < \bar{\sigma}$ also the sequence $\{f(\hat{x}^k) + \sigma_k\} = \{\hat{f}_k\}$ is bounded. Considering also the fact that $\{\hat{f}_k\}$ is nonincreasing one can conclude that it converges.
From (2.8) follows that

$$0 \leq m \sum_{k=1}^{l} \delta_k \leq \sum_{k=1}^{l} \left(\hat{f}_k - \hat{f}_{k+1}\right),$$

so letting $l \to \infty$,

$$0 \leq m \sum_{k=1}^{\infty} \delta_k \leq \hat{f}_1 - \underbrace{\lim_{k\to\infty} \hat{f}_k}_{\neq \pm\infty}.$$

This yields

$$\sum_{k=1}^{\infty} \delta_k = \sum_{k=1}^{\infty} \left(C^k + \left\langle d^k, \left(Q_k + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle\right) < \infty.$$

Hence, $\delta_k \to 0$ as $k \to \infty$. All quantities above are nonnegative due to positive definiteness of $Q_k + \frac{1}{t_k}\mathbb{I}$ and $C_k \geq 0$ so it also holds that

$$C_k \to 0 \quad \text{and} \quad \left\langle d^k, \left(Q_k + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle \to 0.$$

Finally we need to show that for any accumulation point $\bar{x}$ of the sequence $\{\hat{x}^k\}$ holds $S^k \to \bar{S}$ and $S^k + \nu^k \to 0$ for $K \ni k \to \infty$ and the corresponding subsequence $K \subset \{1, 2, ...\}$. Let $\bar{k}$ denote the last iteration that produces a null step. From $\{\hat{x}^k\}_{k\in K}$ follows that for $k > \bar{k}$ $d^k = \hat{x}^{k+1} - \hat{x}^k \to 0$. As $\liminf_{k\to\infty} t_k > 0$ and the eigenvalues of Q_k are bounded the expression

$$S^k + \nu^k = \left(Q_k + \frac{1}{t_k} I\right) d^k \to 0 \quad \text{for} \quad k \in K$$

because

$$\| \left(Q_k + \frac{1}{t_k} \mathbb{I} \right) d^k \| \leq \underbrace{\left(\|Q_k\| + \frac{1}{t_k} \right)}_{\text{bounded}} \underbrace{\|d^k\|}_{\to 0}.$$

The implication $S^k \to \bar{S}$ for $k \in K$ follows from local Lipschitz continuity of f. By Rademacher's theorem this property yields that on any open set U f is differentiable except on a set U_{nd} of zero Lebesgue measure. As f is also Lipschitz continuous on any closed set containing such an open set U the gradient of f on U is bounded. Let $X \subset U$. By theorem 2.5.1 in [1] the subdifferential of f at any point $x^k \in U$ is the convex hull of the limits of gradients ∇f of f on the set $U \setminus U_{nd}$

$$\partial f(x^k) = \text{conv}\{\lim \nabla f(y) \mid y \to x^k, y \notin U_{nd}\}.$$

This means that also all subgradients on U are bounded. As the subgradient error is assumed to be bounded by $\bar{\theta}$ the approximate subgradients $g^j, j \in J^k$ contained in the bundle are bounded as well. From this follows that also the augmented subgradients $s_j^k = g^j + \eta_k(x^j - \hat{x}^k)$ are bounded because η^k is bounded by assumption and $x^j, \hat{x}^k \in X$. This means that

$$\|S^k\| = \| \sum_{j \in J^k} \alpha_j^k s_j^k \| \leq \sum_{j \in J^k} \|\alpha_j^k\| \underbrace{\|s_j^k\|}_{\leq s \in \mathbb{R}} \leq s \underbrace{\sum_{j \in J^k} \alpha_j^k}_{=1} < \infty \quad \forall k$$

yielding the conclusion $S^k \to \bar{S}$.

□

For the case of infinitely many null steps we need result (31) from [4]. It only depends on the definitions of the augmented linearization error and subgradient and can therefore be taken without any adaption.

Whenever x^{k+1} is as declared a null step, the relation

$$-c_{k+1}^{k+1} + \left\langle s_{k+1}^{k+1}, x^{k+1} - \hat{x}^k \right\rangle \geq -m\delta_k \tag{2.9}$$

holds.

Another relation that is used a few times throughout the proof is the estimate

$$\left\langle \nu^k, d^k \right\rangle \geq 0. \tag{2.10}$$

It follows from the subgradient inequality for the convex function $\mathtt{i}_X$ at the point x^{k+1}. As $\nu^k \in \partial \mathtt{i}_X(x^{k+1})$ it holds $\mathtt{i}_X(y) - \mathtt{i}_X(x^{k+1}) \geq \left\langle \nu^k, y \right\rangle$ for all $y \in X$ and as $d^k = x^{k+1} - \hat{x}^k$ and $x^{k+1}, \hat{x}^k \in X$ it follows

$$\underbrace{\mathtt{i}_X(\hat{x}^k)}_{=0} - \underbrace{\mathtt{i}_X(x^{k+1})}_{=0} \geq -\left\langle \nu^k, \hat{x}^k - x^{k+1} \right\rangle$$

yielding inequality (2.10) above.

Theorem 2.4 (c.f. [4, Theorem 7]) *Let a finite number of serious iterates be followed by infinite null steps. Let the sequence* $\{\eta_k\}$ *be bounded and* $\liminf_{k\to\infty} t_k > 0$.
Then $\{x^k\} \to \hat{x}$, $\delta_k \to 0$, $C_k \to 0$, $S^k + \nu^k \to 0$ *and there exist* $K \subset \{1, 2, ...\}$ *and* $\bar{S}$ *such that* $S^k \to \bar{S^k}$ *as* $K \ni k \to \infty$.
In particular if the cardinality of $\{j \in J^k | \alpha_j^k > 0\}$ *is uniformly bounded in* k *then the conclusions of Lemma 2.1 hold for* $\bar{x} = \hat{x}$.

Proof: Let k be large enough such that $k \geq \bar{k}$, where $\bar{k}$ is the iterate of the last serious step. Let $\hat{x}^{\bar{k}} = \hat{x}$ and $\hat{f}_{\bar{k}} = \hat{f}$ be fixed. As Q_k is not updated in null steps it is also fixed and denoted as $Q = Q_{\bar{k}}$. Define the optimal value of the subproblem (2.2) by

$$\Psi_k := M_k(x^{k+1}) + \frac{1}{2}\left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle. \tag{2.11}$$

It is first shown that the sequence $\{\Psi_k\}$ is bounded above. From definition (1.5) follows

$$A_k(\hat{x}) = M_k(x^{k+1}) - \left\langle S^k, d^k \right\rangle.$$

Using (2.3) for the third equality, (2.10) in the first inequality and (1.9) for the second inequality one obtains

$$\begin{aligned}
\Psi^k + \frac{1}{2}\left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle &= A_k(\hat{x}) + \left\langle S^k, d^k \right\rangle + \left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle \\
&= A_k(\hat{x}) + \left\langle S^k + \langle Q + \frac{1}{t_k}\mathbb{I}, d^k \rangle, d^k \right\rangle \\
&= A_k(\hat{x}) - \left\langle \nu^k, d^k \right\rangle
\end{aligned}$$

$$
\begin{aligned}
&\leq A(\hat{x}) \\
&\leq M_k(\hat{x}) \\
&= \hat{f}.
\end{aligned}
$$

By boundedness of d^k and $Q + \frac{1}{t_k}\mathbb{I}$ this yields that $\Psi_k \leq \hat{f}$, so the sequence $\{\Psi_k\}$ is bounded above. In the next step it is shown that $\{\Psi_k\}$ is increasing. By noting that $x^{k+2} = \hat{x} + d^{k+1}$, as the proximal center does not change in the null step case, we obtain

$$
\begin{aligned}
\Psi_{k+1} &= M_k(x^{k+2}) + \frac{1}{2}\left\langle d^{k+1}, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle \\
&\geq A_k(\hat{x} + d^{k+1}) + \frac{1}{2}\left\langle d^{k+1}, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle \\
&= M_k(x^{k+1}) + \left\langle S^k, d^{k+1} - d^k\right\rangle + \frac{1}{2}\left\langle d^{k+1}, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle \\
&= M_k(x^{k+1}) + \left\langle -\langle Q + \frac{1}{t_k}\mathbb{I}, d^k\rangle - \nu^k, d^{k+1} - d^k\right\rangle + \frac{1}{2}\left\langle d^{k+1}, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle \\
&= \Psi_k - \frac{1}{2}\left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k\right\rangle + \frac{1}{2}\left\langle d^{k+1}, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle \\
&\quad - \left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right)\left(d^{k+1} - d^k\right)\right\rangle - \left\langle \nu^k, d^{k+1} - d^k\right\rangle \\
&= \Psi_k + \frac{1}{2}\left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k\right\rangle + \frac{1}{2}\left\langle d^{k+1}, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle \\
&\quad - \left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^{k+1}\right\rangle - \left\langle \nu^k, d^{k+1} - d^k\right\rangle \\
&\geq \Psi_k + \frac{1}{2}\left\langle \left(d^{k+1} - d^k\right), \left(Q + \frac{1}{t_k}\mathbb{I}\right)\left(d^{k+1} - d^k\right)\right\rangle \\
&= \Psi_k + \frac{1}{2}\underbrace{\|d^{k+1} - d^k\|_{Q+\frac{1}{t_k}\mathbb{I}}}_{\geq 0}.
\end{aligned}
\tag{2.12}
$$

Here the first inequality comes from (1.9) and the fact that $t_{k+1} \leq t_k$ for null steps. The second equality follows from (1.5), the fourth equality by (2.3) and (2.11) and the last inequality holds by (2.10).

As Q is fixed in null steps and $\liminf_{k\to\infty} t_k > 0$ the sequence $\{\Psi_k\}$ is increasing and bounded from above. It is therefore convergent.

Looking again at (2.12) and taking into account that $1/t_k \geq 1/t_{\bar{k}}$ in the null step case we have

$$\begin{aligned}\Psi_{k+1} - \Psi_k &\geq \frac{1}{2}\|d^{k+1} - d^k\|_{Q+\frac{1}{t_k}\mathbb{I}} \\ &= \frac{1}{2}\left\langle \left(d^{k+1} - d^k\right), \left(Q + \frac{1}{t_k}\mathbb{I}\right)\left(d^{k+1} - d^k\right)\right\rangle \\ &\geq \frac{1}{2}\left\langle \left(d^{k+1} - d^k\right), \left(Q + \frac{1}{t_{\bar{k}}}\mathbb{I}\right)\left(d^{k+1} - d^k\right)\right\rangle \\ &= \frac{1}{2}\|d^{k+1} - d^k\|_{Q+\frac{1}{t_{\bar{k}}}\mathbb{I}}.\end{aligned}$$

As the sequence $\{\Psi_k\}$ is converging this yields

$$|\Psi_{k+1} - \Psi_k| \to 0 \quad \Rightarrow \quad \|d^{k+1} - d^k\| \to 0 \text{ for } k \to \infty \tag{2.13}$$

due to the equivalence of norms.

By the last line in (2.4) and the fact that $\hat{f} = M_k(\hat{x})$ we have

$$\begin{aligned}\hat{f} &= M_k(\hat{x}) + \delta_k - C_k - \left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k\right\rangle \\ &= M_k(\hat{x}) - \hat{f} - M_k(x^{k+1}) + \delta_k - \left\langle S^k, d^k\right\rangle - \left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k\right\rangle \\ &= \delta_k + M_k(\hat{x} + d^k) + \left\langle \nu^k, d^k\right\rangle \\ &\geq \delta_k + M_k(\hat{x} + d^k),\end{aligned}$$

where the second equality is by (1.8), the third holds because of relation (2.3) and the last inequality is given by (2.10). Therefore

$$\delta^{k+1} \leq \hat{f} - M_{k+1}(\hat{x} + d^{k+1}). \tag{2.14}$$

By assumption (1.10) on the model, written for $d = d^{k+1}$,

$$-\hat{f}_{k+1} + c_{k+1}^{k+1} - \left\langle s_{k+1}^{k+1}, d^{k+1}\right\rangle \geq -M_{k+1}(\hat{x} + d^{k+1}).$$

In the null step case it holds $\hat{f}_{k+1} = \hat{f}$ so combining condition (2.9) and the inequality above, one obtains that

$$m\delta_k + \left\langle s_{k+1}^{k+1}, d^k - d^{k+1} \right\rangle \geq \hat{f} - M_{k+1}(\hat{x} + d^{k+1}).$$

In combination with (2.14) this yields

$$0 \leq \delta_{k+1} \leq m\delta_k + \left\langle s_{k+1}^{k+1}, d^k - d^{k+1} \right\rangle. \tag{2.15}$$

For the next step Lemma 3 and the following corollary from [13] are used. They state that for

$$u_{k+1} \leq qu_k + a_k, \quad q < 1, \quad a_k \geq 0, \quad a_k \to 0 \text{ and } u_k \geq 0$$

holds $u_k \to 0$.

Relation (2.15) also holds in the form

$$\delta_{k+1} \leq m\delta_k + |\left\langle s_{k+1}^{k+1}, d^k - d^{k+1} \right\rangle|.$$

For this inequality we can identify $u_k = \delta_k \geq 0$, $q = m \in (0,1)$ and $a_k = |\left\langle s_{k+1}^{k+1}, d^k - d^{k+1} \right\rangle| \geq 0$. To show that $a_k \to 0$ recall (2.13) and that the augmented subgradient s_{k+1}^{k+1} is bounded due to local Lipschitz continuity of f and boundedness of $\{\eta_k\}$ by the same argumentation as in the case of infinitely many serious steps.

The lemma then gives that

$$\lim_{k\to\infty} \delta_k = 0.$$

From formulation (2.4) of the model decrease follows that $C_k \to 0$ as $k \to \infty$. Since $Q + \frac{1}{t_k}\mathbb{I} \succ \xi\mathbb{I}$ due to $\liminf_{k\to\infty} t_k > 0$ and the bounded eigenvalues of Q we have

$$\xi\|d^k\|^2 \leq \left\langle d^k, \left(Q + \frac{1}{t_k}\mathbb{I}\right) d^k \right\rangle \to 0$$

This means that $d^k \to 0$ for $k \to \infty$ and therefore $\lim_{k\to\infty} x^k = \hat{x}$. It also follows that $\|S^k + \nu^k\| \to 0$ as $k \to \infty$. Passing to some subsequence if necessary we can conclude that S^k converges to some $\bar{S}$ and as $\hat{x}^k = \bar{x}$ for all k all requirements of Lemma 2.1 are fulfilled.

□

Remark: In case the matrix Q_k was also updated in null steps the proof still holds as long as the assumptions on boundedness of Q_k and positive definiteness of $Q_k + \frac{1}{t_k}\mathbb{I}$ are still valid.

Remark: All results deduced in section 1.2 are still valid for this algorithm as they do not depend on the kind of stabilization used.

2.4 Updating the Metric

In [12] and [11] it is not specified how the matrices Q_k are chosen. For convergence it is necessary that the eigenvalues of Q_k are bounded. Additionally the matrix $Q_k + \frac{1}{t_k}\mathbb{I}$ has to be positive definite. Here we present some possibilities to update the metric matrix Q_k that fulfill both conditions.

Both updates are based on the usual BFGS-update formula (named after Broyden, Goldfarb, Fletcher and Shanno)

$$Q_{k+1} = Q_k + \frac{y^k {y^k}^\top}{\langle y^k, d^k \rangle} - \frac{Q_k d^k (Q_k d^k)^\top}{\langle d^k, Q_k d^k \rangle}. \tag{2.16}$$

Usually y^k is defined as the difference of the last two gradients of f. To adapt the formula to the nondifferentiable case the difference $y^k := g^{k+1} - g^k$ of two subgradients of f is taken instead as proposed in [2]. The starting matrix $Q_1 = \mathbb{I}$.

By definition the BFGS update is symmetric. To assure boundedness of the matrix Q_{k+1} the updates can be manipulated in the following ways:

2.4.1 Scaling of the Whole Matrix

A simple way to keep the absolute value all of eigenvalues of the constructed matrix Q_k below some threshold $0 < q < \infty$ is to scale the whole matrix down as soon as the absolute value of one eigenvalue is larger than this number. To do this define $\lambda_{max} := \max\{|\lambda_i^k| \mid \lambda_i^k \text{ is eigenvalue of } Q_k|\}$. If $\lambda_{max}^k > q$, set $Q_k = \frac{q}{\lambda_{max}^k} Q_k$. This way the absolute value of all eigenvalues is always smaller or equal to q. An advantage of this method is besides its simplicity that by scaling the whole matrix the ratio of the eigenvalues of Q_k is preserved. Scaling of Q_k corresponds to shrinking the whole quadratic function and in this way also the 'ratio of curvature' at different points of the graph stays the same.

2.4.2 Adaptive Scaling of Single Eigenvalues

This second method is motivated by the following observation: The variable metric bundle algorithm is to be used for nonsmooth functions. This means that the objective function has some kinks. For locally Lipschitz functions the number of kinks is finite and in between the kinks the function is smooth. This means, that there is indeed curvature information present in the smooth parts of the functions. At the kinks however the curvature is not defined. Taking a look at the one-sided differential quotient for $x \in \mathbb{R}$ shows that it diverges at such points: Let a kink be at $x_{\text{kink}} \in \mathbb{R}$ and the left sided limiting value of the derivative be $f'(x_{\text{kink}}) = a$ the right-sided one $f'(x_{\text{kink}}) = b \neq a$. Because f was assumed to be locally Lipschitz both limits exist and are finite. The following quotient can then be stated:

$$\lim_{h \searrow 0} \frac{f'(x_{\text{kink}} + h) - f'(x_{\text{kink}})}{h} = \frac{\overbrace{f'(x_{\text{kink}} + h)}^{\to b} - a}{h} = \pm\infty,$$

the sign depending on if $a > b$ or vice versa.

In more dimensions this is the same for the components of the Hessian corresponding to the direction where the kink occurs. This supports also to the intuitive thought that at a kink the slope changes 'infinitly fast'. Numerically the BFGS-update (2.16) can result in very large values for the entries of Q_k corresponding to points near the kink.

On the other hand due to the local Lipschitz property the slope of the objective function is always finite on closed sets. This means that there exists a neighborhood $B_\varepsilon(x_{\text{kink}})$ where the function f behaves similar to the scaled modulus $a|\cdot|$, $a \in \mathbb{R}$, in the direction perpendicular to the kink. Therefore in this neighborhood almost no curvature is present.

Summarized this means that on the one hand, the matrix Q_k should be close to zero in the components representing the directions perpendicular to the kink as soon as the iterates approach x_{kink}. But contrary to that the method that constructs Q_k can give very high values for those components.

The idea is now to scale only those eigenvalues of Q_k that are especially large. To do this calculate all eigenvalues λ_i^k of Q_k and additionally a decomposition of the matrix $A \cdot D \cdot B$ of the matrix Q_k, where $D \in \mathbb{R}^{n \times n}$ is matrix with the eigenvalues of Q_k directly accessible, for example a diagonal matrix with the eigenvalues on the diagonal, and $A, B \in \mathbb{R}^{n \times n}$ are the transformation matrices. All eigenvalues of Q_k that are larger than q are scaled and replaced in the matrix D. The bounded version of Q_k is obtained by transforming

D back into the full matrix with help of the transformation matrices A and B.

The metric matrix Q_k is a symmetric real matrix. This means it is always orthogonally diagonalizable. An eigenvalue decomposition $Q_k = A \cdot D \cdot A^\top$ is available. The diagonal matrix D has the eigenvalues of Q_k on its diagonal and the transformation matrix A contains the corresponding eigenvectors. Let $\bar{D}$ denote the matrix with the scaled eigenvalues on the diagonal. The matrix A is not changed. This means that $\bar{Q}_k = A \cdot \bar{D} \cdot A^\top$ has the same eigenvectors as the original update Q_k but bounded eigenvalues.

The above two methods were tested in practice and the results of the algorithm are shown in section 2.5. We also compare them to a hybrid method where the first approach is used for the updates and the matrix is additionally scaled by the stepsize such that the final metric matrix is $Q_k = \frac{1}{k}\bar{Q}_k$ with $\bar{Q}_k$ being the scaled BFGS update suggested first. This way the method starts out as the variable metric method but becomes more equal to the proximal bundle method 1.1 as the algorithm continues.

Remark: There appear many parameters to control the scaling of the metric update. Although these parameters were not especially tuned in this thesis it was observed that they have a considerable impact on the convergence speed of the method also depending on the objective function used. This has to be kept in mind when implementing the method in practice.

2.4.3 Other Updating Possibilities

There are certainly many other possibilities to update the metric Q_k. A third variation based on BFGS-updates is the limited memory update suggested in [10]. If the update is skipped whenever $\|\rho s s^\top\| = \frac{\|d^k\|}{\|y^k\|} > q$ the matrix Q_k stays bounded. This strategy is also supported by the fact that if $\frac{\|d^k\|}{\|y^k\|} > q$ the change in the subgradient relative to the step size is rather small indicating that the current iterate lies within a region with only small changes in curvature. In such regions the update can be skipped. It is also possible to alter the updates presented above by a special choice of the subgradients. For example trying to compute the directional derivative or using more information by considering more subgradients of the bundle.

Another updating method is using the symmetric-rank-1 (SR1) update

$$Q_k = Q_{k-1} + \frac{(y^k - Q_k d^k)(y^k - Q_k d^k)^\top}{\langle y^k - Q_k d^k, d^k \rangle}.$$

Boundedness can be assured in the same as for the normal BFGS update.

Finally the strategies to measure the need of scaling of the matrices to ensure boundedness are diverse. Here it could be interesting to consider the change in the matrix Q_k relative to Q_{k-1} instead of using the absolute values of the eigenvalues as a threshold. This is however hardly possible if the eigenvalues themselves are taken as a measure as they lack of an intrinsic order. This makes it hard to find the corresponding eigenvalues λ_i^{k-1} and λ_i^k in consecutive updates to compute the change between them.

In higher dimensions updating the matrix Q_k can be costly. This is one of the reasons why it is not updated in null steps. Also in null steps the proximal center stays the same, so it can be assumed that not much curvature information can be gained when updating during such steps. Still updating in null steps has the advantage of making use of the additional subgradient information provided in those steps. In [2] where the metric matrix is updated also in null steps a BFGS update is used in serious steps and the less costly SR1 update in null steps.

Remark: Bounding the eigenvalues of Q_k by $q \in \mathbb{R}$ also assures that t_k can be bounded from below without impairing positive definiteness of the matrix $Q_k + \frac{1}{t_k}\mathbb{I}$. This can be done by setting $t_{min} = \frac{1}{q} - \xi$ for a small positive constant ξ.

As a last remark on this topic we want to say that although in this thesis the adaption of update strategies originally developed to be used with gradients seems to work in the presented setting also with subgradients this does not always have to be the case. Although it is argued for example in [7] that locally Lipschitz functions are differentiable almost everywhere and with an adequate linesearch it is improbable to arrive at an iterate that is a nondifferentiable point of the objective function this can still happen. Especially if such a linesearch is not used like in the algorithm presented here. So despite the promising practical behavior this area is still open to research.

2.5 Numerical Testing

To compare the proximal bundle algorithm 1.1 with its variable metric variant algorithm 2.1 it is tested on some academic test functions and on a set of lower-$\mathcal{C}^2$ functions in different dimensions. The tests are done with the following parameters given in [4]: $m = 0.05$, $\gamma = 2$ and $t_0 = 0.1$. The chosen stopping tolerance is $\texttt{tol} = 10^{-6}$. If the algorithms do not meet the stopping condition after $250n$ steps for $x \in \mathbb{R}^n$, they are terminated. Contrary to [4] the stopping test is taken as given in the algorithm and the tolerance not multiplied by $1 + \hat{f}_k$. The proximity control parameters κ_- and κ_+ from (**??**) and (**??**) respectively are chosen as $\kappa_- = 0.8$ and $\kappa_+ \in \{1.2, 2\}$. When the bundle

is updated at the end of each iteration the additionally to the newly computed iterate the current prox-center and all elements that have corresponding Lagrange multiplier $\alpha_j^k > 10^{-15}$ are kept in the bundle.

In the metric matrix updates the threshold q is chosen 10^{-8}. For the adaptive variant of the update eigenvalues that are larger than q are set to $q/10$.

The algorithms are abbreviated in the legends of the plots as 'Bundle Nonconv Inex' for the proximal bundle algorithm 1.1 and 'Variable Metric BFGS' and 'Variable Metric BFGS Adaptive' for the variable metric bundle algorithm 2.1 using the scaled update and the adaptive eigenvalue scaling respectively.

To test the performance for inexact function and subgradient values different types of noise are introduced. This is done by adding randomly generated elements with norm less or equal to σ_k and θ^k to the exact values $f(x^{k+1})$ and $g(x^{k+1})$ respectively.

Five different forms of noise are tested:

- N_0: No noise, $\bar{\sigma} = \sigma_k = 0$ and $\bar{\theta} = \theta^k = 0$ for all k,
- $\mathrm{N}_\mathrm{c}^{\mathrm{f,g}}$: Constant noise, $\bar{\sigma} = \sigma_k = 0.01$ and $\bar{\theta} = \theta^k = 0.01$ for all k,
- $\mathrm{N}_\mathrm{v}^{\mathrm{f,g}}$: Vanishing noise, $\bar{\sigma} = 0.01, \sigma_k = \min\{0.01, \|x^k\|/100\}$ and $\bar{\theta} = 0.01, \theta_k = \min\{0.01, \|x^k\|/100\}$ for all k,
- $\mathrm{N}_\mathrm{c}^{\mathrm{g}}$: Constant subgradient noise, $\bar{\sigma} = \sigma_k = 0$ and $\bar{\theta} = \theta_k = 0.01$ for all k and
- $\mathrm{N}_\mathrm{v}^{\mathrm{g}}$: Vanishing subgradient noise, $\bar{\sigma} = \sigma_k = 0$ and $\bar{\theta} = 0.01, \theta_k = \min\{0.01, \|x^k\|/100\}$ for all k.

The exact case is used for comparison. The constant noise forms represent cases where the inexactness is outside of the optimizer's control. The vanishing noise forms represent cases where the noise can be controlled but the mechanism is considered expensive, so it is only used when approaching the minimum. The two forms of subgradient noise represent the case where the subgradient is approximated numerically.

To compare the performance of the different methods the accuracy is measured by

$$\text{accuracy} = |\log_{10}(\hat{f}_{\bar{k}})|.$$

$\hat{f}_{\bar{k}}$ being the current $\hat{f}_k$ when the algorithm stops.

2.5.1 Academic Test Examples

For the comparison in this section the proximal bundle method and the variable metric method with the two BFGS update rules for Q_k presented in section 2.4 are used.

To explore the benefit of the matrix Q_k the algorithms 1.1 and 2.1 are tested on a smooth and a nonsmooth version of a badly conditioned parabola. The smooth test function is

$$p(x) : \mathbb{R}^2 \to \mathbb{R}, \quad x \mapsto \langle x, Ax \rangle ,$$

where the matrix is chosen as $A = \begin{pmatrix} 1 & 0 \\ 0 & 50 \end{pmatrix}$. The condition number of this matrix is $\kappa_A = \frac{\lambda_{max}}{\lambda_{min}} = 50$, where $\lambda_{min}, \lambda_{max}$ are the smallest and largest eigenvalue of A respectively. From smooth optimization it is known that gradient descent methods have a rather poor convergence rate for such badly conditioned matrices (c.f. Chapter 7.4 in [9]). Figure 1 shows the sequences of serious iterates resulting from the two algorithms on the contour lines of the parabola. On the left the complete sequence is depicted. The plot on the right shows a detail of the left figure near the minimum of the objective. As the descent direction taken in algorithm 1.1 is an aggregate subgradient and second order information is only provided by the stabilization term $\frac{1}{t_k}\|d\|^2$ we can see a zig-zagging behavior of the sequence for the parabola in figure 1. Contrary to that the sequence of serous iterates provided by algorithm 2.1 can take advantage of the second order information provided by Q_k. It approaches the minimum almost in a straight line. The difference in behavior of the two algorithms is especially visible on the detail plot of 1 that shows the situation near the minimum: The proximal bundle algorithm needs a lot of steps circling around the minimum whereas the variable metric algorithm approaches the minimum directly. The resulting advantage of this behavior is the smaller number of steps needed by the variable metric algorithm.

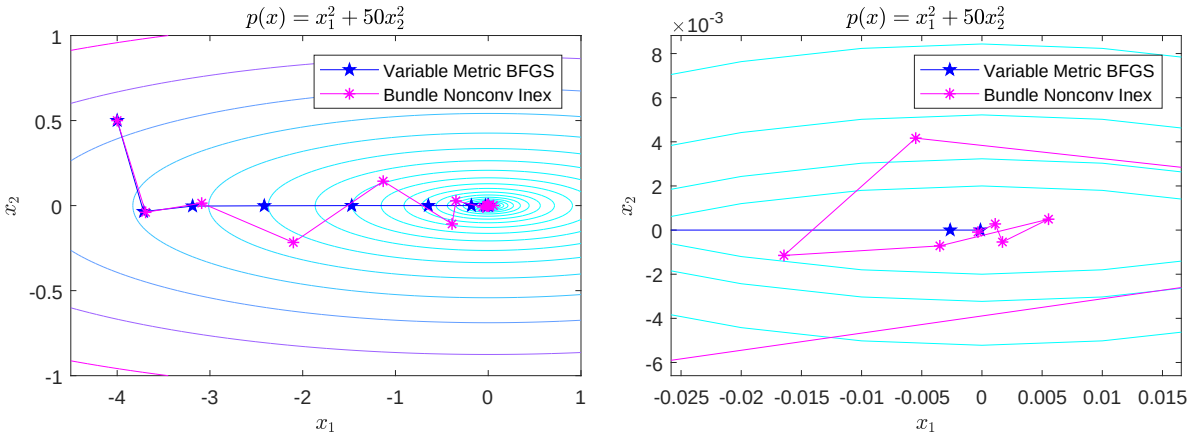

Figure 1: *Sequences of serious steps constructed by the proximal bundle algorithm and the variable metric algorithm respectively on the level lines of parabola p. The right image is a detail of the plot on the left.*
Step size parameter: $\kappa_+ = 2$ *for both algorithms.*

The second test function is a nonsmooth version of the above parabola. The function is given by

$$p_n(x) : \mathbb{R}^2 \to \mathbb{R}, \quad x \mapsto \left\langle \frac{1}{2}x, Ax \right\rangle + \frac{1}{2}|x_1| + 25|x_2|.$$

Due to the kink along the x_1-axis the curvature information supplied by Q_k is less reliable than for the smooth parabola. Figure 2 shows the sequences constructed by the two algorithms. Still the sequence provided by the variable metric algorithm does less zig-zagging than the one coming from the proximal bundle algorithm. It is interesting to note, that the sequence provided by the proximal bundle algorithm is the same for both functions. This is not the case for the sequence generated by the metric bundle algorithm because the second order information of the two objective functions is different.

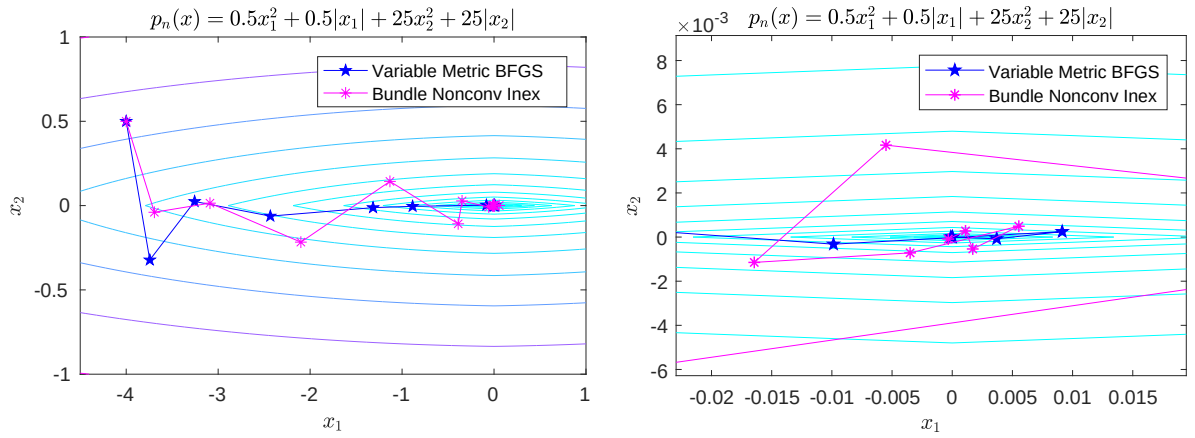

Figure 2: *Sequences of serious steps constructed by the proximal bundle algorithm and the variable metric algorithm respectively on the level lines of the nonsmooth quadratic function p_n. The right image is a detail of the plot on the left.*
Step size parameter: $\kappa_+ = 2$ for both algorithms.

The bar plots in figures 3 and 4 compare the accuracy of the solution and the number of steps that is needed by the different algorithms for the various noise forms. Here the nonconvex proximal bundle algorithm is compared to both variants of the variable metric method.

To address the random nature of the noise the tests are performed 20 times and the results averaged. The number of steps is rounded to end up with integers.

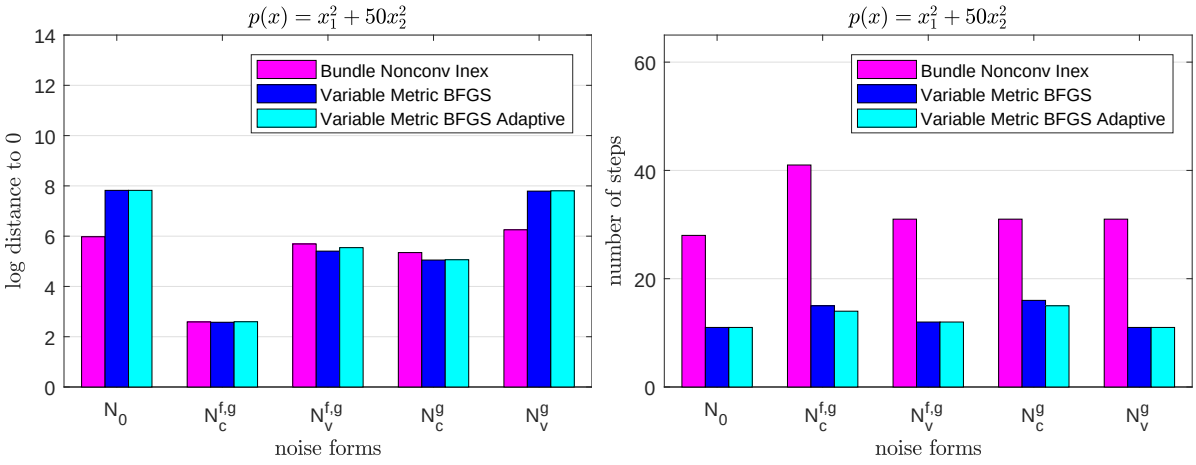

Figure 3: *Left: Accuracy of the solution computed by the different versions of the variable metric bundle algorithm compared to the proximal bundle algorithm for the parabola p under different form of noise.*
Right: Comparison of the number of steps for the three algorithms.
Step size parameters: $\kappa_+ = 1.2$ for the proximal bundle method and $\kappa_+ = 2$ for the variable metric algorithm.

In the smooth case one can see that the accuracy of the two algorithms is comparable. In the case where no noise is present and in the last case, which is the case with the least noise the variable metric algorithm solve more accurately but for the proximal bundle algorithm the actually computed optimal value is still above the chosen tolerance of 10^{-6}. In the cases of the more involved noise the accuracy is less.

A significant difference can be seen between the needed number of steps of the different algorithms. Here the variable metric versions of the bundle method can take advantage of the curvature information and the fact that for the smooth parabola the BFGS update approximates the Hessian matrix very well. The difference in steps between the two update variants of the variable metric algorithm is neglectable and could also be present due to the random noise. This is what we expect as the scaling mechanism should not be invoked in the smooth case.

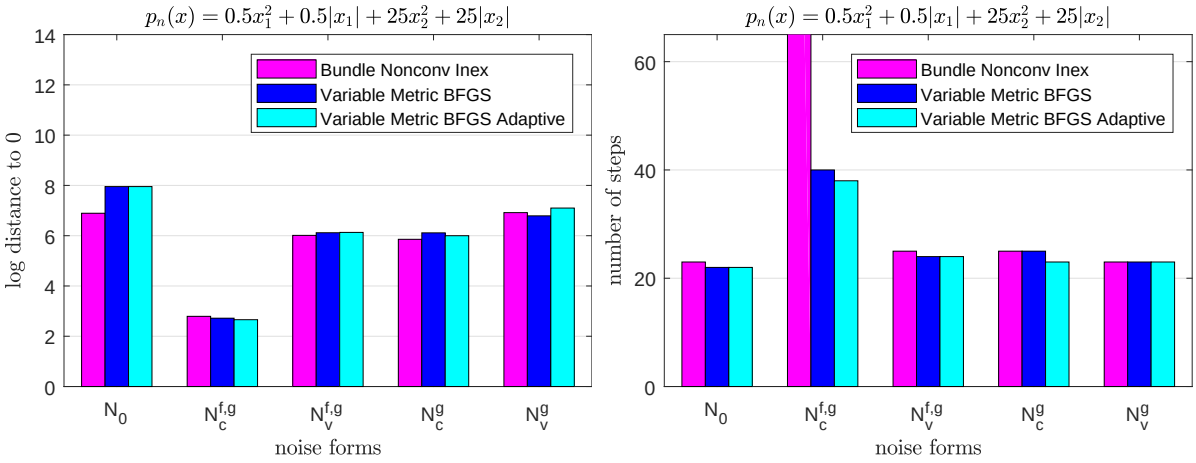

Figure 4: *Left: Accuracy of the solution computed by the different versions of the variable metric bundle algorithm compared to the proximal bundle algorithm for the nonsmooth quadratic function p_n under different form of noise.*
Right: Comparison of the number of steps for the three algorithms. The bar for the number of steps in the case of constant noise is cropped.
Step size parameters: $\kappa_+ = 1.2$ for the proximal bundle method and $\kappa_+ = 2$ for the variable metric algorithm.

In the nonsmooth case (shown in figure 4) the accuracy of all algorithms is very similar. The difference in the number of steps is now very small as well. Only in the case of constant noise the proximal bundle algorithm performs rather badly. Here the number of steps is extremely large (over 250) in order to gain the same accuracy as the other algorithms. The difference between the two update versions of the variable metric algorithm are still very small. It seems that the different scaling strategies have only a minor influence on the algorithm for this kind of objective function. Other tests showed that for

example the choice of the step size updating parameters κ_+, κ_- have a lot more influence on the algorithm than the tested updating strategies. refer to appendix

2.5.2 Test Examples in Higher Dimensions

For the second test, which involves testing the performance of the different algorithms in different dimensions of the minimizer, the Ferrier polynomials are chosen as objective functions. These nonsmooth and nonconvex functions have already been used in [3] and [4]. The polynomials are constructed in the following way:

For $i = 1, ..., n$ we define

$$h_i : \mathbb{R}^n \to \mathbb{R}, \quad h(x) = (ix_i^2 - 2x_i) + \sum_{j=1}^{n} x_j.$$

These functions are used to define

$$\begin{aligned}
f_1(x) &:= \sum_{i=1}^{n} |h_i(x)|, \\
f_2(x) &:= \sum_{i=1}^{n} (h_i(x))^2, \\
f_3(x) &:= \max i \in [1, ..., n] |h_i(x)|, \\
f_4(x) &:= \sum_{i=1}^{n} |h_i(x)| + \frac{1}{2}|x|^2, \\
f_5(x) &:= \sum_{i=1}^{n} |h_i(x)| + \frac{1}{2}|x|.
\end{aligned}$$

The graphs of the Ferrier polynomials for $x \in \mathbb{R}^2$ are shown in figure 5.

Ferrier polynomials are nonconvex, nonsmooth (except for f_2) and lower-$\mathcal{C}^2$. They all have 0 as a global minimizer [3]. The compact constraint set is $X = \{d \in \mathbb{R}^n | d_i + \hat{x}_i^k \leq 10, i = 1, ..., n\}$.

The five test functions f_1 to f_5 are optimized for the dimensions $n = 2, 3, ..., 15, 20, 25, 30, 40, 50$. The starting value for each test problem is $x^1 = [1, \frac{1}{4}, \frac{1}{9}, ..., \frac{1}{n^2}]^\top$.

For the tests the step size of all algorithms is updated with $\kappa_+ = 1.2$, which provided better results for these specific test functions.

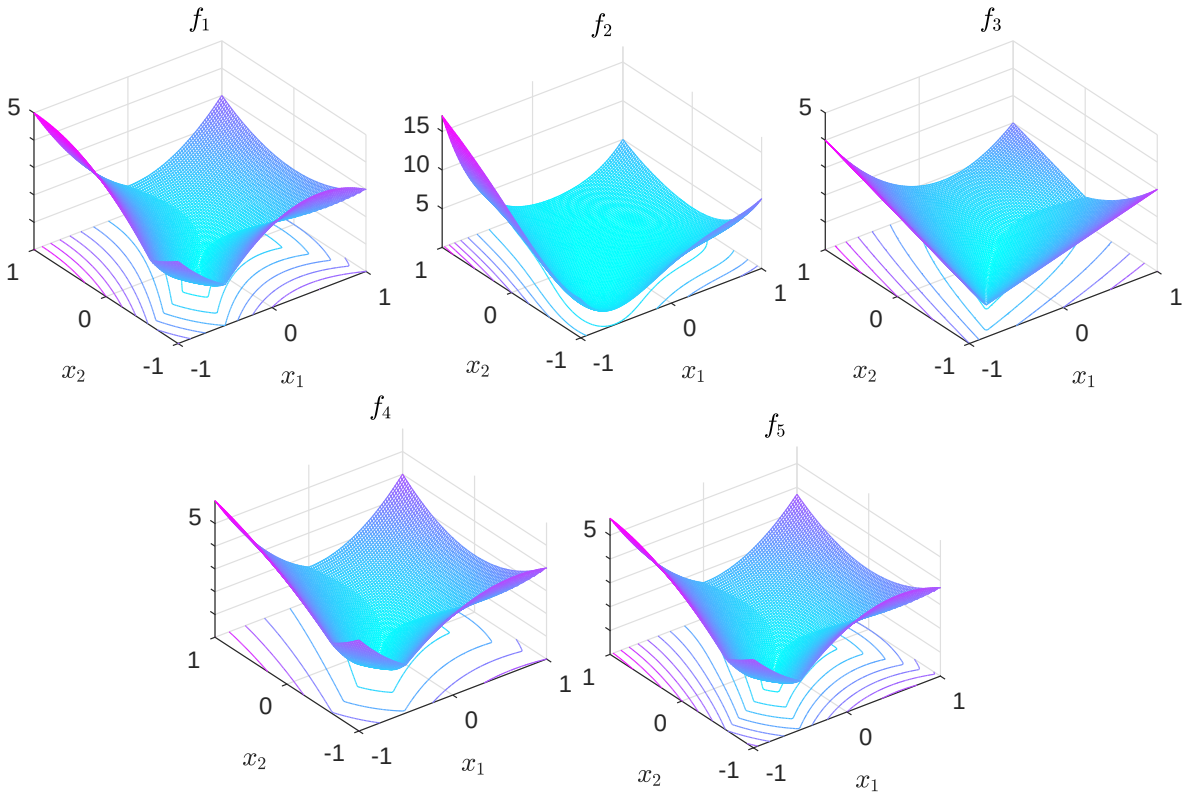

Figure 5: *Graphs of the testfunctions f_1 to f_5 for $x \in \mathbb{R}^2$*

Figures 6 to 8 and 10 to **??** in the appendix the absolute value of the logarithm of the achieved accuracy and the number of steps are shown for the proximal bundle method and two versions of the variable metric method.

Figure 6 shows the situation if no noise is present and can be seen as a benchmark for the other noise forms. It is clearly visible that the desired accuracy of 10^{-6} is not always achieved by the different algorithm. A reason for this is that the objective functions have several local minima, where the algorithms can get stuck. It seems that this happens more seldom to the proximal bundle algorithm than the variable metric method.

For the Ferrier polynomials the proximal bundle algorithm needs significantly less steps than the variable metric algorithm. It can also be observed that the cases where the variable metric algorithm is stuck in a local minimum, the number of steps needed rises significantly. This is not the case for the proximal bundle algorithm.

The performance of the two variants of the two versions of the variable metric method are similar. It seems however as if the adaptive version performed slightly better in terms of the number of steps used.

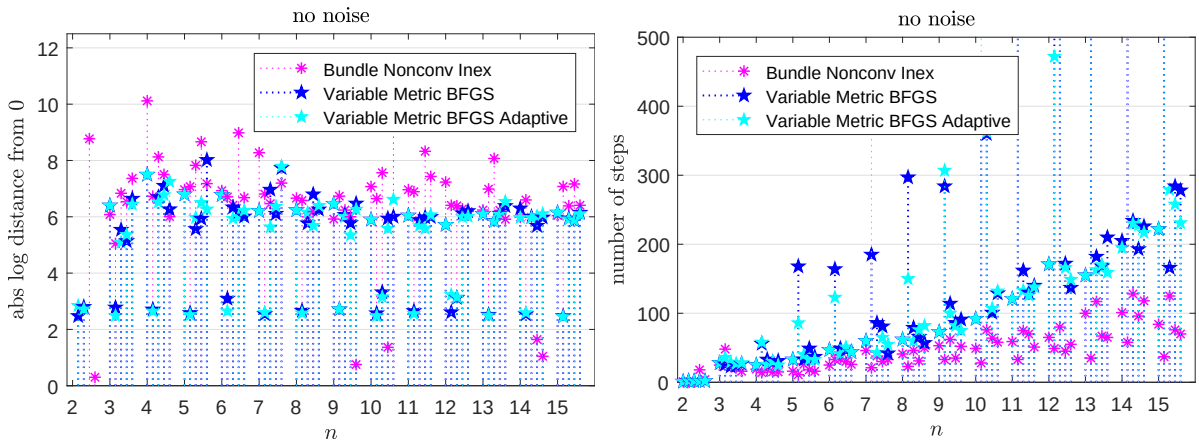

Figure 6: *Comparison of accuracy and number of steps for the proximal bundle algorithm and the variable metric bundle algorithm in the case of no noise.*

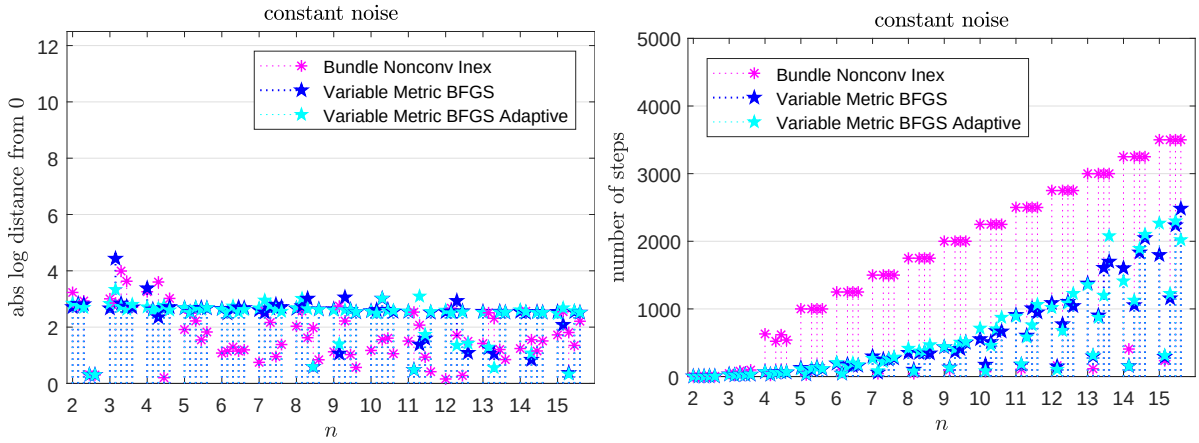

Figure 7: *Comparison of accuracy and number of steps for the proximal bundle algorithm and the variable metric bundle algorithm in the case of constant noise*

In the case of constant noise the variable bundle methods perform better than the proximal version. They are more stable in the achieved accuracy and need considerably less steps than the other method.

For the other noise forms the three algorithms perform similar in terms of the accuracy but the variable bundle methods need consistently more steps. The only exception from this is case of vanishing noise. Here the proximal bundle method needs extremely many more steps than the variable metric bundle method to optimize function f_3. This shows that the performance of the different algorithms depend on both the form of noise and the specific objective function.

The plots for the higher dimensional data $x \in \mathbb{R}^n$ for $n = 20, 25, 30, 40, 50$ are included

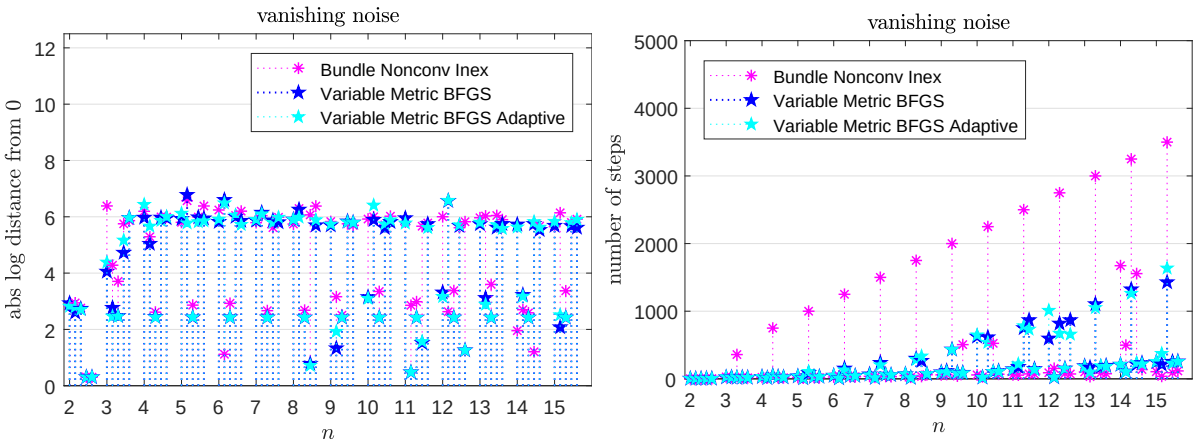

Figure 8: *Comparison of accuracy and number of steps for the proximal bundle algorithm and the variable metric bundle algorithm in the case of vanishing noise*

in the appendix (figures **??** to **??**). Also in higher dimensions the algorithms achieve a similar accuracy. The number of steps needed for convergence in generally higher, but still the proximal bundle method needs less steps in most situations. In the cases where the is noise fut on the function value, the algorithms almost always stop because the maximum number of steps is reached. The only exception is the smooth function f_2. Here the variable metric methods perform a lot better than the proximal bundle algorithm in terms of the number of steps.

Finally the influence of the step size updating parameter κ_+ is shown and the performance of the hybrid method. This last method, denoted by 'Variable Metric BFGS, k-scaled' in the figures, uses the scaled BFGS update for the metric matrix Q_k and then scales this matrix again by the step size. This means the final matrix is $Q_k = \frac{1}{k}\bar{Q}_k$ if $\bar{Q}_k$ denotes the matrix after the scaled BFGS update, lowering the influence of the metric matrix in each serious step. In this way the method starts out as the variable metric method and then behaves more and more like the proximal bundle method.

The algorithms used for the comparison of the different κ_+ are endowed with the scaled BFGS update. The parameters $\kappa_+ = 1.2$ and $\kappa_+ = 2$ are compared.

Here only the exact case for the lower dimensions is depicted in figure 9 and **??**. The other figures **??** to **??** are placed in the appendix.

One can see that contrary to the academic test example, where the choice $\kappa_+ = 2$ gives better results for the number of steps, here the parameter $\kappa_+ = 1.2$ performs better. In the case of constant noise, the numbers of steps are similar. As expected the accuracy of the two methods is very similar, because only one parameter is changed. The number of

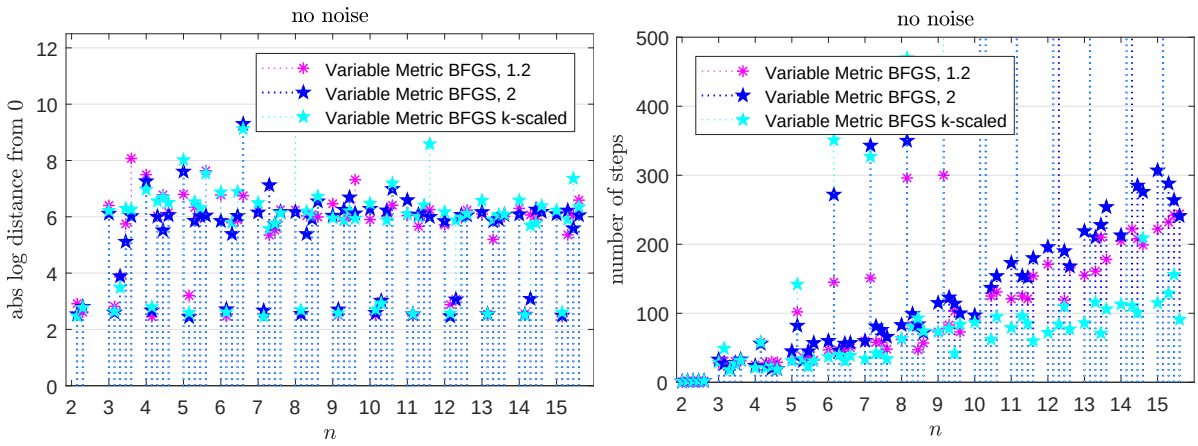

Figure 9: *Influence of the step size updating parameter $\kappa_+ = 1.2$ and $\kappa_+ = 2$ and performance of the hybrid method in the exact case. The reached accuracy is depicted on the left, the needed number of steps on the right.*

steps shows however that parameter tuning can be useful. Here it is important to keep in mind that the optimal parameter depends on the objective function and the noise form.

The performance of the hybrid method is, as can be expected, similar to the proximal bundle method. As the scaling is quite strong the influence of the metric matrix decreases rather quickly. This means that in cases where the proximal bundle method performed better, the same is true for the hybrid method. The increase in the number of steps is only minor. Likewise in the case of constant noise for example, where the proximal bundle method needed a lot of steps, these steps are also needed by the hybrid method. Two advantages of the hybrid method still come into play for this noise forms: The significant decrease in the number of steps starts in only in higher dimensions, where the total number of steps grows higher. Here a 'slower' scaling could yield even better numbers. The other advantage is the more stable accuracy. This holds true for all dimensions.

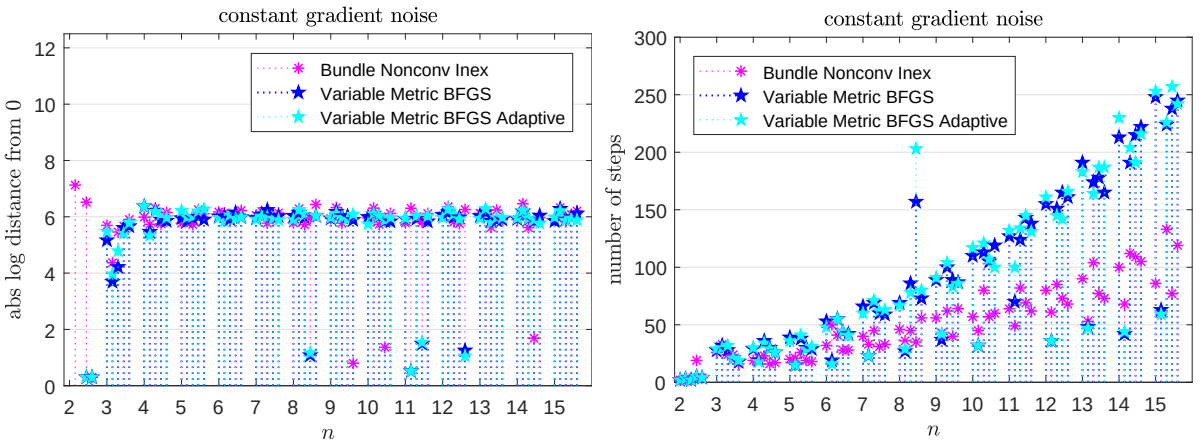

Figure 10: *Comparison of accuracy and number of steps for the proximal bundle algorithm and the variable metric bundle algorithm in the case of constant gradient noise*

Rechtschreibfehler, Namen, Stil überprüfen

References

[1] Frank H. Clarke. *Optimization and nonsmooth analysis*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics Philadelphia, 1990.

[2] Napsu Haarala, Kaisa Miettinen, and Marko M. Mäkelä. Globally convergent limited memory bundle method for large-scale nonsmooth optimization. *Mathematical Programming*, 109(1):181–205, 2007.

[3] Warren Hare and Claudia Sagastizàbal. A redistributed proximal bundle method for nonconvex optimization. *SIAM Journal on Optimization*, 20(5):2442–2473, 2010.

[4] Warren Hare, Claudia Sagastizàbal, and Mikhail Solodov. A proximal bundle method for nonsmooth nonconvex functions with inexact information. *Computational Optimization and Applications*, 63:1–28, 2016.

[5] Claude Lemaréchal and Claudia Sagastizábal. *An approach to variable metric bundle methods*, pages 144–162. Springer Berlin Heidelberg, Berlin, Heidelberg, 1994.

[6] Claude Lemaréchal and Claudia Sagastizábal. Variable metric bundle methods: From conceptual to implementable forms. *Mathematical Programming*, 76(3):393–410, 1997.

[7] Adrian S Lewis and Michael L Overton. Nonsmooth optimization via bfgs. *submitted to SIAM Journal on Optimization*, pages 1–35, 2009.

[8] L. Lukšan and J. Vlček. Globally convergent variable metric method for convex nonsmooth unconstrained minimization. *Journal of Optimization Theory and Applications*, 102(3):593–613, sep 1999.

[9] Stefan Ulbrich Michael Ulbrich. *Nichtlineare Optimierung*. Springer Basel AG, 2012.

[10] Jorge Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782, 1980.

[11] Dominikus Noll. Bundle method for non-convex minimization with inexact subgradients and function values. In *Computational and Analytical Mathematics*, pages 555–592. Springer Nature, 2013.

[12] Dominikus Noll, Olivier Prot, and Aude Rondepierre. A proximity control algorithm to minimize non-smooth and non-convex functions. *Pacific Journal of Optimization*, 4(3):571–604, 2012.

[13] Boris T. Polyak. *Introduction to Optimization*. Optimization Software , Inc., Publications Division, New York, 1987.

[14] R. Tyrrell Rockafellar and Roger J. B. Wets. *Variational Analysis*, volume 317 of *Grundlehren der mathematischen Wissenschaften*. Springer Berlin Heidelberg, 3rd edition, 2009.

[15] Jay S. Treiman. Clarke's gradients and ε-subgradients in banach spaces. *Transactions of the American Mathematical Society*, 294(1):65–65, jan 1986.

[16] J. Vlček and L. Lukšan. Globally convergent variable metric bundle method for nonconvex nondifferentiable unconstrained minimization. *Journal of Optimization Theory and Applications*, 111(2):407–430, 2001.

[17] Claudia Sagastizàbal Warren Hare. Computing proximal points of nonconvex functions. *Mathematical Programming*, 116:221–258, 2009.

[18] Claude Lemaréchal Welington de Oliveira, Claudia Sagastizàbal. Convex proximal bundle methods in depth: a unified analysis for inexact oracles. *Mathematical Programming*, 148:241–277, 2014.