Analysis of Limited-Memory BFGS on a Class of Nonsmooth Convex Functions

Azam Asl* Michael L. Overton†

September 29, 2018

Abstract

The limited memory BFGS (L-BFGS) method is widely used for large-scale unconstrained optimization, but its behavior on nonsmooth problems has received little attention. L-BFGS can be used with or without "scaling"; the use of scaling is normally recommended. A simple special case, when just one BFGS update is stored and used at every iteration, is sometimes also known as memoryless BFGS. We analyze memoryless BFGS with scaling, using any Armijo-Wolfe line search, on the function $f(x) = a|x^{(1)}| + \sum_{i=2}^{n} x^{(i)}$, initiated at any point x_0 with $x_0^{(1)} \neq 0$. We show that if $a \geq 2\sqrt{n-1}$, the absolute value of the normalized search direction generated by this method converges to a constant vector, and if, in addition, a is larger than a quantity that depends on the Armijo parameter, then the iterates converge to a non-optimal point $\bar{x}$ with $\bar{x}^{(1)} = 0$, although f is unbounded below. As we showed in previous work, the gradient method with any Armijo-Wolfe line search also fails on the same function if $a \geq \sqrt{n-1}$ and a is larger than another quantity depending on the Armijo parameter, but scaled memoryless BFGS fails under a *weaker* condition relating a to the Armijo parameter than that implying failure of the gradient method. Furthermore, in sharp contrast to the gradient method, if a specific standard Armijo-Wolfe bracketing line search is used, scaled memoryless BFGS fails when $a \geq 2\sqrt{n-1}$ *regardless* of the Armijo

*Courant Institute of Mathematical Sciences, New York University. Supported by a grant from the Simons Foundation (417314,MHW).

†Courant Institute of Mathematical Sciences, New York University. Supported in part by National Science Foundation Grant DMS-1620083.

parameter. Finally, numerical experiments indicate that similar results hold for scaled L-BFGS with any fixed number of updates.

1 Introduction

The limited memory BFGS (L-BFGS) method is widely used for large-scale unconstrained optimization, but its behavior on nonsmooth problems has received little attention. In this paper we give the first analysis of an instance of the method, sometimes known as memoryless BFGS with scaling, on a specific class of nonsmooth convex problems, showing that under given conditions the method generates iterates whose function values are bounded below, although the function itself is unbounded below.

The "full" BFGS method [NW06, Sec. 6.1], independently derived by Broyden, Fletcher, Goldfarb and Shanno in 1970, is remarkably effective for unconstrained optimization, both when the minimization objective $f : \mathbb{R}^n \to \mathbb{R}$, is smooth and when it is not, but even in the smooth case, its analysis is nontrivial. Powell [Pow76] gave the first convergence analysis for full BFGS using an Armijo-Wolfe line search, assuming that f is smooth and convex. In the smooth, nonconvex case it is generally accepted that the method is very reliable for finding stationary points (usually local minimizers), although pathological counterexamples exist [Dai02, Mas04]. Furthermore, substantial experience [LO13] shows that even when f is nonsmooth, the method is remarkably reliable for finding Clarke stationary points (again, usually local minimizers). Although BFGS requires gradient information, this is well defined for any locally Lipschitz function almost everywhere; hence, the method almost always generates iterates at which f is differentiable, and we observe that typically, it is only in the limit, at local minimizers, where the gradient is not defined. Indeed, no non-pathological counterexamples, meaning in particular ones where the starting point is not predetermined but generated randomly, are known. Some convergence results *are* known for full BFGS with an Armijo-Wolfe line search applied to specific nonsmooth problem instances: (i) $f(x) = \|x\|_2$, for which the iterates x_k converge to zero [GL18] (the special case $n = 1$ was analyzed in some detail in [LO13]), and (ii) $f(x) = |x^{(1)}| + \sum_{i=2}^{n} x^{(i)}$, for which eventually a search direction is generated on which f is unbounded below [XW17] (the special case $n = 2$ was analyzed in [LZ15]). But the question of convergence of full BFGS even on the two-variable function $f(x) = |x_1| + x_2^2$ remains open.

The full BFGS method maintains and updates an approximation to the inverse (or a factorization) of the Hessian matrix $\nabla^2 f(x)$ at every iteration, defined by current known gradient difference information $y_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1})$ along with $s_{k-1} = x_k - x_{k-1}$. The use of the Wolfe condition in the line search, requiring an increase in the directional derivative of f along the descent direction generated by BFGS, ensures that the updated inverse Hessian approximation is positive definite. (For the rationale for why this update makes sense even when f is nonsmooth, and hence $\nabla f(x)$ is not continuous, see [LO13].) Since the update has rank two, the cost of full BFGS is $O(n^2)$ operations per iteration. While this was a great advance over the cost of Newton's method in the 1970s, already in the 1980s it was realized that the cost was too high for problems where n is large, and hence the limited memory version, L-BFGS, became popular, and is widely used today (see [MR15] and [LNC+11], for example). The standard version of L-BFGS was introduced by Nocedal in 1990 and is discussed in detail in [NW06, Sec. 7.2]. Let $m \ll n$ be given. Instead of maintaining an approximation to the inverse Hessian, at the kth iteration a proxy for this matrix is implicitly defined by application of the most recent m BFGS updates (which are defined by saving y_j and s_j from the past m iterations) to a given sparse matrix H_k^0. In the absence of other information, H_k^0 may be set to the identity matrix I, but a popular choice is to instead use *scaling*, defining

$$H_k^0 = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}} I.$$

Analysis of L-BFGS is more straightforward than analysis of full BFGS in the case that f is smooth and strongly convex, and is given in [LN89, Theorem 7.1], where linear convergence to minimizers is established, regardless of whether scaling is used or not. Furthermore, it is stated in [LN89] that scaling greatly accelerates L-BFGS, and this seems to be the current accepted wisdom. However, we show in this paper that it is exactly the choice of scaling that may result in failure of L-BFGS on a specific class of nonsmooth functions. This situation is in sharp contrast to our experience with full BFGS on nonsmooth functions, where the same algorithm that is normally used for smooth functions works well also on nonsmooth functions, which can be viewed as limits of increasingly ill-conditioned smooth functions. (The superlinear convergence rate that holds generically for smooth functions is not attained in the nonsmooth case; instead, full BFGS is observed to converge linearly, in a sense described in [LO13], on nonsmooth

functions.)

We consider the convex function

$$f(x) = a|x^{(1)}| + \sum_{i=2}^{n} x^{(i)}. \tag{1}$$

Note that although f is unbounded below, it is bounded below along the line defined by the negative gradient direction from any point x with $x^{(1)} \neq 0$. In [AO18] we analyzed the gradient method with *any* Armijo-Wolfe line search applied to (1). We showed that if $a \geq \sqrt{n-1}$ and

$$a > \sqrt{(1/c_1 - 1)(n-1)}, \tag{2}$$

where c_1 is the Armijo parameter, the gradient method, initiated at *any* point x_0 with $x_0^{(1)} \neq 0$, fails in the sense that it generates a sequence converging to a non-optimal point $\bar{x}$ with $\bar{x}^{(1)} = 0$, although f is unbounded below. In the present paper, we analyze scaled L-BFGS with $m = 1$, i.e., with just one update — this method is sometimes known as *memoryless* BFGS [NW06, p. 180] — applied to the function f.

The paper is organized as follows. In §2, we define the memoryless BFGS method, using any line search satisfying the Armijo and Wolfe conditions, and derive some properties of the method applied to the function f in (1), with and without scaling, initiated at any point x_0 with $x_0^{(1)} \neq 0$. In §3, we give our theoretical results for the case when scaling is used. First, in §3.1, we show that if $a \geq 2\sqrt{n-1}$, in the limit the absolute value of the normalized search direction generated by the method converges to a constant vector, deferring the most technical parts of the proof to Appendices A and B. Then, in §3.2, we show that if a further satisfies a condition depending on the Armijo parameter, the method converges to a non-optimal point $\bar{x}$ with $\bar{x}^{(1)} = 0$. Furthermore, this condition is *weaker* than the corresponding condition (2) for the gradient method. Then, in §3.3, we show that, if a specific standard Armijo-Wolfe bracketing line search is used, scaled memoryless BFGS fails when $a \geq 2\sqrt{n-1}$ *regardless* of the Armijo parameter. This is in sharp contrast to the gradient method using the same line search, for which success or failure on the function f depends on the Armijo parameter. In §4 we present some numerical experiments which support our theoretical results, and which indicate that the results extend to scaled L-BFGS with any fixed number of updates m. We make some concluding remarks in §5.

2 The Memoryless BFGS Method

First let f denote any locally Lipschitz function mapping $\mathbb{R}^n$ to $\mathbb{R}$, and let $x_k \in \mathbb{R}^n$, $k = 0, 1, \ldots$, denote the kth iterate of an optimization algorithm where f is differentiable at x_k with gradient $\nabla f(x_k)$. Let $d_k \in \mathbb{R}^n$ denote a descent direction at the kth iteration, i.e., satisfying $\nabla f(x_k)^T d_k < 0$, and assume that f is bounded below on the line $\{x_k + td_k : t \geq 0\}$. Let c_1 and c_2, respectively the Armijo and Wolfe parameters, satisfy $0 < c_1 < c_2 < 1$. We say that the step t satisfies the Armijo condition at iteration k if

$$f(x_k + td_k) \leq f(x_k) + c_1 t \nabla f(x_k)^T d_k \tag{3}$$

and that it satisfies the Wolfe condition if

$$f \text{ is differentiable at } x_k + td_k \text{ with } \nabla f(x_k + td_k)^T d_k \geq c_2 \nabla f(x_k)^T d_k. \tag{4}$$

Note that as long as f is differentiable at the initial iterate, defining subsequent iterates by $x_{k+1} = x_k + t_k d_k$, where (4) holds for $t = t_k$, ensures that f is differentiable at all x_k.

We are now ready to define the memoryless BFGS method (L-BFGS with $m = 1$), with and without scaling. The algorithm is defined for any f, but its analysis will be specifically for (1).

Algorithm 1 (**Memoryless BFGS**), **with input $\mathbf{x_0}$**

$d_0 = -\nabla f(x_0)$ (5)

For $k = 1, 2, 3, \ldots,$ **define**

$t_{k-1} = t$ satisfying (3) and (4)

$x_k = x_{k-1} + t_{k-1} d_{k-1}$ (6)

$s_{k-1} = x_k - x_{k-1}$ (7)

$y_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1})$ (8)

if using scaling

$$\gamma_k = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}} \tag{9}$$

else

$\gamma_k = 1$ (10)

end

$$V_{k-1} = I - \frac{y_{k-1}s_{k-1}^T}{y_{k-1}^T s_{k-1}} \tag{11}$$

$$H_k = \gamma_k V_{k-1}^T V_{k-1} + \frac{s_{k-1}s_{k-1}^T}{y_{k-1}^T s_{k-1}} \tag{12}$$

$$d_k = -H_k \nabla f(x_k) \tag{13}$$

end

Note that from (6) and (7) it is immediate that for any $k \geq 0$

$$s_k = t_k d_k. \tag{14}$$

For the function f given in (1), requiring t_k to satisfy the Wolfe condition (4) is equivalent to the condition

$$\text{sgn}(x_k^{(1)}) = -\text{sgn}(x_{k-1}^{(1)}). \tag{15}$$

Via (7) we see that (15) is equivalent to the condition

$$|s_{k-1}^{(1)}| = |x_{k-1}^{(1)}| + |x_k^{(1)}|. \tag{16}$$

Without loss of generality, we assume that the initial point x_0 has a positive first component, i.e., $x_0^{(1)} > 0$, so that

$$\nabla f(x_k) = \begin{bmatrix} (-1)^k a \\ \mathbb{1} \end{bmatrix}, \tag{17}$$

where $\mathbb{1} \in \mathbb{R}^{n-1}$ is the column vector of all ones. Via (15) and (17), (8) is simply

$$y_{k-1} = \begin{bmatrix} (-1)^k 2a \\ \mathbb{0} \end{bmatrix}, \tag{18}$$

where $\mathbb{0} \in \mathbb{R}^{n-1}$ is the column vector of all zeros.

In what follows we analyze the search directions $\{d_k\}$ generated by the algorithm. Let start by defining $b_k^{(i)}$ at iteration k and for $i = 2, \ldots, n$ to be the ratio of the displacement along the ith coordinate direction relative to the 1st coordinate direction,

$$b_k^{(i)} = \frac{s_k^{(i)}}{s_k^{(1)}}. \tag{19}$$

So from (5), we have

$$b_0^{(2)} = b_0^{(3)} = \ldots = b_0^{(n)} = 1/a. \tag{20}$$

Due to the symmetry of the components $x^{(2)}, \ldots, x^{(n)}$ in the definition of the objective function (1) and also in the definition of the inverse Hessian approximation (12), as well as (20), observe that for all k

$$b_k^{(2)} = b_k^{(3)} = \ldots = b_k^{(n)}. \tag{21}$$

So, let us denote this quantity by b_k. From (18) we have

$$y_{k-1}^T s_{k-1} = (-1)^k 2a s_{k-1}^{(1)}. \tag{22}$$

Via (21) and (19), now we can rewrite V_{k-1} in (11) in terms of b_{k-1}

$$V_{k-1} = \left[\begin{array}{c|c} 0 & -b_{k-1}\mathbb{1}^T \\ \hline \mathbb{0} & I_{n-1} \end{array}\right]. \tag{23}$$

So,

$$V_{k-1}^T V_{k-1} = \left[\begin{array}{c|c} 0 & \mathbb{0}^T \\ \hline \mathbb{0} & b_{k-1}^2 \mathbb{1}\mathbb{1}^T + I_{n-1} \end{array}\right], \;\; s_{k-1} s_{k-1}^T = (s_{k-1}^{(1)})^2 \left[\begin{array}{c|c} 1 & b_{k-1}\mathbb{1}^T \\ \hline b_{k-1}\mathbb{1} & b_{k-1}^2 \mathbb{1}\mathbb{1}^T \end{array}\right].$$

Let

$$\chi_{k-1} = \frac{(s_{k-1}^{(1)})^2}{y_{k-1}^T s_{k-1}} \tag{24}$$

so (12) becomes

$$H_k = \gamma_k \left[\begin{array}{c|c} 0 & \mathbb{0}^T \\ \hline \mathbb{0} & b_{k-1}^2 \mathbb{1}\mathbb{1}^T + I_{n-1} \end{array}\right] + \chi_{k-1} \left[\begin{array}{c|c} 1 & b_{k-1}\mathbb{1}^T \\ \hline b_{k-1}\mathbb{1} & b_{k-1}^2 \mathbb{1}\mathbb{1}^T \end{array}\right].$$

We can write this more compactly as

$$H_k = \chi_{k-1} \left[\begin{array}{c|c} 1 & b_{k-1}\mathbb{1}^T \\ \hline b_{k-1}\mathbb{1} & (1 + \frac{\gamma_k}{\chi_{k-1}}) b_{k-1}^2 \mathbb{1}\mathbb{1}^T + \frac{\gamma_k}{\chi_{k-1}} I_{n-1} \end{array}\right]. \tag{25}$$

In Algorithm 1 **without scaling**, $\gamma_k = 1$ and in Algorithm 1 **with scaling**, γ_k is set to (9). In the latter case,

$$\gamma_k = \frac{(-1)^k s_{k-1}^{(1)}}{2a} = \frac{|s_{k-1}^{(1)}|}{2a}, \tag{26}$$

so, via (22) it is easy to verify that (26) and (24) are equal:

$$\gamma_k = \frac{(s_{k-1}^{(1)})^2}{y_{k-1}^T s_{k-1}} = \chi_{k-1}. \tag{27}$$

Applying (26) and (27) to (25) and using (17) we can compute the direction given by (13). So, the direction generated by Algorithm 1 **without scaling**, $\bar{d}_k$ is

$$\bar{d}_k = -\frac{|s_{k-1}^{(1)}|}{2a}\begin{bmatrix} (-1)^k a + (n-1)b_{k-1} \\ \Big((-1)^k a b_{k-1} + (1 + \frac{2a}{|s_{k-1}^{(1)}|})(n-1)b_{k-1}^2 + \frac{2a}{|s_{k-1}^{(1)}|}\Big)\mathbb{1} \end{bmatrix}, \tag{28}$$

and the direction generated by Algorithm 1 **with scaling**, d_k is

$$d_k = -\frac{|s_{k-1}^{(1)}|}{2a}\begin{bmatrix} (-1)^k a + (n-1)b_{k-1} \\ \Big((-1)^k a b_{k-1} + 2(n-1)b_{k-1}^2 + 1\Big)\mathbb{1} \end{bmatrix}. \tag{29}$$

3 Failure of Scaled Memoryless BFGS

In this section we confine ourselves to the application of memoryless BFGS (Algorithm 1) **with scaling** applied to the function defined in (1). We start by noting that b_k as defined in (19) could be equally expressed in terms of d_k,

$$b_k = \frac{d_k^{(i)}}{d_k^{(1)}} \quad \text{for } i = 2, 3, \ldots, n, \tag{30}$$

so, via (29), we can write b_k recursively as

$$b_k = \frac{(-1)^k a b_{k-1} + 2(n-1)b_{k-1}^2 + 1}{(-1)^k a + (n-1)b_{k-1}}. \tag{31}$$

3.1 Convergence of the Absolute Value of the Normalized Search Direction when $2\sqrt{n-1} \leq a$

We begin with a lemma.

Lemma 1. *Suppose* $\sqrt{3(n-1)} \leq a$, $b_0 = 1/a$ *and* b_k *is defined by* (31). *Then* $|b_k| \leq 1/a$ *and furthermore* $\{b_k\}$ *alternates in sign with*

$$|b_k| = \frac{1 + (n-1)b_{k-1}^2}{a - (n-1)|b_{k-1}|} - |b_{k-1}|. \tag{32}$$

Proof. See Appendix A for the proof. □

Now define

$$b = \frac{a - \sqrt{a^2 - 3(n-1)}}{3(n-1)} \tag{33}$$

and note that

$$\frac{1}{2a} \leq b \leq \frac{1}{a}.$$

Next we show the sequence $\{|b_k|\}$ converges to b under a slightly stronger assumption.

Theorem 2. *For* $2\sqrt{n-1} \leq a$ *the sequence defined by* (32) *converges and moreover*

$$\lim_{k\to\infty} |b_k| = b.$$

Proof. See Appendix B for the proof. □

The technique that we use to prove Theorem 2 requires the assumption $2\sqrt{n-1} \leq a$; however, in our experiments we observe that convergence occurs when $\sqrt{3(n-1)} \leq a$. In particular, with $\sqrt{3(n-1)} = a$ we get $|b_k| = 1/a$ for $k = 0, 1, \ldots$, which means the normalized direction is exactly the same as the normalized direction generated by the gradient method.

Note that the convergence result established in this theorem does not require any assumption of symmetry with respect to variables $2, 3, \ldots, n$ in the initial point x_0. The only assumption on x_0 is that $x_0^{(1)} > 0$. We need $x_0^{(1)} \neq 0$ so that f is differentiable at x_0; the assumption on the sign is purely for convenience.

Assumption 1. For the rest of this article we assume that

$$2\sqrt{n-1} \leq a.$$

With this assumption, as a direct implication of Theorem 2, for any given positive ϵ there exists K such that for $k \geq K$ we have

$$||b_k| - b| < \frac{\epsilon}{n-1}. \tag{34}$$

As we showed in Lemma 1, for $k \geq 0$ we have $|b_k| \leq 1/a$ and therefore

$$\frac{3(n-1)}{a} \leq a - \frac{n-1}{a} \leq a - (n-1)|b_k|. \tag{35}$$

Thus, $a - (n-1)|b_k|$ is positive and bounded away from zero.

Putting (31) and (32) together we can rewrite (29) as

$$d_k = -\frac{|s_{k-1}^{(1)}|}{2a}(a - (n-1)|b_{k-1}|)\begin{bmatrix} (-1)^k \\ |b_k|\mathbb{1} \end{bmatrix}. \tag{36}$$

Since $|b_k|$ converges by Theorem 2, we see that in the limit the normalized direction $d_k/\|d_k\|_2$ alternates between two limiting directions. For an illustration, see Figures 1 and 2. It is this property that allows us to establish, under some subsequent assumptions, that scaled memoryless BFGS generates iterates x_k for which $f(x_k)$ is bounded below even though f is unbounded below.

3.2 Dependence on the Armijo Condition

Combining (17) and (36) we get

$$\nabla f(x_k)^T d_k = -|d_k^{(1)}|\begin{bmatrix} (-1)^k a \\ \mathbb{1} \end{bmatrix}^T \begin{bmatrix} (-1)^k \\ |b_k|\mathbb{1} \end{bmatrix} = -|d_k^{(1)}|\big(a + (n-1)|b_k|\big), \tag{37}$$

so the Armijo condition (3) with $t = t_k$ at iteration k is

$$c_1 t_k |d_k^{(1)}|\big(a + (n-1)|b_k|\big) \leq f(x_k) - f(x_k + t_k d_k). \tag{38}$$

Define

$$\varphi_k = \frac{c_1\big(a + (n-1)|b_k|\big) + a - (n-1)|b_k|}{2a}. \tag{39}$$

Since $0 < c_1 < 1$, we have

$$0 < \frac{a - (n-1)|b_k|}{2a} < \varphi_k < 1 \tag{40}$$

for all k. The left-most inequality is obtained via (35).

Lemma 3.

$$\frac{(n-1)|b_k|}{a} < \varphi_k. \tag{41}$$

Proof. Using Lemma 1 we know $3(n-1)|b_k| \le a$ for all k, and so

$$2(n-1)|b_k| \le a - (n-1)|b_k|,$$

and since

$$\frac{a-(n-1)|b_k|}{2a} = \varphi_k - c_1\frac{a+(n-1)|b_k|}{2a},$$

and $c_1 > 0$, (41) follows. □

If t_k satisfies the Wolfe condition, i.e. t_k is large enough that the sign change (15) occurs, this implies that

$$|x_k^{(1)}| < t_k|d_k^{(1)}|. \tag{42}$$

Given this we can derive $f(x_k) - f(x_k + t_k d_k)$ using the definition of b_k in (30) as follows:

$$f(x_k) - f(x_k + t_k d_k) = 2a|x_k^{(1)}| - \big(a-(n-1)|b_k|\big)t_k|d_k^{(1)}|. \tag{43}$$

This allows us to express the Armijo condition more concisely assuming that the Wolfe condition holds.

Lemma 4. *Suppose t_k satisfies the Wolfe condition* (15)*. Then for t_k to satisfy the Armijo condition* (38) *we must have*

$$\varphi_k t_k|d_k^{(1)}| \le |x_k^{(1)}|. \tag{44}$$

Proof. Combining (43) and (38) we get

$$c_1 t_k|d_k^{(1)}|\big(a+(n-1)|b_k|\big) \le 2a|x_k^{(1)}| - \big(a-(n-1)|b_k|\big)t_k|d_k^{(1)}|,$$

and using the definition of φ_k in (39), (44) follows. □

Corollary 5. *For $k \ge 1$ we have*

$$|s_k^{(1)}| \le |s_{k-1}^{(1)}|\frac{1-\varphi_{k-1}}{\varphi_k}. \tag{45}$$

Proof. Summing the Armijo inequality (44) for two consecutive iterations we obtain

$$|s_{k-1}^{(1)}|\varphi_{k-1} + |s_k^{(1)}|\varphi_k \le |x_{k-1}^{(1)}| + |x_k^{(1)}|,$$

and noticing that the R.H.S., according to (16), is equal to $|s_{k-1}^{(1)}|$ we get (45). □

Lemma 6. *For any given $\epsilon > 0$ let K be the smallest integer such that for any $k \ge K$, (34) holds. Then for all $N > K$ we have*

$$f(x_K) - f(x_N) < a|x_K| + \big((n-1)b + \epsilon\big) \sum_{k=K}^{N-1} |s_k^{(1)}|. \tag{46}$$

Proof. Using $t_k d_k = s_k$ and $x_{k+1} = x_k + s_k$ in (43) and then applying (34) we obtain

$$f(x_k) - f(x_{k+1}) < 2a|x_k^{(1)}| - a|s_k^{(1)}| + \big((n-1)b + \epsilon\big)|s_k^{(1)}|. \tag{47}$$

Summing up (47) from $k = K$ to $k = N-1$ and recalling (16), we get

$$f(x_K) - f(x_N) <$$
$$a \sum_{k=K}^{N-1} |s_k^{(1)}| + a|x_K| - a|x_N| - a \sum_{k=K}^{N-1} |s_k^{(1)}| + \big((n-1)b + \epsilon\big) \sum_{k=K}^{N-1} |s_k^{(1)}|.$$

Canceling the first and fourth terms and dropping $-a|x_N|$, we arrive at (46). □

From applying Theorem 2 to the definition of φ_k in (39) it is immediate that $\{\varphi_k\}$ converges. Let

$$\varphi = \frac{c_1\big(a + (n-1)b\big) + a - (n-1)b}{2a}, \tag{48}$$

so

$$\lim_{k\to\infty} \varphi_k = \varphi. \tag{49}$$

Lemma 7. *If*

$$0 < \epsilon \le \frac{\sqrt{a^2 - 3(n-1)}}{3}, \tag{50}$$

then after a sufficient number of iterations we have

$$\left| \frac{1-\varphi_{k-1}}{\varphi_k} - \frac{1-\varphi}{\varphi} \right| < \frac{15}{a}\epsilon. \tag{51}$$

Proof. By rearranging terms in (33) and using (50)we get

$$(n-1)b+\epsilon \leq (n-1)b+\frac{\sqrt{a^2-3(n-1)}}{3}=\frac{a}{3}. \tag{52}$$

Define K as in Lemma 6. Using (34) and (52), for $k \geq K$ we have

$$0 < a-(n-1)b-\epsilon < a-(n-1)|b_k|.$$

Combining this with (40) we get

$$0 < \frac{a-(n-1)b-\epsilon}{2a} < \varphi_k < 1.$$

Hence,

$$1 < \frac{1}{\varphi_k} < \frac{2a}{a-(n-1)b-\epsilon} \leq \frac{2a}{a-\frac{a}{3}} = 3.$$

Since $0 < c_1 < 1$, from (34), (39), (48) and (49) we get

$$|\varphi_k - \varphi| < \frac{(1+c_1)\epsilon}{2a} < \frac{\epsilon}{a}.$$

So,

$$\begin{aligned}\left|\frac{1-\varphi_{k-1}}{\varphi_k}-\frac{1-\varphi}{\varphi}\right| &= \left|\frac{1}{\varphi_k}-1+\frac{\varphi_k-\varphi_{k-1}}{\varphi_k}-\frac{1}{\varphi}+1\right| \\ &< \left|\frac{\varphi-\varphi_k}{\varphi_k\varphi}\right|+\left|\frac{\varphi_k-\varphi_{k-1}}{\varphi_k}\right| < \frac{\epsilon}{a\varphi_k}\left(\frac{1}{\varphi}+2\right).\end{aligned}$$

Note that $1 < 1/\varphi_k < 3$ applies to all φ_k (as well as the limit φ) with $k \geq K$, and therefore we conclude (51). □

Let

$$\psi_\epsilon = \frac{1-\varphi}{\varphi}+\frac{15}{a}\epsilon. \tag{53}$$

If Lemma 7 applies then from (45) and (51) we conclude

$$|s_k^{(1)}| < \psi_\epsilon |s_{k-1}^{(1)}|. \tag{54}$$

That is to say, with ϵ satisfying (50), after at most K iterations, (54) holds. Consequently, with the additional assumption $\psi_\epsilon < 1$, we obtain

$$\sum_{k=K}^{N-1} |s_k^{(1)}| < |s_K^{(1)}| \frac{1}{1-\psi_\epsilon}. \tag{55}$$

Now we can prove the main result of this subsection. Recall that $c_1 < 1$.

Theorem 8. *Suppose c_1 is chosen large enough that*

$$\frac{1}{c_1} - 1 < \frac{a}{(n-1)b} \tag{56}$$

holds. Then, using any Armijo-Wolfe line search with any starting point x_0 with $x_0^{(1)} \neq 0$, memoryless BFGS with scaling applied to (1) *fails in the sense that $f(x_N)$ is bounded below as $N \to \infty$.*

Proof. It follows from (56) and (48) that $\varphi > 1/2$. Therefore, using (53), we can choose ϵ small enough such that $\psi_\epsilon < 1$ holds in addition to (50). Applying Lemmas 6 and 7, we conclude that there exists K such that for for any $N > K$, (55) holds, and, substituting this into (46) we get

$$f(x_K) - f(x_N) < a|x_K| + |s_K^{(1)}| \frac{(n-1)b + \epsilon}{1 - \psi_\epsilon}. \tag{57}$$

This establishes that $f(x_N)$ is bounded below for all $N > K$. □

Using (33) we see the failure condition (56) for scaled memoryless BFGS with any Armijo-Wolfe line search applied to (1), is equivalent to

$$\frac{1 - c_1}{c_1}(n-1) \leq a^2 + a\sqrt{a^2 - 3(n-1)}. \tag{58}$$

The corresponding failure condition for the gradient method on the same function, again using any Armijo-Wolfe line search, is, as we showed in [AO18],

$$\frac{1 - c_1}{c_1}(n-1) \leq a^2. \tag{59}$$

Hence, scaled memoryless BFGS fails under a *weaker* condition relating a to the Armijo parameter than the condition for failure of the gradient method on the same function with the same line search conditions. Indeed, Assumption 1 implies

$$a^2 + a\sqrt{a^2 - 3(n-1)} \geq 4(n-1) + 2\sqrt{n-1}\sqrt{n-1} = 6(n-1).$$

So, if the Armijo parameter $c_1 \geq 1/7$, then (58) holds. In contrast, the same assumption implies that if $c_1 \geq 1/5$, then (59) holds. So, scaled memoryless BFGS with any Armijo-Wolfe line search applied to (1) fails under a weaker condition on the Armijo parameter than the gradient method does.

3.3 Results for a specific Armijo-Wolfe line search, independent of the Armijo parameter

Considering only the first component of the direction d_k in (36) we have

$$\frac{2a}{a-(n-1)|b_{k-1}|}|d_k^{(1)}| = |s_{k-1}^{(1)}|. \tag{60}$$

Using (14), it follows that if

$$t_k < \frac{2a}{a-(n-1)|b_{k-1}|}, \tag{61}$$

we have $|s_k^{(1)}| < |s_{k-1}^{(1)}|$. Note that the R.H.S. of (61) is greater than two. However, as shown in the next lemma, except at the initial iteration ($k = 0$), $t = 2$ is always large enough to satisfy the Wolfe condition, implying that there exists $t \leq 2$ satisfying both the Armijo and Wolfe conditions.

Lemma 9. *For $k \geq 1$, the steplength $t_k = 2$ always satisfies the Wolfe condition* (15)*, i.e., we have*

$$|x_k^{(1)}| < 2|d_k^{(1)}|. \tag{62}$$

Proof. Since $k \geq 1$, we know that the Armijo and Wolfe conditions hold at iteration $k-1$ by definition of Algorithm 1. So, using (44) and (14) we have

$$\varphi_{k-1}|s_{k-1}^{(1)}| \leq |x_{k-1}^{(1)}|. \tag{63}$$

Using the inequality (41) in the L.H.S. and the equality (16) in the R.H.S. we get

$$\frac{(n-1)|b_{k-1}|}{a}|s_{k-1}^{(1)}| < |s_{k-1}^{(1)}| - |x_k^{(1)}|,$$

i.e.

$$|x_k^{(1)}| < |s_{k-1}^{(1)}|\frac{a-(n-1)|b_{k-1}|}{a}.$$

Substituting (60) into the R.H.S., we obtain (62). □

Suppose that we use the Armijo-Wolfe bracketing line search given in [LO13, AO18]. This line search begins with the unit step. If this step, $t = 1$, does not satisfy the Armijo condition (3), then the step is contracted, so the final step is less than one. On the other hand, if $t = 1$ satisfies (3),

then the line search checks whether the Wolfe condition (4) is satisfied too. If it is, then the line search quits; if not, the step is doubled and hence the line search next checks whether $t = 2$ satisfies (4). At the initial iteration ($k = 0$), several doublings might be needed before (4) is eventually satisfied. But for subsequent steps ($k \geq 1$), we know that $t = 2$ must satisfy the Wolfe condition, so the final step must satisfy $t_k = 2$ (if $t = 2$ satisfies (3)) or $t_k < 2$ (otherwise). Thus, for $k \geq 1$ we always have $t_k \leq 2$.

Now we can present the main result of this subsection: using a line search with the property just described, the optimization method fails.

Theorem 10. *When memoryless BFGS with scaling is applied to* (1)*, using an Armijo-Wolfe line search (such as the bracketing line search of [LO13]) which returns steplength $t_k \leq 2$ for $k \geq 1$, the method fails in the sense that $f(x_N)$ is bounded below as $N \to \infty$.*

Proof. Recalling $t_{k+1} d_{k+1}^{(1)} = s_{k+1}^{(1)}$ again, using (60) and $t_{k+1} \leq 2$ we find that

$$|s_{k+1}^{(1)}| \leq \frac{a - (n-1)|b_k|}{a} |s_k^{(1)}|. \tag{64}$$

Let $\epsilon > 0$ satisfy

$$\delta_\epsilon \equiv \frac{a - (n-1)b}{a} + \frac{\epsilon}{a} < 1.$$

Define K as in Lemma 6, so that (34) holds, and hence

$$\frac{a - (n-1)|b_k|}{a} < \delta_\epsilon.$$

Applying this inequality to (64) we get

$$|s_{k+1}^{(1)}| \leq \delta_\epsilon |s_k^{(1)}|, \tag{65}$$

and since $\delta_\epsilon < 1$ we have

$$\sum_{k=K}^{N-1} |s_k^{(1)}| < |s_K^{(1)}| \frac{1}{1 - \delta_\epsilon}. \tag{66}$$

By substituting this into (46) we get

$$f(x_K) - f(x_N) < a|x_K| + |s_K^{(1)}| \frac{(n-1)b + \epsilon}{1 - \delta_\epsilon},$$

which shows $f(x_N)$ is bounded below. □

Finally, we have the following corollary to Theorems 8 and 10. Recall that γ_k is the scaling parameter (see (26)).

Corollary 11. *If the assumptions required by either Theorem 8 or 10 hold, then*

$$\lim_{N\to\infty} \gamma_N = 0 \tag{67}$$

and x_N converges to a non-optimal point $\bar{x}$ such that

$$\bar{x} = [0, \bar{x}^{(2)}, \ldots, \bar{x}^{(n)}]^T. \tag{68}$$

Proof. It is immediate from (55) or (66) that $|s_N^{(1)}| \to 0$ as $N \to \infty$, so from (26), we conclude (67). Also due to (16) we have $|x_N^{(1)}| \to 0$, and since $f(x_N) = a|x^{(1)}| + \sum_{i=2}^{n-1} x_N^{(i)}$ is bounded below, so is $\sum_{i=2}^{n-1} x_N^{(i)}$. Due to (35) and (36), we have $d_{N-1}^{(i)} < 0$, for $i = 2, 3, \ldots, n$, so $t_{N-1} d_{N-1}^{(i)} = x_N^{(i)} - x_{N-1}^{(i)} < 0$, and therefore $x_N^{(i)}$ is strictly decreasing as $N \to \infty$. Hence, $x_N^{(i)}$ converges to a limit $\bar{x}^{(i)}$. □

Due to the symmetry we discussed earlier, the total decrease along each component, $x_0^{(i)} - \bar{x}^{(i)} = \sum_{k=0}^{N} s_k^{(i)}$, is the same for $i = 2, 3, \ldots, n$.

4 Experiments

Our experiments use the Armijo-Wolfe bracketing line search given in [LO13, AO18], so, according to the results of §3.3, scaled memoryless BFGS (L-BFGS with $m = 1$) should fail when a satisfies Assumption 1: $2\sqrt{n-1} \leq a$. This is illustrated in Figure 1, which shows an experiment where we ran both the gradient method and memoryless BFGS with scaling on function (1) with $a = 3$ and $n = 2$, starting from the same random initial point. We see that memoryless BFGS with scaling fails, in the sense that it converges to a non-optimal point, while the gradient method succeeds, in the sense that it generates iterates with $f(x_k) \downarrow -\infty$. The figure also shows the iterates generated by full BFGS, which finds a direction along which f is unbounded below in just five iterations.

However, although the proof of Theorem 2 does require Assumption 1 we observe that $\sqrt{3(n-1)} \leq a$ suffices for $\{|b_k|\}$ and consequently $|d_k|/\|d_k\|_2$

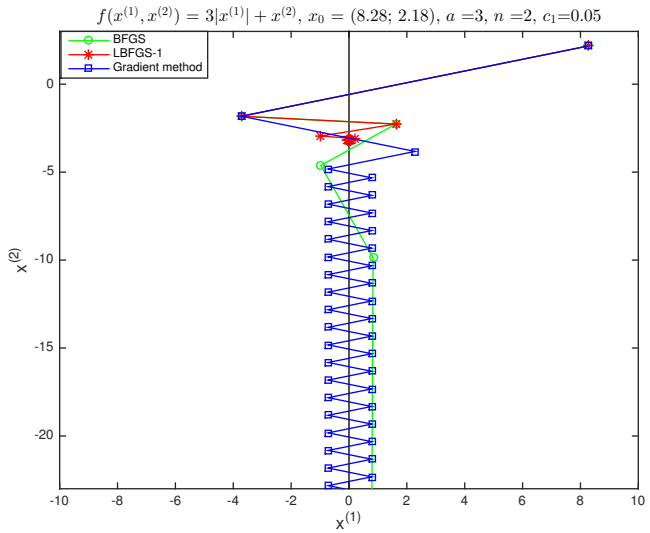

Figure 1: Full BFGS (green circles), scaled memoryless BFGS (red asterisks) and the gradient method (blue squares) applied to the function (1) defined by $a = 3$ and $n = 2$. Scaled memoryless BFGS fails while full BFGS and the gradient method succeed.

to converge. In Figure 2 we repeat the same experiment with $a = \sqrt{3}$ and $n = 2$, showing that scaled memoryless BFGS still fails. In this case, as noted in Section 3, the normalized direction is the same as the normalized direction generated by the gradient method, but unlike in the gradient method, the magnitude of the directions d_k converge to zero so scaled memoryless BFGS fails.

However, if we set a to $\sqrt{3} - 0.001$ the method succeeds. This is demonstrated in Figure 3: observe that although one at first has the impression that x_k is converging to a non-optimal point, a search direction is generated on which f is unbounded below "at the last minute".

Extensive additional experiments verify that the condition $\sqrt{3(n-1)} \leq a$, as opposed to Assumption 1, is sufficient for failure. The magenta asterisks in Figure 4 confirm this observation. Starting from 5000 random points generated from the normal distribution, we called memoryless BFGS with scaling to minimize function (1) with $n = 30$ and for values of a ranging from 9.317 to 9.337, since for $n = 30$, $\sqrt{3(n-1)} \approx 9.327$. We see that for $9.327 \leq a$ the failure rate is 1 (100%), while for $9.32 > a$ the failure rate is 0. In comparison to a similar experiment in [AO18] for the gradient method,

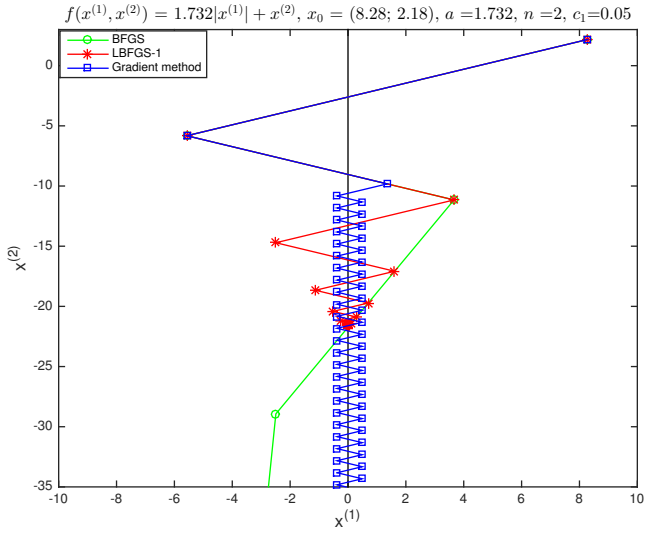

Figure 2: Full BFGS (green circles), scaled memoryless BFGS (red asterisks) and the gradient method (blue squares) applied to the function (1) defined by $a = \sqrt{3}$ and $n = 2$. Scaled memoryless BFGS fails while BFGS and the gradient method succeed.

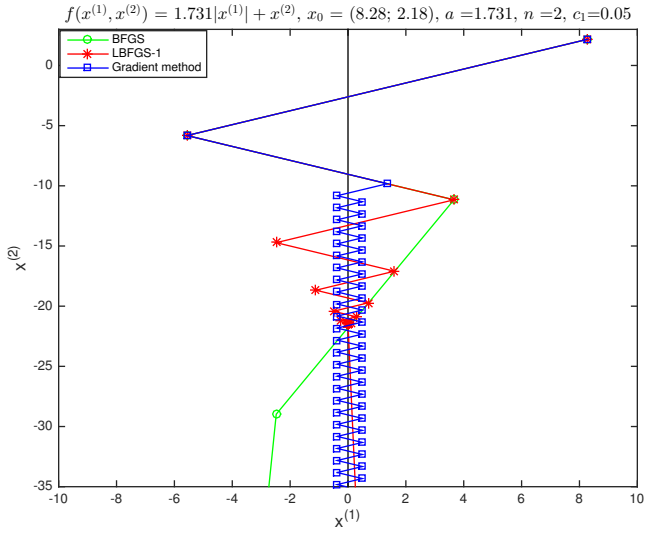

Figure 3: Full BFGS (green circles), scaled memoryless BFGS (red asterisks) and the gradient method (blue squares) applied to the function (1) defined by $a = \sqrt{3} - 0.001$ and $n = 2$. All methods succeed.

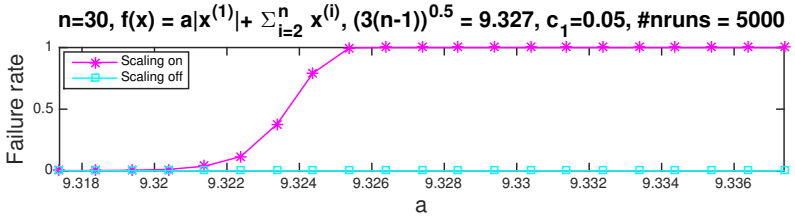

Figure 4: The failure rate of memoryless BFGS with scaling (magenta asterisks) and without scaling (cyan squares) applied to function (1) with $n = 30$ and 21 different values of a, initiating the method from 5000 random points. With scaling, the failure rate is 1 for $9.327 \leq a$. Without scaling, the failure rate is 0 regardless of a.

the transition from failure rate 0 to 1 is quite sharp here. This might be explained by the fact that the gradient method fails because the stepsize $t_k \to 0$, whereas for scaled memoryless BFGS, t_k does not converge to zero; it is the scale γ_k and consequently the norm of d_k which converges to zero. Hence, rounding error prevents the observation of a sharp transition in the results for the gradient method, as explained in [AO18]; by comparison, rounding error plays a less significant role in the experiments reported here.

The cyan squares in Figure 4 show the results from the same experiment for memoryless BFGS *without* scaling, using the same 5000 initial points. In this case, the method is successful regardless of the value of a.

Finally, our experiments with scaled L-BFGS with any fixed number of updates m suggest that the theoretical results we presented for scaled L-BFGS with only one update might extend, although undoubtedly in a far more complicated form, to any number of updates. In the top plot in Figure 5 we show results from many experiments, with each initiated from 1000 random points. The horizontal axis shows m, the number of updates, while the vertical axis shows the observed failure rate. We set $n = 4$, so that $\sqrt{3(n-1)} = 3$, and show results for values of a ranging from 2.99 to 300.

The top plot shows results for $c_1 = 0.01$ and the bottom plot for $c_1 = 0.001$. We see in both plots that as a gets larger for a fixed m, the failure rate increases. On the other hand, as m gets larger for a fixed a, the failure rate decreases. Since our experiments use the Armijo-Wolfe bracketing line search, the results do not depend significantly on the Armijo parameter; in particular, we know from the results of §3.3 that there is no dependence on the Armijo parameter when $m = 1$. However, we do observe small differences for the larger values of m, where the failure rate is slightly higher for the

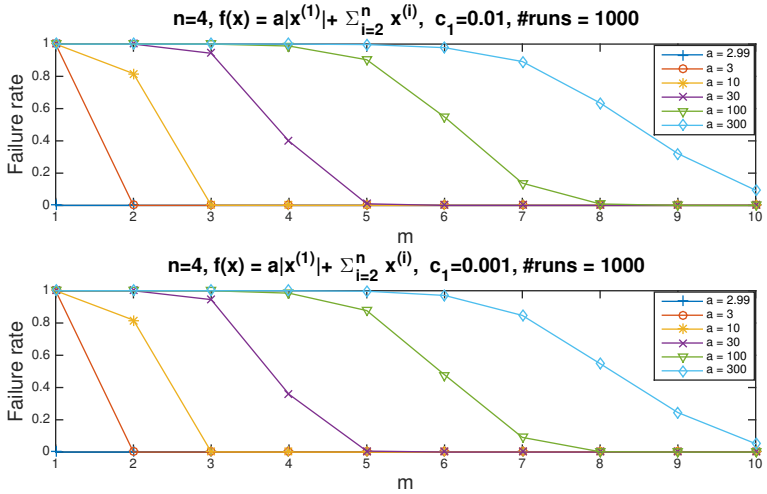

Figure 5: Top: The failure rate for each scaled L-BFGS-m, where the number of updates m ranges from 1 to 10, applied to function (1) with $a = 2.99$ (blue pluses), $a = 3$ (orange circles), $a = 10$ (yellow asterisks), $a = 30$ (purple crosses), $a = 100$ (green triangles) and finally $a = 300$ (cyan diamonds), with $c_1 = 0.01$ and $n = 4$ and hence $\sqrt{3(n-1)} = 3$, and with each experiment initiated from 1000 random points. Bottom: The same experiment as in the top plot except that $c_1 = 0.001$.

larger Armijo parameter. This is consistent with the theoretical results in §3.2 as well as those in [AO18], where, if a is relatively large, then to avoid failure c_1 should not be too large.

5 Concluding Remarks

We have given the first analysis of a variant of L-BFGS applied to a non-smooth function, showing that the scaled version of memoryless BFGS (L-BFGS with just one update) applied to (1) generates iterates converging to a non-optimal point under simple conditions. One of these conditions applies to the method with any Armijo-Wolfe line search and depends on the Armijo parameter. The other condition applies to the method using a standard Armijo-Wolfe bracketing line search and does not depend on the Armijo parameter. Experiments suggest that extended results likely hold for L-BFGS with more than one update, though clearly a generalized analysis would be much more complicated.

We do not know whether L-BFGS without scaling applied to the same function can converge to a non-optimal point, but numerical experiments suggest that this cannot happen. More generally, it remains an open question as to whether scaling is generally inadvisable when applying L-BFGS to nonsmooth functions, despite its apparent advantage for smooth optimization.

Acknowledgment. Many thanks to Margaret H. Wright for arranging financial support for the first author from the Simons Foundation.

A Proof of Lemma 1

Suppose $\sqrt{3(n-1)} \leq a$. Using a change of variable such that $\beta_k = b_k$ when k is even, and $\beta_k = -b_k$ when k is odd, (31) becomes

$$\beta_k = \frac{1 + (n-1)\beta_{k-1}^2}{a - (n-1)\beta_{k-1}} - \beta_{k-1}. \tag{69}$$

From (20) we have $\beta_0 = 1/a$. Using induction we prove that $0 < \beta_k \leq 1/a$. This is clearly true for $k = 0$. Suppose we have $0 < \beta_{k-1} \leq 1/a$. Hence

$$\beta_{k-1} < \frac{1}{a - (n-1)\beta_{k-1}} < \frac{1 + (n-1)\beta_{k-1}^2}{a - (n-1)\beta_{k-1}},$$

so, dropping the middle term and moving β_{k-1} to the R.H.S., we get exactly the definition of β_k according to (69). So, we have $0 < \beta_k$. Next, starting from $\sqrt{3(n-1)} \leq a$, we show that $\beta_k \leq 1/a$:

$$\begin{aligned}
&\frac{3(n-1)}{a} \leq a \Rightarrow \\
&\frac{(n-1)}{a} + 2(n-1)\beta_{k-1} \leq a \Rightarrow \\
&\frac{a^2 + n - 1}{a} \leq 2(a - (n-1)\beta_{k-1}) \Rightarrow \\
&\frac{a^2 + n - 1}{a(a - (n-1)\beta_{k-1})} \leq 2.
\end{aligned}$$

Multiplying both sides by β_{k-1} we get

$$\frac{a\beta_{k-1} + 1}{a - (n-1)\beta_{k-1}} - \frac{1}{a} \leq 2\beta_{k-1},$$

and finally by moving $1/a$ to the right and $2\beta_{k-1}$ to the left we get

$$\frac{1+(n-1)\beta_{k-1}^2}{a-(n-1)\beta_{k-1}} - \beta_{k-1} \leq \frac{1}{a}.$$

The L.H.S. is β_k as it's defined in (69), so $\beta_k \leq 1/a$. Recalling the change of variable in the beginning of the proof it follows that $\beta_k = |b_k|$. So, from (69) we get (32).

B Proof of Theorem 2

We continue to use the same change of variable as before, that is $\beta_k = b_k$ when k is even, and $\beta_k = -b_k$ when k is odd. In this way, (69) is equivalent to (32), and we prove that if $2\sqrt{n-1} \leq a$, then $\{\beta_k\}$ converges. From a little rearrangement in (69) we can easily get

$$a(\beta_k + \beta_{k-1}) = 1 + 2(n-1)\beta_{k-1}^2 + (n-1)\beta_{k-1}\beta_k, \tag{70}$$

and by moving $(n-1)\beta_{k-1}\beta_k$ to the left and adding 1 to both sides we get

$$a(\beta_k + \beta_{k-1}) - (n-1)\beta_{k-1}\beta_k + 1 = 2\Big(1 + (n-1)\beta_{k-1}^2\Big). \tag{71}$$

For further simplification we define

$$\rho_k = \frac{1+(n-1)\beta_k^2}{a-(n-1)\beta_k}, \tag{72}$$

so we can rewrite (69) as

$$\beta_{k+1} = \rho_k - \beta_k. \tag{73}$$

By applying (73) recursively we obtain

$$\beta_{k+1} - \beta_{k-1} = \rho_k - \rho_{k-1}. \tag{74}$$

Note that from (72) we have

$$\rho_k - \rho_{k-1} = \frac{1+(n-1)\beta_k^2}{a-(n-1)\beta_k} - \frac{1+(n-1)\beta_{k-1}^2}{a-(n-1)\beta_{k-1}}$$
$$= \frac{\Big(1+(n-1)\beta_k^2\Big)\Big(a-(n-1)\beta_{k-1}\Big) - \Big(1+(n-1)\beta_{k-1}^2\Big)\Big(a-(n-1)\beta_k\Big)}{\Big(a-(n-1)\beta_k\Big)\Big(a-(n-1)\beta_{k-1}\Big)}$$
$$= \frac{(\beta_k-\beta_{k-1})(n-1)\Big(a(\beta_k+\beta_{k-1}) - (n-1)\beta_{k-1}\beta_k + 1\Big)}{\Big(a-(n-1)\beta_k\Big)\Big(a-(n-1)\beta_{k-1}\Big)}. \tag{75}$$

The last factor in the numerator is the L.H.S. in (71), so

$$\rho_k - \rho_{k-1} = \frac{(\beta_k-\beta_{k-1})(n-1)2\Big(1+(n-1)\beta_{k-1}^2\Big)}{\Big(a-(n-1)\beta_k\Big)\Big(a-(n-1)\beta_{k-1}\Big)}. \tag{76}$$

Hence, since all of the factors in this product except $(\beta_k - \beta_{k-1})$ are known to be positive, we have

$$(\rho_k - \rho_{k-1})(\beta_k - \beta_{k-1}) \geq 0. \tag{77}$$

Putting (74) and (77) together we conclude

$$(\beta_{k+1} - \beta_{k-1})(\beta_k - \beta_{k-1}) \geq 0. \tag{78}$$

As the next step we will show that

$$(\beta_{k+1} - \beta_k)(\beta_k - \beta_{k-1}) \leq 0. \tag{79}$$

Since $a \geq 2\sqrt{n-1}$ and using $1/a \geq \beta_{k-1}$ we get

$$\Big(a^2 - 4(n-1)\Big)\Big(a^2 + (n-1)\Big) \geq 0 \Rightarrow$$
$$a^2 - 3(n-1) \geq \frac{4(n-1)^2}{a^2} \Rightarrow$$
$$a^2 - 3(n-1) \geq 4(n-1)^2\beta_{k-1}^2 \Rightarrow$$
$$a^2 - 3(n-1) - 4(n-1)^2\beta_{k-1}^2 \geq 0.$$

By adding and deducting $2(n-1)^2\beta_k\beta_{k-1}$ to the L.H.S. above we get

$$a^2 - 2(n-1)\Big(1 + 2(n-1)\beta_{k-1}^2 + (n-1)\beta_{k-1}\beta_k\Big) + 2(n-1)^2\beta_k\beta_{k-1} - (n-1) \geq 0.$$

By combining this with (70) we get

$$a^2 - 2(n-1)a(\beta_k + \beta_{k-1}) + 2(n-1)^2\beta_k\beta_{k-1} - (n-1) \geq 0.$$

By moving some of the terms to the R.H.S. and factorizing the L.H.S. we get

$$\Big(a-(n-1)\beta_k\Big)\Big(a-(n-1)\beta_{k-1}\Big) \geq a(n-1)(\beta_k+\beta_{k-1})-(n-1)^2\beta_k\beta_{k-1}+(n-1),$$

which we can write as

$$1 \geq \frac{(n-1)\Big(a(\beta_k + \beta_{k-1}) - (n-1)\beta_k\beta_{k-1} + 1\Big)}{\Big(a-(n-1)\beta_k\Big)\Big(a-(n-1)\beta_{k-1}\Big)}. \tag{80}$$

Now, suppose $\beta_k - \beta_{k-1} \geq 0$. Multiplying both sides of the inequality (80) by $\beta_k - \beta_{k-1}$, according to (75) we get

$$\beta_k - \beta_{k-1} \geq \rho_k - \rho_{k-1},$$

so,

$$\rho_{k-1} - \beta_{k-1} \geq \rho_k - \beta_k$$

which means that via (73) we have shown $\beta_k \geq \beta_{k+1}$. Alternatively, if we had $\beta_k - \beta_{k-1} \leq 0$ above, then we would get $\beta_k \leq \beta_{k+1}$. Hence, we always have $(\beta_{k+1} - \beta_k)(\beta_k - \beta_{k-1}) \leq 0$, which is exactly inequality (79).

Since we start with $\beta_0 = 1/a$, according to Lemma 1 we have $\beta_1 \leq \beta_0$. Using (79) inductively we get

$$\beta_1 - \beta_0 \leq 0, \quad 0 \leq \beta_2 - \beta_1, \quad \beta_3 - \beta_2 \leq 0, \ldots$$

and from applying (78) to each one of these inequalities we conclude

$$\beta_2 - \beta_0 \leq 0, \quad 0 \leq \beta_3 - \beta_1, \quad \beta_4 - \beta_2 \leq 0, \ldots$$

which shows that we can split $\{\beta_k\}$ into two separate monotonically decreasing and increasing subsequences:

$$0 < \ldots \beta_4 \leq \beta_2 \leq \beta_0 = 1/a,$$
$$0 < \beta_1 \leq \beta_3 \leq \beta_5 \ldots < 1/a.$$

By the bounded monotone convergence theorem we conclude that each one of these subsequences converge, i.e.

$$\lim_{k\to\infty} |\beta_{k+2} - \beta_k| = 0,$$

and recalling (74) we get

$$\lim_{k\to\infty} |\rho_{k+1} - \rho_k| = 0.$$

On the other hand, looking at the equality in (75) we know that except $(\beta_{k+1} - \beta_k)$ all the factors in the numerator and denominator are bounded away from zero. So therefore we must have

$$\lim_{k\to\infty} |\beta_{k+1} - \beta_k| = 0,$$

and hence, since the even and odd sequences both converge, they must have the same limit. Using the definition of β_{k+1} in (69) we get

$$\lim_{k\to\infty} \left| \frac{1 + (n-1)\beta_k^2}{a - (n-1)\beta_k} - 2\beta_k \right| = 0.$$

Since the denominator is bounded away from zero we must have

$$\lim_{k\to\infty} 3(n-1)\beta_k^2 - 2a\beta_k + 1 = 0.$$

The two roots of the limiting quadratic equation are

$$\frac{a \pm \sqrt{a^2 - 3(n-1)}}{3(n-1)}.$$

The smaller root is b as defined in (33) and the larger root is greater than $1/a$, which according to Lemma 1 is not possible. Hence,

$$\lim_{k\to\infty} \beta_k = \lim_{k\to\infty} |b_k| = b.$$

References

[AO18] Azam Asl and Michael L. Overton. Analysis of the Gradient Method with an Armijo-Wolfe Line Search on a Class of Non-smooth Convex Functions. September 2018. arXiv:1711.08517v2.

[Dai02] Yu-Hong Dai. Convergence properties of the BFGS algorithm. *SIAM J. Optim.*, 13(3):693–701 (2003), 2002.

[GL18] J. Guo and A. Lewis. Nonsmooth variants of Powell's BFGS convergence theorem. *SIAM Journal on Optimization*, 28(2):1301–1311, 2018.

[LN89] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Math. Programming*, 45(3, (Ser. B)):503–528, 1989.

[LNC+11] Quoc V. Le, Jiquan Ngiam, Adam Coates, Abhik Lahiri, Bobby Prochnow, and Andrew Y. Ng. On optimization methods for deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 265–272, USA, 2011. Omnipress.

[LO13] Adrian S. Lewis and Michael L. Overton. Nonsmooth optimization via quasi-Newton methods. *Math. Program.*, 141(1-2, Ser. A):135–163, 2013.

[LZ15] A. S. Lewis and S. Zhang. Nonsmoothness and a variable metric method. *J. Optim. Theory Appl.*, 165(1):151–171, 2015.

[Mas04] Walter F. Mascarenhas. The BFGS method with exact line searches fails for non-convex objective functions. *Math. Program.*, 99(1, Ser. A):49–61, 2004.

[MR15] Aryan Mokhtari and Alejandro Ribeiro. Global convergence of online limited memory bfgs. 16:3151–3181, 2015.

[NW06] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006.

[Pow76] M. J. D. Powell. Some global convergence properties of a variable metric algorithm for minimization without exact line searches. In *Nonlinear Programming*, pages 53–72, Providence, 1976. Amer. Math. Soc. SIAM-AMS Proc., Vol. IX.

[XW17] Yuchen Xie and Andreas Waechter. On the convergence of BFGS on a class of piecewise linear non-smooth functions. December 2017. arXiv:1712.08571.