A Nonmonotone Line Search Method for Noisy Minimization

Nataša Krejić* Zorana Lužanin* Filip Nikolovski†
Irena Stojkovska†

December 18, 2013

Abstract

A nonmonotone line search method for optimization in noisy environment is proposed. The method is defined for arbitrary search directions and uses only the noisy function values. Convergence of the proposed method is established under a set of standard assumptions. The computational issues are considered and the presented numerical results affirm that nonmonotone strategies are worth considering. Four different line search rules with three different directions are compared numerically. The influence of nonmonotonicity is discussed.

Key words. line search method, nonmonotone line search rule, unconstrained optimization, noisy function

AMS subject classification. 90C56, 65K05

1 Introduction

The problem under consideration is

$$\min_{x\in\mathbb{R}^n} f(x) \tag{1}$$

assuming that $f : \mathbb{R}^n \to \mathbb{R}$ has continuous partial derivatives and only noisy measurements $F(x)$ are available,

$$F(x) = f(x) + \delta(x), \tag{2}$$

at every $x \in \mathbb{R}^n$, where $\delta(x)$ represents the noise at x.

Such problems are very common when physical system measurements or computer simulations are used for approximations. One approach to cope with noise is to collect several function evaluations $F(x)$ at each value x generated in the optimization process, then take the average of these values as an estimate for $f(x)$, Andradottir [1]. The averaging procedure increases the number of function

*Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia, e-mail: {natasak@uns.ac.rs, zorana@dmi.uns.ac.rs}

†Department of Mathematics, Faculty of Natural Sciences and Mathematics, Ss. Cyril and Methodius University, Arhimedova 3, 1000 Skopje, Macedonia, e-mail: {filipnikolovski@gmail.com, irenatra@pmf.ukim.mk}

evaluations but saves the total number of iterations and it is well justified for a number of problems. However, it is not always possible to get an arbitrary number of function evaluation at the same point.

Random search is another approach in solving the problem (1). Within this approach a random search direction d_k is generated at each iteration and the new point is defined as $x_k + d_k$ if the following inequality is satisfied

$$F(x_k + d_k) < F(x_k) - \tau_k. \tag{3}$$

Here $\tau_k > 0$ is a threshold value and it is convenient to set it at the noise level. The drawback of this approach is that an inappropriate threshold value may results in rejecting many iterates, see [22].

Among direct search methods that have been considered for optimization in presence of noise is the coordinate search method, Lucidi and Sciandrone [17]. Namely, starting from an iterate x_k, this method searches along a coordinate direction d_k for a stepsize α_k that satisfies

$$F(x_k + \alpha_k d_k) < F(x_k) - \gamma \alpha_k^2, \tag{4}$$

where $\gamma > 0$. Only noisy functional values are used, which makes the method derivative-free. As smaller steps are allowed the method is more immune to the noise influence than the threshold approach. Some of the recent methods for optimization of noisy functions are considered in [3, 14, 23, 25].

Nonmonotone line search strategies are a well developed class of methods for classical optimization. The dominant three nonmonotone rules are originally presented in Grippo et al. [11], Li and Fukushima [16] and Zhang and Hager [26]. All of these three strategies are successfully used for solving different problems in either derivative based or derivative-free methods, [24, 8, 19, 5, 15, 4]. There are several important properties of nonmonotone line search methods. First of all, one can consider search directions which are not necessarily descent in all iterations as happens in many cases, starting with Symmetric Rank 1, Spectral gradient, gradient approximations etc. Further more these methods are applicable even if the gradient is not available. The importance of this property is even more emphasized in noisy environment where the original function (and its gradient) might be unknown. An additional property of nonmonotone line search rules that makes them attractive is the ability of converging to a global solution of problems with multiple local and global solutions. This property is reported in several papers, see for example [26].

There are several papers dealing with unconstrained optimization problems within line search framework and nonmonotone methods in particular, with results that are applicable on noisy problems even if the noise is not explicitly assumed, [5, 6, 8, 17, 19]. Different line search rules are considered and the set of search directions varies from gradient based directions, then second-order directions like QN directions, approximate gradient directions to random directions. The main idea in this paper is to present a unified theory for such methods if the noisy functional values are the only available values and the line search has to be defined with these values. Therefore the method proposed in this paper is a derivative-free nonmonotone line search method for an arbitrary direction d_k. Nonmonotonicity help in coping with the noise. The convergence analysis relays heavily on results from [5, 8, 17] but extends them in the sense

that the presented statements cover the case of noisy functional values. In practical implementation we consider search directions that require approximations of gradient and/or Hessian in presence of noise, and we give a gradient approximation procedure in presence of noise. Thus in Section 2 we present the model algorithm and analyze its convergence in Section 3. Numerical results are presented in Section 4.

2 Model algorithm

For solving (1) using the noisy measurements (2), we will consider two nonmonotone line search strategies of the following form

$$F(x_k + \alpha_k d_k) \leq \overline{F}_k + \eta_k - \alpha_k^2 \beta_k. \tag{5}$$

The term $\overline{F}_k$ in the first line search strategy is defined by

$$\overline{F}_k = \max\{F(x_k), \ldots, F(x_{\max\{k-M+1,0\}})\}, \tag{6}$$

for an arbitrary but fixed $M \in \mathbb{N}$. The above formula (6) is analogous to the one in [11], where the true functional values $f(x)$ are used.

In the second line search strategy $\overline{F}_k$ is defined by

$$Q_{k+1} = r_k Q_k + 1, \ \overline{F}_{k+1} = \frac{r_k Q_k (\overline{F}_k + \eta_k) + F(x_{k+1})}{Q_{k+1}}, \tag{7}$$

with $r_k \in [r_{\min}, r_{\max}]$, $0 \leq r_{\min} \leq r_{\max} \leq 1$, $\overline{F}_0 = F(x_0)$ and $Q_0 = 1$. The definition of the sequence $\{\overline{F}_k\}$, given by (7), is similar to the one proposed in [26] and modified in [5], but here we use the noisy functional values.

The sequence $\{\eta_k\}$, [16] is such that

$$\eta_k > 0, \ \sum_{k=0}^{\infty} \eta_k = \eta < \infty, \tag{8}$$

while $\{\beta_k\}$, [8] is bounded and

$$\beta_k > 0 \text{ and } \lim_{k \in K} \beta_k = 0 \Rightarrow \lim_{k \in K} g_k = 0, \tag{9}$$

for all infinite subsets of indices $K \subset \mathbb{N}$, where $g_k = g(x_k)$ is the gradient of f at x_k.

It is easy to see that

$$\overline{F}_k \geq F(x_k), \tag{10}$$

if $\overline{F}_k$ is defined either by (6) or by (7). Furthermore $\eta_k > 0$ and thus the line search (5) indeed generates a nonmonotone sequence of noisy functional values $\{F(x_k)\}$. Roughly speaking, when the iterates are far from the solution, the sequence of true functional values $\{f(x_k)\}$ will also be nonmonotone, since far from the solution the noise has small influence. On the other hand when the iterates are approaching the solution the sequence $\{F(x_k)\}$ becomes more monotone but due to the larger influence of the noise in the vicinity of the solution, the sequence $\{f(x_k)\}$ is more likely to stay nonmonotone. The sequence $\{\beta_k\}$ has a forcing nature, and the trivial choice $\beta_k \equiv 1$ is admissible.

Both nonmonotone line search methods can be stated as follows.

Model algorithm. Given $\{\eta_k\}$ such that (8) holds and $\{\beta_k\}$ such that (9) holds, $x_0 \in \mathbb{R}^n$ and $D > 0$.

Step 1. Compute d_k such that $\|d_k\| \leq D$.

Step 2. Compute $\overline{F}_k$.

Step 3. Choose $\alpha_k > 0$ such that (5) is satisfied.

Step 4. Set $x_{k+1} = x_k + \alpha_k d_k$ and $k = k + 1$.

Specifying $\overline{F}_k$ in Step 2 of Model algorithm as (6) or (7), we cover both of the line search strategies presented above. The positive sequence $\{\eta_k\}$ ensures that the the line search rule (5) is well defined as $\alpha_k > 0$ exists for an arbitrary direction d_k.

3 Convergence analysis

For establishing the convergence of proposed methods the following assumptions on the objective function and noise are needed.

A1 The objective function $f \in C^1(\mathbb{R}^n)$ is bounded from below i.e. there exists m such that $f(x) \geq m$ for all $x \in \mathbb{R}^n$

A2 The realized noise is bounded from above i.e there exists a constant $\Delta > 0$ such that

$$|\delta(x_k)| \leq \Delta. \tag{11}$$

The boundedness of noise stated in A2 might look as a strong assumption at the first glance. For example the white noise is not bounded in general. However we are interested only in the realized noise and thus A2 is not a big obstacle in practical implementation of the algorithm. The same set of assumptions is used in [17] as well.

Let us first consider Model algorithm with $\overline{F}_k$ defined by (6). The analysis presented below follows closely the results presented in [6, 8] with the important difference being the fact that the noisy functional values are used. So we define an additional sequence, [8],

$$V_l = \max\{F(x_{(l-1)M+1}), \ldots, F(x_{lM})\},\ l = 1, 2, \ldots$$

and $\nu(l) \in \{(l-1)M, \ldots, lM\}$ such that $F(x_{\nu(l)}) = V_l$. The following result for the line search rule (5), where $\overline{F}_k$ is defined by (6), can be obtained.

Theorem 3.1. *Assume that $\{x_k\}$ is generated by Model algorithm with $\overline{F}_k$ defined by (6) and that A1-A2 hold. Then,*

$$\lim_{l \to \infty} \alpha^2_{\nu(l)-1} \beta_{\nu(l)-1} = 0.$$

Proof. The line search rule implies for $l = 1, 2, ...$

$$\begin{aligned} F(x_{lM+1}) &\leq \max_{1 \leq j \leq M} F(x_{(l-1)M+j}) + \eta_{lM} - \alpha^2_{lM} \beta_{lM} \\ &= V_l + \eta_{lM} - \alpha^2_{lM} \beta_{lM} \\ &\leq V_l + \eta_{lM} \end{aligned}$$

With similar reasoning we obtain

$$\begin{aligned} F(x_{lM+2}) &\leq \max_{1\leq j\leq M} F(x_{(l-1)M+1+j}) + \eta_{lM+1} - \alpha_{lM+1}^2\beta_{lM+1} \\ &\leq \max\{V_l, F(x_{lM+1})\} + \eta_{lM+1} - \alpha_{lM+1}^2\beta_{lM+1} \\ &\leq V_l + \eta_{lM} + \eta_{lM+1} - \alpha_{lM+1}^2\beta_{lM+1} \\ &\leq V_l + \eta_{lM} + \eta_{lM+1} \end{aligned}$$

By induction we have

$$F(x_{lM+s}) \leq V_l + \sum_{j=0}^{s-1} \eta_{lM+j} - \alpha_{lM+s-1}^2\beta_{lM+s-1},$$

for all $s = 1, 2, ..., M$.

Since $\nu(l+1) \in \{lM+1, ..., lM+M\}$,

$$V_{l+1} = F(x_{\nu(l+1)}) \leq V_l + \sum_{j=0}^{M-1} \eta_{lM+j} - \alpha_{\nu(l+1)-1}^2\ \beta_{\nu(l+1)-1}.$$

So, for all $l = 1, 2, ...$ we have

$$F(x_{\nu(l+1)}) \leq F(x_{\nu(l)}) + \sum_{j=0}^{M-1} \eta_{lM+j} - \alpha_{\nu(l+1)-1}^2\ \beta_{\nu(l+1)-1}. \tag{12}$$

Now, adding the above inequalities for $l = 1, 2, ..., L$ we obtain

$$F(x_{\nu(L+1)}) \leq F(x_{\nu(1)}) + \sum_{j=M}^{(L+1)M-1} \eta_j - \sum_{j=1}^{L} \alpha_{\nu(j+1)-1}^2\beta_{\nu(j+1)-1}.$$

As $F(x) = f(x) + \delta(x)$ the assumption A2 implies

$$f(x_{\nu(L+1)}) \leq f(x_{\nu(1)}) + \sum_{j=M}^{(L+1)M-1} \eta_j - \sum_{j=1}^{L} \alpha_{\nu(j+1)-1}^2\beta_{\nu(j+1)-1} + 2\Delta.$$

Now, A1 and (8) imply that for all $L = 1, 2, ...,$

$$\begin{aligned} \sum_{j=1}^{L} \alpha_{\nu(j+1)-1}^2\beta_{\nu(j+1)-1} &\leq f(x_{\nu(1)}) + \sum_{j=M}^{(L+1)M-1} \eta_j + 2\Delta - f(x_{\nu(L+1)}) \leq \\ &\leq f(x_{\nu(1)}) + \eta + 2\Delta - m \end{aligned}$$

Therefore $\sum_{j=1}^{\infty} \alpha_{\nu(j+1)-1}^2\beta_{\nu(j+1)-1} < \infty$, and $\lim_{j\to\infty} \alpha_{\nu(j)-1}^2\beta_{\nu(j)-1} = 0$, which completes the proof. ■

The analogous result for the line search rule (5), where $\overline{F}_k$ is defined by (7), can be proved as demonstrated below. The proof is closely relying on [5].

Theorem 3.2. *Assume that $\{x_k\}$ is generated by Model algorithm with $\overline{F}_k$ be defined by (7) and that A1-A2 are satisfied. Then there exists an infinite subset $K \subset \mathbb{N}$ such that,*

$$\lim_{k\in K} \alpha_k^2\beta_k = 0.$$

Moreover, if $r_{max} < 1$ *then*

$$\lim_{k\to\infty} \alpha_k^2 \beta_k = 0.$$

Proof. Let $\{x_j\}$ be a sequence generated by Model algorithm and $\overline{F}_k$ be defined by (7). Then, for any j

$$F(x_{j+1}) \leq \overline{F}_j + \eta_j - \alpha_j^2 \beta_j$$

and

$$\begin{aligned} \overline{F}_{j+1} &= \frac{r_j Q_j (\overline{F}_j + \eta_j) + F(x_{j+1})}{Q_{j+1}} \leq \\ &\leq \frac{r_j Q_j (\overline{F}_j + \eta_j) + \overline{F}_j + \eta_j - \alpha_j^2 \beta_j}{Q_{j+1}} = \\ &= \frac{Q_{j+1} (\overline{F}_j + \eta_j) - \alpha_j^2 \beta_j}{Q_{j+1}} = \\ &= \overline{F}_j + \eta_j - \frac{\alpha_j^2 \beta_j}{Q_{j+1}}. \end{aligned}$$

Summing up the above inequalities for $j = 0, 1, ..., k$ we have

$$\overline{F}_{k+1} \leq \overline{F}_0 + \sum_{j=0}^{k} \eta_j - \sum_{j=0}^{k} \frac{\alpha_j^2 \beta_j}{Q_{j+1}}.$$

As $\overline{F}_0 = F(x_0)$, $\sum_j \eta_j < \eta$ and $\overline{F}_{k+1} \geq F(x_{k+1})$ by (10),

$$\sum_{j=0}^{k} \frac{\alpha_j^2 \beta_j}{Q_{j+1}} \leq \overline{F}_0 + \sum_{j=0}^{k} \eta_j - \overline{F}_{k+1} \leq F(x_0) + \eta - F(x_{k+1}).$$

Having in mind that $F(x) = f(x) + \delta(x)$, A1 and A2 imply

$$\sum_{j=0}^{k} \frac{\alpha_j^2 \beta_j}{Q_{j+1}} \leq f(x_0) + \eta - f(x_{k+1}) + 2\Delta \leq f(x_0) + \eta - m + 2\Delta.$$

So,

$$\sum_{j=0}^{\infty} \frac{\alpha_j^2 \beta_j}{Q_{j+1}} < \infty. \tag{13}$$

Given that

$$Q_{j+1} = r_j Q_j \leq Q_j + 1 \leq \ldots \leq j + 2$$

we can conclude that $\liminf_{j\to\infty} \alpha_j^2 \beta_j = 0$ i.e. there exists an infinite subset $K \subset \mathbb{N}$ such that $\lim_{j\in K} \alpha_j^2 \beta_j = 0$.

If $r_{\max} < 1$ then

$$Q_{j+1} = 1 + \sum_{p=0}^{j} \prod_{q=0}^{p} r_{j-q} \leq 1 + \sum_{p=0}^{j} r_{\max}^{p+1} \leq \sum_{p=0}^{\infty} r_{\max}^{p} = \frac{1}{1 - r_{\max}},$$

and from (13) we conclude that $\lim_{j\to\infty} \alpha_j^2 \beta_j = 0$, which completes the proof.■

The last two theorems provide an infinite subsequence $\{x_k\}_{k\in K}$ of $\{x_k\}$ such that $\lim_{k\in K}\alpha_k^2\beta_k = 0$, if $\{x_k\}$ is generated by either of the two considered line searches in Model algorithm. Furthermore, under Assumption A2, for a sequence $\{x_k\}$, we define the sequence $\{\delta_k\}$ by

$$\delta_k = \sup_{x\in B_k} |\delta(x)|, \tag{14}$$

where $B_k = \{x \in \mathbb{R}^n | \|x_k - x\| \leq D\}$, and $D > 0$ is the upper bound of the search directions d_k from Model algorithm.

The following theorem gives the convergence result for any sequence $\{x_k\}$ generated by Model algorithm. The idea is similar to the one presented in [17].

Theorem 3.3. *Assume that A1-A2 hold and that $\{x_k\}$ is an iterative sequence generated by Model algorithm. Assume that (x^*, d) is a limit point of the subsequence $\{(x_k, d_k)\}_{k\in K}$, where K is an infinite subset of $\mathbb{N}$ such that $\lim_{k\in K}\alpha_k^2\beta_k = 0$. Assume also that for the sequence $\{\delta_k\}$ defined by (14) the following condition is satisfied,*

$$\lim_{k\in K}\frac{\delta_k}{\alpha_k} = 0. \tag{15}$$

Then,

$$\langle g(x^*), d\rangle \geq 0.$$

Proof. Let K_1 be an infinite subset of K such that $\lim_{k\in K_1} x_k = x^*$ and $\lim_{k\in K_1} d_k = d$. The theorem conditions imply

$$\lim_{k\in K_1}\alpha_k^2\beta_k = 0.$$

If some subsequence of $\{\beta_k\}$ converges to zero, then $g(x^*) = 0$ and the proof is done.

Otherwise, we have that $\lim_{k\in K_1}\alpha_k = 0$. So, for $k \in K_1$ large enough we have that $\alpha_k < 1$. Without loss of generality let us assume that $\alpha_k < 1$ for all $k \in K_1$. Therefore, for $k \in K_1$, the stepsize α_k that fulfills the line-search rule is necessarily preceded by stepsize α'_k such that

$$\lim_{k\in K_1}\alpha'_k = 0 \tag{16}$$

and

$$F(x_k + \alpha'_k d_k) > \overline{F}_k + \eta_k - (\alpha'_k)^2\beta_k.$$

From (8) and (10), we have

$$F(x_k + \alpha'_k d_k) > F(x_k) - (\alpha'_k)^2\beta_k.$$

As $F(x) = f(x) + \delta(x)$, taking into account (14) we get

$$f(x_k + \alpha'_k d_k) > f(x_k) - (\alpha'_k)^2\beta_k - 2\delta_k$$

and

$$\frac{f(x_k + \alpha'_k d_k) - f(x_k)}{\alpha'_k} > -\alpha'_k\beta_k - 2\frac{\delta_k}{\alpha'_k},$$

for all $k \in K_1$. Since, $\alpha'_k > \alpha_k$ we have that for all $k \in K_1$,

$$\frac{f(x_k + \alpha'_k d_k) - f(x_k)}{\alpha'_k} > -\alpha'_k \beta_k - 2\frac{\delta_k}{\alpha_k}.$$

By the Mean Value Theorem, for all $k \in K_1$ there exists $\xi_k \in [0, 1]$ such that

$$\langle g(x_k + \xi_k \alpha'_k d_k), d_k \rangle > -\alpha'_k \beta_k - 2\frac{\delta_k}{\alpha_k}.$$

Therefore, for all $k \in K_1$,

$$\langle g(x_k + \xi_k \alpha'_k d_k) - g(x_k), d_k \rangle + \langle g(x_k), d_k \rangle > -\alpha'_k \beta_k - 2\frac{\delta_k}{\alpha_k}$$

and

$$\langle g(x_k), d_k \rangle > -\alpha'_k \beta_k - 2\frac{\delta_k}{\alpha_k} - \|g(x_k + \xi_k \alpha'_k d_k) - g(x_k)\| \|d_k\|.$$

Since β_k and $\|d_k\|$ are bounded, from (15) and (16) we have that

$$\lim_{k \in K_1} \left(\alpha'_k \beta_k + 2\frac{\delta_k}{\alpha_k} + \|g(x_k + \xi_k \alpha'_k d_k) - g(x_k)\| \|d_k\| \right) = 0.$$

As $\lim_{k \in K_1} \langle g(x_k), d_k \rangle = \langle g(x^*), d \rangle$, we conclude that $\langle g(x^*), d \rangle \geq 0$. ∎

We can prove that under additional assumptions on search direction, stationary points can be achieved up to any arbitrary precision by Model algorithm, as in [8], even when only noisy measurements (2) are used for optimization.

Theorem 3.4. *Assume that all conditions from Theorem 3.3 hold,* $0 < \rho < 1$, $0 < d < D < +\infty$. *Suppose that the level set* $\Omega = \{x \in \mathbb{R}^n | f(x) \leq f(x_0) + \eta + 2\Delta\}$ *is bounded and that* K_1 *is an infinite subset of* K *such that for all* $k \in K_1$ *the search directions* d_k *satisfy*

$$d \leq \|d_k\| \leq D \quad \textit{and} \quad \langle d_k, g(x_k) \rangle \leq \rho \|g(x_k)\| \|d_k\|. \tag{17}$$

Than, for all $\varepsilon > 0$, *there exists* $k \in \mathbb{N}$ *such that* $\|g(x_k\| \leq \varepsilon$.

Proof. From Model algorithm, equation (2) and Assumption A2, we have that $f(x_k) \leq f(x_0) + \eta + 2\Delta$ for all $k \in \mathbb{N}$, so the sequence $\{x_k\}$ is bounded. Furthermore, the proof proceeds as the proof of Corollary 1 in [8]. ∎

Note that even thought the nonmonotone line search method does not necessary produce descent directions, it is natural to suppose that the condition (17) in Theorem 3.4 can hold for an infinitely many search directions, since one of the main purpose of line search methods is eventually making a reduction in objective functional values.

4 Numerical results

We tested three different nonmonotone line-search methods and compare them to the classical Armijo rule but with noisy function values on a set of 18 problems from Moré, Garbow and Hillstrom, [18]. All problems have the loss function of

Problem	n	x_0
Helical valley function	3	$(-1, 0, 0)$
Biggs EXP6 function	6	$(10, 20, 10, 10, 10, 10)$
Gaussian function	3	$(4, 10, 0)$
Powell badly scaled function	2	$(0, 5)$
Box three-dimensionaly function	3	$(0, 10, 20)$
Variably dimensioned function	10	$(9/10, 8/10, ..., 0)$
Watson function	6	$(0, 0, ..., 0)$
Penalty function I	4	$(1, 2, 3, 4)$
Penalty function II	4	$(5/2, 5/2, 5/2, 5/2)$
Brown badly scaled function	2	$(1, 1)$
Brown and Dennis function	4	$(25, 5, -5, 1)$
Gulf research and development function	3	$(5, 2.5, 0.15)$
Trigonometric function	10	$(1, 1, ..., 1)$
Extended Rosenbrock function	10	$(-1.2, 1, ..., -1.2, 1)$
Extended Powell singular function	12	$(3, -1, 0, 1, ..., 3, -1, 0, 1)$
Beale function	2	$(1, 1)$
Wood function	4	$(-3, -1, -3, -1)$
Chebyquad function	10	$(5/11, 10/11, ..., 50/11)$

Table 1: Test problems

the form $f(x) = \sum_{i=1}^{m} f_i^2(x)$. The test functions as well as the dimensions n and the initial points x_0 are given in Table 1.

The noisy measurements of the objective function are obtained with the simulated normally distributed noise $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ that is multiplied with the exact functional value to obtain $F(x) = f(x)(1+\varepsilon)$ at each point x_k generated by the algorithm. Six different noise levels $\sigma = 0.01, 0.1, 0.5, 1, 5, 10$ are considered. We report the results for $\sigma = 0.1, 1, 10$ in this paper while all results are available at `http://people.dmi.uns.ac.rs/~natasa/additional.pdf`. The gradient approximation with centered differences is implemented using the noisy function values as follows. Several other possibilities for the gradient estimation could be considered as well, see [10]. For a positive sequence $\{h_k\}$, the gradient of f at x_k is approximated with $\hat{g}_k$, given by

$$[\hat{g}_k]_j = \frac{F(x_k + h_k e_j) - F(x_k - h_k e_j)}{2h_k}, \; j = 1, 2, ..., n, \tag{18}$$

where e_j is jth coordinate vector. The choice of h_k is crucial for the approximation of the gradient in presence of noise. If several of these values are too small than the approximation of the gradient might be rather poor resulting in very small steps in the line search and preventing the progress of the algorithm, [12]. Assuming that the Hessian $\nabla^2 f$ is Lipschitz continuous with the constant L, it can be shown that the estimation error is the following

$$\|\hat{g}_k - \nabla f_k\| \leq \left(\frac{L}{2} h_k^2 + \frac{\Delta}{h_k} \right) \sqrt{n}, \tag{19}$$

where Δ is an upper bound for the noise, introduced in the assumption A2. Since the right-hand side in (19) achieves minimum for $h_k = \sqrt[3]{\Delta/L}$, it is clear that very small h_k is not a good choice in noisy environments. We have established empirically that the choice $h_k = 3\sigma$, where σ is the noise level, is appropriate for the test collection we considered.

Three different choices of the search direction d_k in Step 1 of Model algorithm are tested:

(SGR) The spectral gradient search direction at each iteration is defined as

$$d_k = -\hat{g}_k/\sigma_k,$$

where $0 < \sigma_{\min} < \sigma_k < \sigma_{\max} < \infty$ is the spectral coefficient obtained recursively by

$$\sigma_{k+1} = \max\left\{\sigma_{\min}, \min\left\{\sigma_{\max}, \frac{(\hat{g}_{k+1} - \hat{g}_k)^T(x_{k+1} - x_k)}{\|x_{k+1} - x_k\|^2}\right\}\right\} \tag{20}$$

for $k = 0, 1, \ldots$ with $\sigma_0 = 1$, $\sigma_{\min} = 10^{-10}$, $\sigma_{\max} = 10^{10}$. See [2, 21] for spectral gradient noise-free methods.

(BFGS) The BFGS search direction that we use is

$$d_k = -H_k\hat{g}_k,$$

where the inverse Hessian approximation is updated by

$$H_{k+1} = \left(I - \rho s_k y_k^T\right) H_k \left(I - \rho y_k s_k^T\right) + \rho s_k s_k^T \tag{21}$$

for all $k = 0, 1, \ldots,$ with $H_0 = I$, $s_k = x_{k+1} - x_k$, $y_k = \hat{g}_{k+1} - \hat{g}_k$ and $\rho = 1/(y_k^T s_k)$. The initial approximation H_0 is rescaled before the update in the first iteration to:

$$H_0 \leftarrow \frac{y_k^T s_k}{y_k^T y_k} I.$$

The rescaling makes H_0 similar to $\left(\nabla^2 F(x_0)\right)^{-1}$, see [20]. If the positive curvature condition $y_k^T s_k > 0$ is not satisfied, we set $H_{k+1} = H_k$.

(SR1) The SR1 search direction is given by

$$d_k = -H_k\hat{g}_k,$$

where H_k is updated by

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^T}{(s_k - H_k y_k)^T y_k} \tag{22}$$

for all $k = 0, 1, \ldots,$ with the same definitions of H_0, s_k and y_k as in the BFGS implementation above. If the stability condition $|(s_k - H_k y_k)^T y_k| \geq \rho\|y_k\|\|s_k - H_k y_k\|$ is not satisfied for $\rho = 10^{-8}$ we set $H_{k+1} = H_k$.

The term $\overline{F}_k$ in Model algorithm is calculated by (6) for $M = 1$ and $M = 10$ and by (7) with $r_k = 0.85$ resulting in three different nonmonotone line-search rules LS2, LS3 and LS4. They are compared with the Armijo rule stated for the noisy functional values and obtained for $M = 1$ and $\eta_k = 0$, denoted here as LS1.

(LS1) The monotone line-search defined by

$$F(x_k + \alpha_k d_k) \leq F(x_k) - \alpha_k^2\beta_k.$$

(LS2) The nonmonotone line-search defined by

$$F(x_k + \alpha_k d_k) \leq F(x_k) + \eta_k - \alpha_k^2 \beta_k.$$

(LS3) The line-search rule

$$F(x_k + \alpha_k d_k) \leq \overline{F}_k + \eta_k - \alpha_k^2 \beta_k,$$

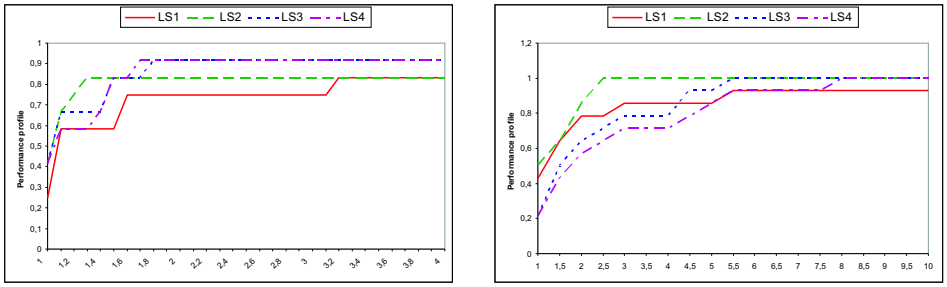
where $\overline{F}_k$ is computed by (6) with $M = 10$.

(LS4) The line-search rule

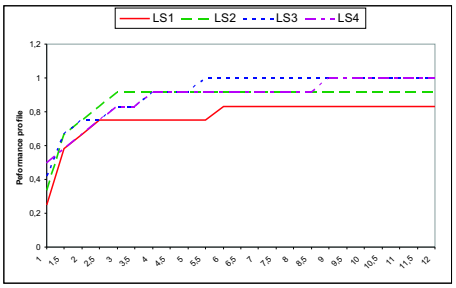
$$F(x_k + \alpha_k d_k) \leq \overline{F}_k + \eta_k - \alpha_k^2 \beta_k,$$

where $\overline{F}_k$ is computed by (7) with $r_k = 0.85$.

The specified choices of the search directions and line-search rules give 12 different methods that are tested.



a. SGR methods b. BFGS methods



c. SR1 methods

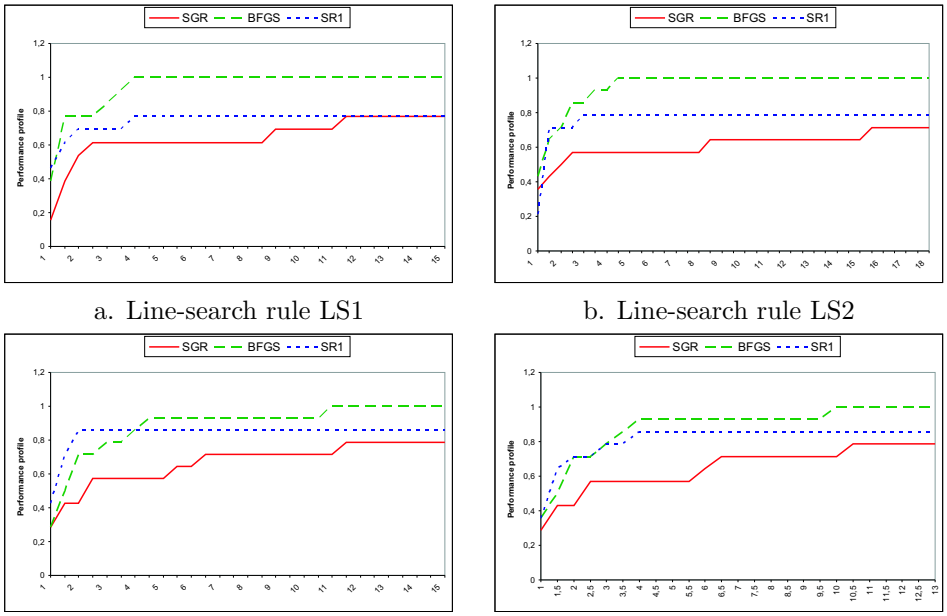
Figure 1: The performance profiles for different directions and $\sigma = 0.1$

The initial step length tested in Step 3 of Model algorithm is $\alpha = 1$. If the line search rule (5) is not satisfied then a smaller step is computed using the safeguarded quadratic interpolation, [7]. For the sequences η_k and β_k defined by (8) and (9), we set

$$\eta_k = |F(x_0)|/k^{1.1} \quad \text{and} \quad \beta_k \equiv 1.$$

The algorithm stops when the criterium

$$|F(x_k)| < (1 + 2\sigma) \cdot |F(x_0)| \cdot 10^{-3} \tag{23}$$



a. Line-search rule LS1 b. Line-search rule LS2

c. Line-search rule LS3 d. Line-search rule LS4

Figure 2: The performance profiles for different line search rules and $\sigma = 0.1$

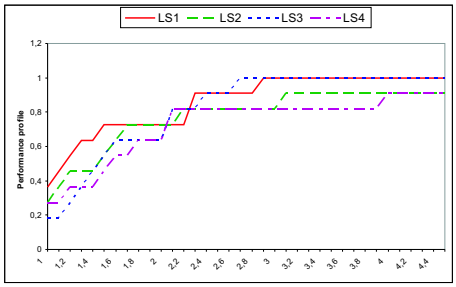
is satisfied, where σ is the noise level. Alternatively, the algorithm terminates when the maximum number of $400 \cdot n$ function evaluations is exceeded.

For each problem 50 independent runs are conducted. We considered a test run successful if the stopping criterium (23) is satisfied before exceeding the maximal number of function evaluations.

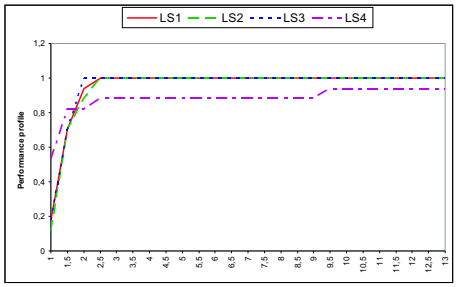
	LS2	LS3	LS4
$\sigma = 0.1$			
SGR	0.296	0.340	0.369
BFGS	0.312	0.383	0.402
SR1	0.281	0.334	0.354
$\sigma = 1$			
SGR	0.204	0.331	0.297
BFGS	0.164	0.391	0.404
SR1	0.187	0.345	0.268
$\sigma = 10$			
SGR	0.155	0.350	0.282
BFGS	0.069	0.408	0.377
SR1	0.108	0.361	0.305

Table 2: Nonmonotonicity index for different noise levels

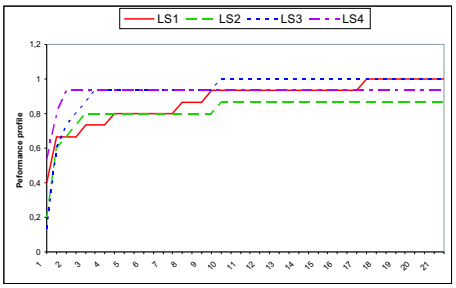
To compare the performance of methods we present the performance profiles defined in [9]. The measure for the performance profile is defined as the number of function evaluations as common for noisy problems. More precisely, let us denote by N_{ij} the number of successful runs out of 50 for the method i solving the problem j and let φ_{ij} be the average number of function evaluations needed



a. SGR methods



b. BFGS methods



c. SR1 methods

Figure 3: The performance profiles for different directions and $\sigma = 1$

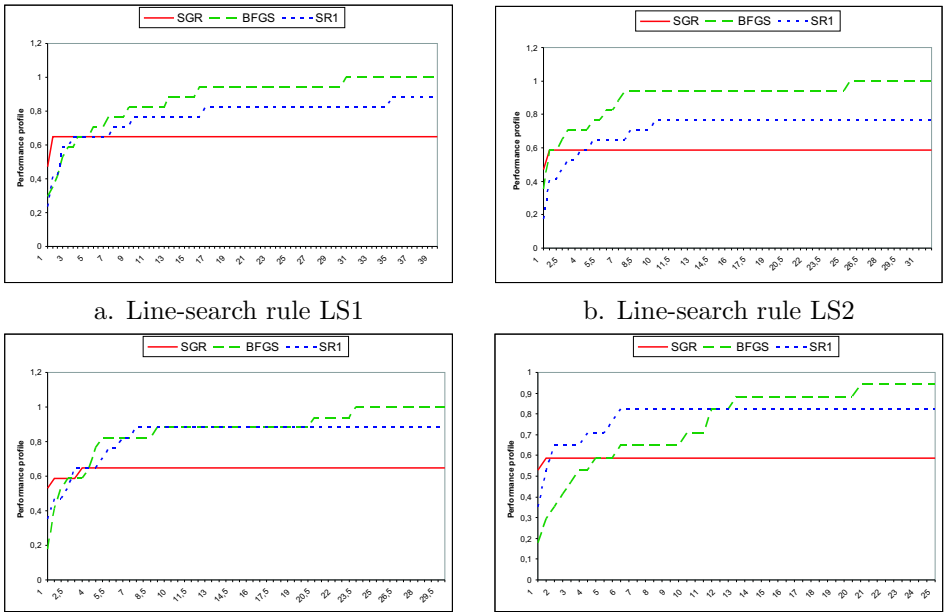
for the method i to solve the problem j, in the successful runs. Clearly the smaller φ_{ij} is the method i is more efficient. However in noisy environment one is also interested in the variation of φ_{ij} as it shows, roughly speaking, the robustness of the method. Thus we take the linear combination of φ_{ij} and the corresponding standard deviation $\sigma(\varphi)_{ij}$ for the performance measure i.e. the performance measure is

$$\pi_{ij} = \varphi_{ij} + \sigma(\varphi)_{ij}.$$

Figures 1- 6 show the performance profiles for problems given in Table 1, that are solved at least by one of the methods presented on the graph. The number of solved problems by the methods presented on the performance profile graph are reported for each of the graphs. Besides the graphs presented here a number of additional indicators is available at `http://people.dmi.uns.ac.rs/~natasa/additional.pdf`. The graph below are made for two different comparisons - the comparison of different search directions for all four line search rules and the comparison of line search rules for each of the considered directions.

Starting from Figure 1, the set of problems that are solved in at least one run by the SGR and SR1 methods is 12 while the BFGS methods solved 14 of the considered 18 problems. As expected the nonmonotone line search rules clearly outperform the monotone Armijo rule with noisy values for all three directions. The performance profile presented at Figure 2 compare each of the rules for all three directions. The monotone LS1 solved 13 problems while the remaining 3 methods solved 14 problems out of the test collection. For this noise level ($\sigma = 0.1$) the BFGS direction seems to be superior to both SGR and SR1.

Increasing the noise to $\sigma = 1$ we obtain the graphs shown at Figure 3 and Figure 4. Again the number of solved problems differs, ranging from 11 solved



a. Line-search rule LS1

b. Line-search rule LS2

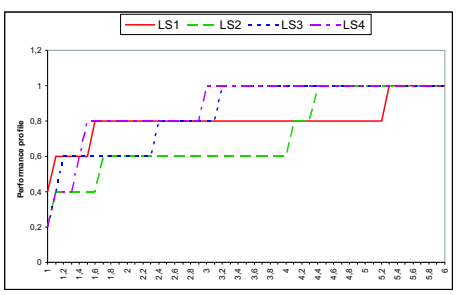
c. Line-search rule LS3

d. Line-search rule LS4

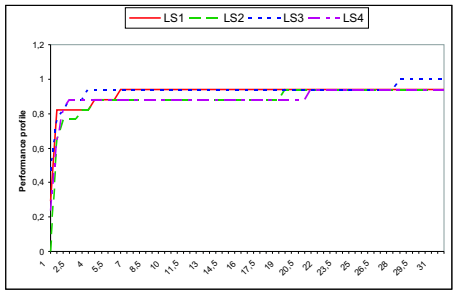
Figure 4: The performance profiles for different line search rules and $\sigma = 1$

by the SGR methods, 15 solved by the SR1 methods and 17 solved by the BFGS methods. Although the influence of the line search choice is less clear than for $\sigma = 0.1$, it is again evident that the nonmonotone rules perform better than the monotone one. Looking at Figure 4, one can clearly notice that the SGR direction perform rather poorly with LS1 which is consistent with the results obtained in the classical case without noise, [21] as SGR is known for its nonmonotone behavior and imposing the monotonicity request destroys its main advantages. This property is persistent in the noisy environment as well. For this level of noise the BFGS direction again appears to be the most suitable one.

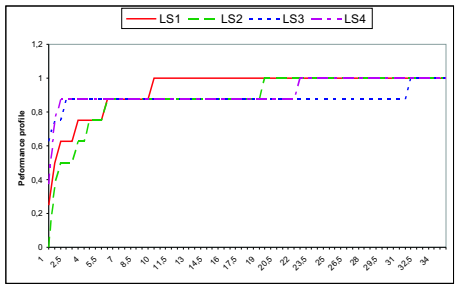
Finally, Figure 5 and 6 show the performance profiles for a large level of noise generated with $\sigma = 10$. The existing difference in the number of solved problems, ranging from 5 for the SGR methods, 8 for the SR1 methods and 17 for the BFGS methods again favors BFGS methods. The difference in line search rules is less evident for this direction than it is for the other two, with SGR strongly preferring nonmonotone rules and SR1 mildly preferring the same. Nevertheless the graphs at Figure 5b again favor the use of nonmonotone rules even for the BFGS direction. The graphs presented at Figure 6 contain the performance profiles for all four line search rules and 17 problems. An interesting point we have noticed is that SGR might be the worst direction for this performance measure but if it is converging than the number of function evaluations needed for convergence is significantly smaller than the number of function evaluations needed by the other two directions. The lack of success in many runs is somehow contrary to its behavior in noise-free optimization, where it tends to be very competitive, see for example [5] and [6]. Our conjecture is that there might be



a. SGR methods



b. BFGS methods



c. SR1 methods

Figure 5: The performance profiles for different directions and $\sigma = 10$

a need to adjust the spectral coefficient to the noisy environment and future research is planned in this direction.

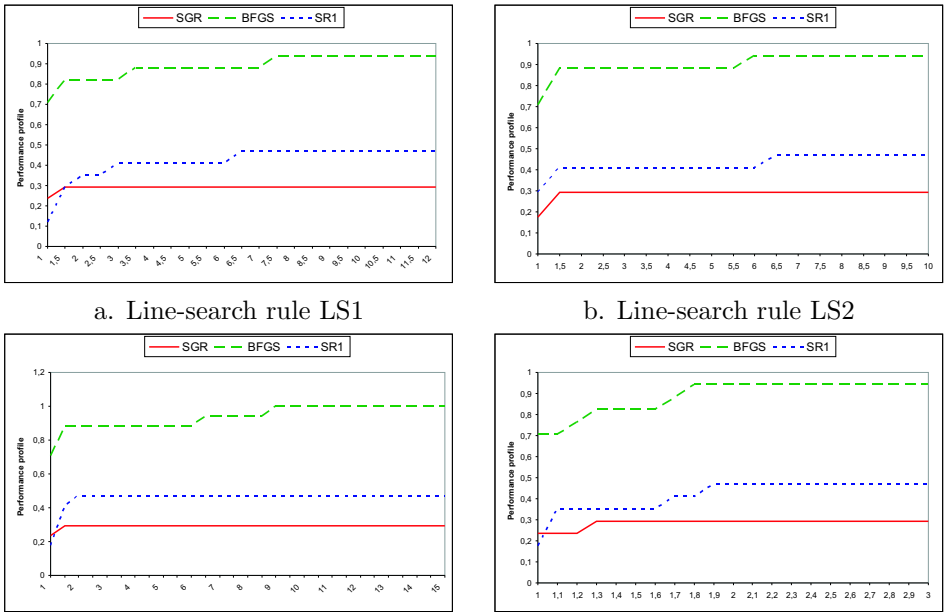
One of the conclusions we can drawn from this set collection is that nonmonotonicity improves the behavior of all tested directions. Thus we report the nonmonotonicity index in Table 2 as a measure of the influence of nonmonotonicity, [13]. The index is defined as the average ratio of the number of iterations with the step size that would not be accepted if the line search was LS1 and the total number of iterations. The average calculated over all successful runs in all tested problems. The values of this index clearly show that the number of accepted step lengths is significant for all three nonmonotone line search rules.

References

[1] S. Andradottir, *A Scaled Stochastic Approximation Algorithm*, Management Science 42(4) (1996), 475-498.

[2] Barzilai, J., Borwein, J. M., *Two point step size gradient methods*, IMA J. Numer. Anal. 8 (1988), 141–148

[3] S. C. Billups, J. Larson, P. Graf, *Derivative-Free Optimization of Expensive Functions with Computational Error Using Weighted Regression,* SIAM J. Optim., Vol. 23, No. 1 (2013),pp.27-53



a. Line-search rule LS1

b. Line-search rule LS2

c. Line-search rule LS3

d. Line-search rule LS4

Figure 6: The performance profiles for different line search rules and $\sigma = 10$

[4] Birgin, E.G., , Krejić, N., Martínez, J.M., *Globally convergent inexact quasi-Newton methods for solving nonlinear systems*, Numerical Algorithms 32, 2-4 (2003), 249-260.

[5] W. Cheng, D. H. Li, *A derivative-free nonmonotone line searh and its application to the spectral residual method,* IMA Journal of Numerical Analysis, Vol. 29 (2009), pp.814-825

[6] W. L. Cruz, J. M. Martinez, M. Raydan, *Spectral residual method without gradient information for solving large-scale nonlinear systems of equations,* Math. Comput, Vol. 75, No. 255 (2006), pp.1429-1448

[7] J. E. Dennis Jr., R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Philadelphia, SIAM, 1996

[8] M. A. Diniz-Ehrhardt, J. M. Martínez, M. Raydan, *A derivative free nonmonotone line-search technique for unconstrained optimization,* J. Comp. Appl. Math., Vol. 219, No. 2 (2008), pp.383-397

[9] E. D. Dolan, J. J. Moré, *Benchmarking optimization software with performance profiles,* Math. Program., Ser. A, Vol. 91 (2002), pp.201-213

[10] M. C. Fu, Gradient Estimation, *S.G. Henderson and B.L. Nelson (Eds.), Handbook in OR & MS Vol. 13 (2006), pp. 575-616.*

[11] L. Grippo, F. Lampariello, S. Lucidi, *A nonmonotone line search technique for Newton's method,* SIAM. J. Numer. Anal., Vol. 23 (1986), pp.707-716

[12] Kelley, C. T., *Iterative Methods for Optimization*, 1999, SIAM

[13] N. Krejić, N. Krklec Jerinkić *Nonmonotone line search methods with variable sample size,* http://www.optimization-online.org/DB_HTML/2013/05/3902.html

[14] N. Krejić, J. Lužanin, I. Stojkovska *A gradient method for unconstrained optimization in noisy environment,* Applied Numerical Mathematics, Vol. 70 (2013), pp.1-21

[15] N. Krejić, S. Rapajić *Gllobaly convergent Jacobian smoothing inexact Newton methods for NCP,* Computational Optimization and Applications, Vol. 41, No. 2 (2008), pp.243-261

[16] D. H. Li, M. Fukushima, *A derivative-free line search and global convergence of Broyden-like method for nonlinear equations,* Opt. Methods Software, Vol. 13 (2000), pp.181-201

[17] S. Lucidi, M. Sciandrone, *A derivative-free algorithm for bound constrained optimization,* Computational Optimization and Applications, Vol. 21 (2002), pp.119-142

[18] Moré, J. J., Garbow, B. S., Hillstrom, K. E. *Testing unconstrained optimization software*, ACM Transactions on Mathematical Software (1981), 7(1), 17–41

[19] F. Nikolovski, I. Stojkovska, *New derivative-free nonmonotone line search methods for unconstrained minimization*, Proceedings of the Fifth International Scientific Conference - FMNS2013, Vol.1 Mathematics and Informatics (2013), pp.47-53

[20] Nocedal, J., Wright, S. J. *Numerical Optimization. Second edition*, New York, Springer-Verlag, 2006.

[21] Raydan, M., *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Opt. 7 (1997), 26–33

[22] J. C. Spall, *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control,* John Wiley & Sons, Inc., Hoboken, New Jersey, 2003

[23] Z. Xu, Y-H. Dai, *New stochastic approximation algorithms with adaptive step size,* Optim. Lett., Vol. 6, No. 8 (2012), pp.1831-1846

[24] Z. Yu, D. Pu, *A new nonmonotone line search technique for unconstrained optimization,* J. Comp. Appl. Math., Vol. 219 (2008), pp.134-144

[25] B. Zhang, Z. Zhu, S. Li, *A modified spectral conjugate gradient projection algorithm for total variation image restoration,* Appl. Math. Lett., Vol. 27 (2014), pp.26-35

[26] H. Zhang, W. W. Hager, *A nonmonotone line search technique and its application to unconstrained optimization,* SIAM J. Optim., Vol. 14, No. 4 (2004), pp.1043-1056