Retaining Convergence Properties of Trust Region Methods Without Extra Gradient Evaluations [1]

Humaid Fayez Khalfan[2], Richard H. Byrd [3], Robert B. Schnabel[3]

August 1998

University of Colorado at Boulder
Department of Computer Science
Campus Box 430
University of Colorado
Boulder, Colorado 80309-0430 USA

[1]Research supported by Air Force Office of Scientific Research grant F49620-97-1-0164, Army Research Office grant DAAL DAAG55-98-1-0176, and National Science Foundation grant CCR-9101795.

[2]Department of Mathematics and Computer Science, United Arab Emirates University, Al-Ain, United Arab Emirates.
[3]Department of Computer Science, Campus Box 430, University of Colorado, Boulder, Colorado 80309 U.S.A.

Retaining Convergence Properties of Trust Region Methods Without Extra Gradient Evaluations

Humaid Fayez Khalfan, Richard H. Byrd, Robert B. Schnabel

Abstract

Several recent computational studies have shown that trust-region quasi-Newton methods using the SR1, PSB, and BFGS updates are effective methods for solving unconstrained optimization problems. In addition, the analyses in Powell [1975] and Byrd, Khalfan, and Schnabel [1993] demonstrate strong convergence properties for some trust-region quasi-Newton methods. A computational disadvantage of the methods analyzed in these papers, for which the strongest convergence properties among trust-region quasi-Newton methods have been shown, is that the update at rejected trial points requires a gradient evaluation that would not otherwise be made. In this paper, we consider a modification of these methods that makes a different, less expensive update at rejected trial points. In particular, we propose a modification of the PSB method that uses only the function value at rejected points to make the update at those points. We then show how to modify Powell's analysis of the PSB method to prove the same q-superlinear convergence result for the new method. Finally, we discuss the issues and difficulties involved in extending this approach to trust regions methods using updates in the Broyden class, such as the BFGS, SR1, and DFP.

1. Introduction

Several recent computational studies (Conn, Gould, and Toint [1988, 1992], Khalfan, Byrd, and Schnabel [1993]) have shown that trust-region quasi-Newton methods using the SR1, PSB, and BFGS updates are effective methods for solving the unconstrained optimization problem

$$\text{minimize } f(x), \quad x \in R^n.$$

In addition, the analyses in Powell [1975] and Byrd, Khalfan, and Schnabel [1993] demonstrate strong convergence properties for some trust-region quasi-Newton methods. Namely, they show that for the PSB, DFP, and SR1 updates, the rate of convergence is q-superlinear. These analyses are made under realistic assumptions, without any assumption of linearly independent steps or positive definite Hessian approximations. They both require, however, that an update be made at all rejected as well as accepted trial points. Some feature of this sort seems essential for a trust-region quasi-Newton method to have good theoretical properties, for without it, the method does not improve on its approximation in a bad step direction, and might attempt an infinite number of bad steps in this direction.

A computational disadvantage of these methods that require an update at rejected trial points is that this update requires a gradient evaluation that would not otherwise need to be made. In this paper, we consider a modification of these methods that makes a different, less expensive update at rejected trial points. In particular, we propose a modification of the PSB method that uses only the function value at rejected points to make the update at those points. (The function value is of course already required in order to decide whether to reject the point in question.) We then show how to modify Powell's analysis of the PSB method (Powell [1975]) to prove the same q-superlinear convergence result for the new method. We have focussed on the PSB update, primarily because it is not clear how to adapt this approach efficiently to trust regions methods using updates in the Broyden class, such as the BFGS, SR1, and DFP. In the last section, we discuss the issues and difficulties involved in attempting such an extension. Throughout this paper, unless otherwise spesified, $\|.\|$ represents the l_2 vector norm or matrix norm.

In trust region methods, at the current iterate x_k, a trial step s_k is obtained by minimizing the local quadratic model

$$m(x_k + s) = f(x_k) + \nabla f(x_k)^T s + \frac{1}{2} s^T B_k s \tag{1.1}$$

over the region $\|s\| \leq \Delta_k$, where B_k is an approximation to the Hessian matrix, and Δ_k is a positive number representing the radius of the sphere around the current point in which $m(x_k + s)$ is trusted to represent the objective function. The step s_k is accepted if the ratio of the actual reduction in functional values,

$$\text{ared}_k = f(x_k) - f(x_k + s_k),$$

to the reduction predicted by the quadratic model,

$$\text{pred}_k = f(x_k) - m(x_k + s_k),$$

has an acceptably large positive value. If not, the trust region radius is reduced and the quadratic model is minimized again on the reduced region. For a background on trust region algorithms and their global convergence see for example, Moré and Sorensen [1981] and Shultz, Schnabel, and Byrd [1985].

Powell [1975] suggests a general quasi-Newton trust region algorithm which, in the case of the PSB update, is equivalent to the following algorithm.

Algorithm 1.1: PSB – Trust Region Method.

Step 0 Select algorithm parameters, $\eta_1 \in (0,1)$, $\eta_2 \geq 1$, and $0 \leq \eta_3 \leq \eta_4 \leq 1$, and $\bar{\Delta} > 0$. Choose an initial point x_1, an initial positive definite matrix B_1, and an initial trust region radius Δ_1, and set $k = 1$.

Step 1 If a convergence criterion is achieved, then stop.

Step 2 Compute the step $s_k(\Delta_k)$ by solving approximately

$$\begin{array}{ll} \text{minimize} & m(x_k + s) = f(x_k) + \nabla f(x_k) s^T + \frac{1}{2} s^T B_k s \\ \text{subject to} & \|s\| \leq \Delta_k. \end{array}$$

Step 3 If $f(x_k + s_k) < f(x_k)$,

then $x_{k+1} = x_k + s_k$;

else $x_{k+1} = x_k$.

Step 4 Determine the next trust region radius:
compute $\text{ared}_k = f(x_k) - f(x_k + s_k)$ and $\text{pred}_k = f(x_k) - m(x_k + s_k)$;
if $\dfrac{\text{ared}_k}{\text{pred}_k} \geq \eta_1$,

then choose $\Delta_{k+1} \in [\|s_k\|, \eta_2 \|s_k\|]$;

else choose $\Delta_{k+1} \in [\eta_3 \|s_k\|, \eta_4 \|s_k\|]$;

set $\Delta_{k+1} = \min\{\Delta_{k+1}, \bar{\Delta}\}$.

Step 5 Update B_k:

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k) s_k^T + s_k (y_k - B_k s_k)^T}{s_k^T s_k} - \frac{(y_k - B_k s_k)^T s_k s_k s_k^T}{(s_k^T s_k)^2}$$

where $y_k = \nabla f(x_k + s_k) - \nabla f(x_k)$.

Step 6 Set $k = k + 1$, and go to Step 1.

We notice in this algorithm that, although a point $x_k + s_k$ may be rejected in Step 3, derivative information at this point is still used in updating the Hessian approximation by the PSB formula in Step 5.

The modification of this algorithm that we will consider in this paper is the following. Instead of updating B_k using the PSB update at rejected points, we use a different update that does not require evaluation of the gradients at these points. (If a finite difference gradient is used, this saves us n function evaluations at each of these iterations, which may be important when function evaluation is quite expensive.) This update is determined by requiring that the next Hessian approximation matrix, B_{k+1}, satisfy the scalar equation

$$s_k^T B_{k+1} s_k = 2(f(x_k + s_k) - f(x_k) - s_k^T \nabla f(x_k)). \tag{1.2}$$

This condition is motivated by the fact that from Taylor series approximation,

$$s_k^T \nabla^2 f(x_k) s_k \approx 2(f(x_k + s_k) - f(x_k) - s_k^T \nabla f(x_k)),$$

and the fact that (1.2) causes the updated quadratic model around x_k,

$$m(x_k + s) = f(x_k) + \nabla f(x_k)^T s + \frac{1}{2} s^T B_{k+1} s,$$

to interpolate $f(x_k + s_k)$.

As usual in quasi-Newton updates, the equation (1.2) does not uniquely determine B_{k+1}, and so we choose B_{k+1} to be the smallest change to B_k, in the Frobenius norm, $\|.\|_F$, which is consistent with (1.2). In general, given $B_c \in R^{n\times n}$, $s \in R^n$, and $\alpha \in R$, the solution to

$$\min \|B - B_c\|_F \quad \text{subject to} \quad s^T B s = \alpha \tag{1.3}$$

is given by

$$B_+ = B_c + (\alpha - s^T B_c s)\frac{ss^T}{(s^T s)^2}. \tag{1.4}$$

To see this, let $A \in R^{n\times n}$ with $s^T A s = \alpha$. Then,

$$\begin{aligned}
\|B_+ - B_c\|_F &= |(\alpha - s^T B_c s)/s^T s| \left\|\frac{ss^T}{s^T s}\right\|_F \\
&= |(s^T A s - s^T B_c s)/s^T s| \\
&= |[s^T (A - B_c) s]/s^T s| \\
&\leq \|A - B_c\|_F.
\end{aligned}$$

Thus, the update that we will consider, which we will refer to as the scalar update, is given by

$$B_{k+1} = B_k + (2(f(x_k + s_k) - f(x_k) - s_k^T \nabla f(x_k)) - s_k^T B_k s_k)\frac{s_k s_k^T}{(s_k^T s_k)^2}. \tag{1.5}$$

Using this update at rejected points, we have the following algorithm which differs from Algorithm 1.1 only in Step 5.

Algorithm 1.2 : Modified PSB – Trust Region Method.

Step 0 Select algorithm parameters, $\eta_1 \in (0,1)$, $\eta_2 \geq 1$, and $0 \leq \eta_3 \leq \eta_4 \leq 1$, and $\bar{\Delta} > 0$. Choose an initial point x_1, an initial positive definite matrix B_1, and an initial trust region radius Δ_1, and set $k = 1$.

Step 1 If a convergence criterion is achieved, then stop.

Step 2 Compute the step $s_k(\Delta_k)$ by solving approximately

$$\begin{array}{ll}
\text{minimize} & m(x_k + s) = f(x_k) + \nabla f(x_k) s^T + \frac{1}{2} s^T B_k s \\
\text{subject to} & \|s\| \leq \Delta_k.
\end{array}$$

Step 3 If $f(x_k + s_k) < f(x_k)$,

then $x_{k+1} = x_k + s_k$;

else $x_{k+1} = x_k$.

Step 4 Determine next trust region radius:
compute $\text{ared}_k = f(x_k) - f(x_k + s_k)$ and $\text{pred}_k = f(x_k) - m(x_k + s_k)$;
if $\dfrac{\text{ared}_k}{\text{pred}_k} \geq \eta_1$,
then choose $\Delta_{k+1} \in [\|s_k\|, \eta_2\|s_k\|]$;
else choose $\Delta_{k+1} \in [\eta_3\|s_k\|, \eta_4\|s_k\|]$;
set $\Delta_{k+1} = \min\{\Delta_{k+1}, \bar{\Delta}\}$.

Step 5 If $f(x_k + s_k) < f(x_k)$, then

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)s_k^T + s_k(y_k - B_k s_k)^T}{s_k^T s_k} - \frac{(y_k - B_k s_k)^T s_k s_k s_k^T}{(s_k^T s_k)^2},$$

else

$$B_{k+1} = B_k + 2(f(x_k + s_k) - f(x_k) - s_k^T \nabla f(x_k) - \frac{1}{2} s_k^T B_k s_k) \frac{s_k s_k^T}{(s_k^T s_k)^2}.$$

Step 6 Set $k = k + 1$ and go to Step 1.

In Step 2 of Algorithm 1.2 we require that the trust region subproblem is solved approximately by some method like the hook-step or dogleg (see e.g. Dennis and Schnabel [1983]), so that the resulting steps satisfy the condition

$$\text{pred}_k \geq \eta \|\nabla f(x_k)\| \min\{\|s_k\|, \|\nabla f(x_k)\|/\|B_k\|\}, \tag{1.6}$$

where η is a positive constant. In order to assure superlinear convergence, we will also assume that the step be a quasi-Newton step whenever the trust region constraint is not active, i.e. that $\|s_k\| < \Delta_k$ only when $s_k = B_k^{-1}\nabla f(x_k)$. Virtually all trust region algorithms that are used in practice, as well as the ideal trust region step selection strategy of choosing s_k to be the exact solution of the trust subproblem, satisfy these two requirements, at least approximately (see e.g. Shultz, Schnabel and Byrd [1985]).

2. Convergence of the New Method

Powell [1975] shows that using any sequence of Hessian approximating matrices, $\{B_k\}$, satisfying

$$\|B_{k+1}\| \leq \beta_1 + \beta_2 \sum_{j=1}^{k} \|s_j\|, \tag{2.1}$$

the sequence of iterates produced by an algorithm of the form of Algorithm 1.2 has the property

$$\liminf \|\nabla f(x_k)\| = 0, \tag{2.2}$$

provided that $f(x)$ is differentiable and bounded below, $\nabla f(x)$ is uniformly continuous, and the steps produced by the trust region strategy satisfy condition (1.6). Fletcher [1972], in a result he credits to Powell, shows that the PSB update satisfies (2.1). We will show that the scalar update also satisfies (2.1) and thus (2.2) is true for Algorithm 1.2.

Powell [1975] also shows that if $x_k \to x_*$, then q-superlinear convergence can be established for updates satisfying (2.1), provided that, in addition to some assumptions on $f(x)$ around x_*, the condition

$$\frac{\|y_k - B_k s_k\|}{\|s_k\|} \to 0, \tag{2.3}$$

which holds for the PSB update, is satisfied. For Algorithm 1.2, we will show that the same result can be obtained by showing that the algorithm satisfies

$$\frac{|s_k^T(y_k - B_k s_k)|}{\|s_k\|^2} \to 0 \tag{2.4}$$

when the scalar update is used (and (2.3) otherwise). This condition will allow us to show that asymptotically there are no rejected points, implying that asymptotically (2.3) holds at all iterations.

We now make the following assumptions which will be needed in the following analysis.

Assumptions

(A1) The sequence of points $\{x_k\}$ generated by Algorithm 1.2 converges to the point x_*.

(A2) $f(x)$ is twice continuously differentiable in a neighborhood D of x_*, where $\nabla^2 f(x_*)$ is positive definite, and $\nabla^2 f(x)$ is Lipschitz continuous in D. That is, $\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq \gamma\|x - y\|$ for all $x, y \in D$.

(A3) The steps generated by Algorithm 1.2 satisfy condition (1.6).

Under these assumptions, we will prove q-superlinear convergence for Algorithm 1.2 as follows. We will first show that the sequence of matrices, $\{B_k\}$, generated by the algorithm satisfies certain bounded deterioration inequalities. Using this result, we will show that $\{B_k\}$ satisfies (2.1). Then using Powell's [1975] result that

$$\sum_{k=1}^{\infty} \|s_k\| < \infty, \tag{2.5}$$

which by (2.1) implies that the sequence $\{B_k\}$ is bounded, we will establish (2.4) for the scalar updates (and (2.3) for the standard PSB updates). Finally this result will be used to prove q-superlinear convergence.

In the following, S_1 denotes the set of iterations at which the PSB update is used, and S_2 the set of iterations at which the scalar update is used.

Lemma 2.1 Let $E_k = B_k - \nabla^2 f(x_k)$ and $E_{k+1} = B_{k+1} - \nabla^2 f(x_k + s_k)$, and suppose that assumptions (A1) and (A2) hold. Then

$$\|E_{k+1}\| \leq \|E_k\| - \frac{1}{2\|E_k\|}\left(\frac{\|E_k s_k\|}{\|s_k\|}\right)^2 + O(\|s_k\|) \tag{2.6}$$

for $k \in S_1$, and

$$\|E_{k+1}\| \leq \|E_k\| - \frac{1}{2\|E_k\|}\left(\frac{s_k^T E_k s_k}{\|s_k\|^2}\right)^2 + O(\|s_k\|) \tag{2.7}$$

for $k \in S_2$.

Proof. The proof of (2.6) is similar to the proof of a related result in Fletcher [1972]. If the PSB update is used at the kth iteration, then

$$E_{k+1} = P_k E_k P_k + \frac{v_k s_k^T + s_k v_k^T}{s_k^T s_k} - \frac{s_k s_k^T v_k^T s_k}{(s_k^T s_k)^2} + \nabla^2 f(x_k) - \nabla^2 f(x_k + s_k) \qquad (2.8)$$

where

$$P_k = \left(I - \frac{s_k s_k^T}{s_k^T s_k} \right)$$

and

$$v_k = y_k - \nabla^2 f(x_k) s_k.$$

By Lemma 8.2.5 of Dennis and Schnabel [1983], we have

$$\begin{aligned} \|E_k P_k\|_F &= \left\| E_k \left(I - \frac{s_k s_k^T}{s_k^T s_k} \right) \right\|_F \\ &\le \|E_k\|_F - \frac{1}{2\|E_k\|_F} \left(\frac{\|E_k s_k\|}{\|s_k\|} \right)^2 . \end{aligned}$$

Therefore, since $\|P_k\| = 1$ and $\|AB\|_F \le \|A\| \|B\|_F$, we have that

$$\begin{aligned} \|P_k E_k P_k\|_F &\le \|P_k\| \|E_k P_k\|_F & (2.9) \\ &\le \|E_k\| - \frac{1}{2\|E_k\|} \left(\frac{\|E_k s_k\|}{\|s_k\|} \right)^2 . & (2.10) \end{aligned}$$

Also, since

$$\begin{aligned} \|v_k\| &= \|y_k - \nabla^2 f(x_k) s_k\| \\ &\le \frac{\gamma}{2} \|s_k\|^2, \end{aligned}$$

(see for example Lemma 4.1.12 of Dennis and Schnabel [1983]) we have that

$$\begin{aligned} \left\| \frac{v_k s_k^T + s_k v_k^T}{s_k^T s_k} - \frac{s_k s_k^T v_k^T s_k}{(s_k^T s_k)^2} \right\| &\le \frac{3\|v_k\|}{\|s_k\|} \\ &\le \frac{3\gamma}{2} \|s_k\|. & (2.11) \end{aligned}$$

Therefore, since the Lipschitz continuity of $\nabla^2 f(x_k)$ implies that

$$\|\nabla^2 f(x_k + s_k) - \nabla^2 f(x_k)\| = O(\|s_k\|), \qquad (2.12)$$

(2.8) , (2.10) and (2.11) imply (2.6).

Suppose now, that the scalar update is used at the kth iteration instead. Then using (1.5), (2.12) and the fact that

$$f(x_k + s_k) - \left(f(x_k) + s_k^T \nabla f(x_k) + \frac{1}{2} s_k^T \nabla^2 f(x_k) s_k \right) = O(\|s_k\|^3)$$

(see for example Lemma 4.1.14 of Dennis and Schnabel [1983]), we have that

$$\begin{aligned}
\|E_{k+1}\| &\le \left\| E_k + (2(f(x_k+s_k) - f(x_k) - s_k^T \nabla f(x_k)) - s_k^T B_k s_k) \frac{s_k s_k^T}{\|s_k\|^4} \right\| \\
&\quad + \|\nabla^2 f(x_k+s_k) - \nabla^2 f(x_k)\| \\
&= \left\| E_k + (2(f(x_k+s_k) - f(x_k) - s_k^T \nabla f(x_k)) - s_k^T B_k s_k) \frac{s_k s_k^T}{\|s_k\|^4} \right\| + O(\|s_k\|) \\
&= \left\| E_k + (s_k^T \nabla^2 f(x_k) s_k - s_k^T B_k s_k + O(\|s_k\|^3)) \frac{s_k s_k^T}{\|s_k\|^4} \right\| + O(\|s_k\|) \\
&\le \left\| E_k - \left(\frac{s_k^T E_k s_k}{\|s_k\|^4} \right) s_k s_k^T \right\| + O(\|s_k\|) \left\| \frac{s_k s_k^T}{s_k^T s_k} \right\| + O(\|s_k\|) \\
&\le \left\| E_k - \left(\frac{s_k^T E_k s_k}{\|s_k\|^4} \right) s_k s_k^T \right\| + O(\|s_k\|). \qquad (2.13)
\end{aligned}$$

Now let $u_k = \left(\frac{s_k^T E_k s_k}{\|s_k\|^4} \right)^{1/2} s_k$. Then

$$\begin{aligned}
\left\| E_k - \left(\frac{s_k^T E_k s_k}{\|s_k\|^4} \right) s_k s_k^T \right\|^2 &= \|E_k - u_k u_k^T\|^2 \\
&= \mathrm{Tr}(E_k^T E_k - E_k u_k u_k^T - u_k u_k^T E_k + u_k u_k^T u_k u_k^T) \\
&= \|E_k\|^2 - 2(u_k^T E_k u_k)^2 + \|u_k\|^4 \\
&= \|E_k\|^2 - \frac{(s_k^T E_k s_k)^2}{\|s_k\|^4}. \qquad (2.14)
\end{aligned}$$

Therefore, since for any $\alpha \ge |\beta| \ge 0, (\alpha^2 - \beta^2)^{1/2} \le \alpha - \frac{\beta^2}{2\alpha}$, the last inequality implies that

$$\left\| E_k - \left(\frac{s_k^T E_k s_k}{\|s_k\|^4} \right) s_k s_k^T \right\| \le \|E_k\| - \frac{(s_k^T E_k s_k)^2}{2\|E_k\|\|s_k\|^4},$$

and so by (2.13), it follows that (2.7) is true.□

In the lemma below, we show that the sequence of matrices, $\{B_k\}$, generated by Algorithm 1.2 is majorized by the sequence of partial sums of the generated steps. This, of course, will imply that $\{B_k\}$ is bounded if $\sum_{k=1}^{\infty} \|s_k\| < \infty$.

Lemma 2.2 Suppose that assumptions (A1) and (A2) hold. Then the sequence of matrices generated by Algorithm 1.2 satisfies the inequality

$$\|B_{k+1}\| \le \beta_1 + \beta_2 \sum_{j=1}^{k} \|s_j\| \qquad (2.15)$$

where β_1 and β_2 are positive constants.

Proof. Since $x_k \to x_*$, we may assume that $\|\nabla^2 f(x_k)\| \leq \beta_3$ for all k, where β_3 is a positive constant. By Lemma 2.1, we have that

$$\|E_{k+1}\| \leq \|E_k\| + \beta_2 \|s_k\|$$

where β_2 is some positive constant. Therefore, applying this inequality k times, we have

$$\|E_{k+1}\| \leq \|E_1\| + \beta_2 \sum_{j=1}^{k} \|s_j\|.$$

Hence, using the triangle inequality we have that

$$\|B_{k+1}\| \leq \|\nabla^2 f(x_k + s_k)\| + \|E_1\| + \beta_2 \sum_{j=1}^{k} \|s_j\|,$$

and (2.15) follows with $\beta_1 = \beta_3 + \|E_1\|$. $\square$

Powell [1975] proved that the iterates generated by an algorithm of the from of Algorithm 1.2 satisfy

$$\sum_{k=1}^{\infty} \|s_k\| < \infty. \qquad (2.16)$$

The only property about the matrices B_k that the proof of this result requires is that the sequence $\{B_k\}$ generated by the algorithm satisfy (2.15). Since the PSB update and the scalar update satisfy this requirement, (2.16) is true for Algorithm 1.2

Lemma 2.3 Suppose that assumptions (A1), (A2) and (A3) hold. Then the iterates generated by Algorithm 1.2 satisfy

$$\sum_{k=1}^{\infty} \|s_k\| < \infty, \qquad (2.17)$$

and the sequence $\{B_k\}$ is bounded.

Proof. The hypotheses of this lemma together with (2.15) are the hypotheses of Theorem 2 of Powell [1975], which gives (2.17). Boundedness of $\{B_k\}$ follows immediately from (2.17) and (2.15).

Now we use this lemma to show that (2.4) is true.

Lemma 2.4 Suppose that assumptions (A1), (A2) and (A3) hold. Then the iterates generated by Algorithm 1.2 satisfy

$$\lim_{k\to\infty} \frac{|s_k^T(y_k - B_k s_k)|}{\|s_k\|^2} = 0, \qquad (2.18)$$

and for $k \in S_1$,

$$\lim_{k\to\infty} \frac{\|y_k - B_k s_k\|}{\|s_k\|} = 0. \qquad (2.19)$$

Proof. The proof is related to the proof of a result in Fletcher [1972] for the PSB update. Since the sequence $\{x_k\}$ converges to x_*, we may assume, without loss of generality, that $\|\nabla^2 f(x_k)\|$ is bounded. Therefore, by the boundedness of $\{B_k\}$ there exists ξ such that for all k,

$$\begin{aligned} \|E_k\| &= \|B_k - \nabla^2 f(x_k)\| && (2.20)\\ &\leq \xi/2. && (2.21) \end{aligned}$$

Also, by Lemma 2.1 there are positive constants α_1 and α_2 such that

$$\begin{aligned} \left(\frac{\|E_k s_k\|}{\|s_k\|}\right)^2 &\leq 2\|E_k\|(\|E_k\| - \|E_{k+1}\| + \alpha_1\|s_k\|) \\ &\leq \xi(\|E_k\| - \|E_{k+1}\| + \alpha_1\|s_k\|) && (2.22) \end{aligned}$$

for $k \in S_1$, and similarly

$$\left(\frac{s_k^T E_k s_k}{\|s_k\|^2}\right)^2 \leq \xi(\|E_k\| - \|E_{k+1}\| + \alpha_2\|s_k\|) \qquad (2.23)$$

for $k \in S_2$. Therefore, summing the left and right sides of (2.22) and (2.23) for $k = 1, 2, ..., m$, we have that,

$$\sum_{k\in S_1}^{m}\left(\frac{\|E_k s_k\|}{\|s_k\|}\right)^2 + \sum_{k\in S_2}^{m}\left(\frac{s_k^T E_k s_k}{\|s_k\|^2}\right)^2 \leq \xi(\|E_1\| - \|E_{m+1}\| + \alpha_3\sum_{k=1}^{m}\|s_k\|) \qquad (2.24)$$

where $\alpha_3 = \max\{\alpha_1, \alpha_2\}$. By (2.17) this implies that,

$$\sum_{k\in S_1}^{\infty}\left(\frac{\|E_k s_k\|}{\|s_k\|}\right)^2 < \infty \qquad (2.25)$$

and

$$\sum_{k\in S_2}^{\infty}\left(\frac{s_k^T E_k s_k}{\|s_k\|^2}\right)^2 < \infty. \qquad (2.26)$$

Therefore, for $k \in S_1$, we have that

$$\begin{aligned} \|E_k s_k\|^2/\|s_k\|^2 &= \|(B_k - \nabla^2 f(x_k))s_k\|^2/\|s_k\|^2 \\ &= \|y_k - B_k s_k\|^2/\|s_k\|^2 + O(\|s_k\|) \\ &\to 0, \end{aligned}$$

and this implies (2.18) and (2.19) for $k \in S_1$. For $k \in S_2$ we have that

$$\begin{aligned} |s_k^T E_k s_k|/\|s_k\|^2 &= |s_k^T \nabla^2 f(x_k)s_k - s_k B_k s_k|/\|s_k\|^2 \\ &= |s_k^T \nabla^2 f(x_k)s_k - s_k^T y_k + s_k^T y_k - s_k^T B_k s_k|/\|s_k\|^2 \\ &\geq |s_k^T(y_k - B_k s_k)|/\|s_k\|^2 - |s_k^T(y_k - \nabla^2 f(x_k)s_k)|/\|s_k\|^2. \end{aligned}$$

Therefore, since the last term on the right side $\to 0$, and by (2.26) the term on the left side $\to 0$, (2.18) follows. $\square$

The next two lemmas show that $\dfrac{\text{ared}_k}{\text{pred}_k} \to 1$, which will be used to show that the trust region does not shrink after a finite number of iterations. The techniques of the proofs of the following two results are similar to those given in Powell [1975] except that we use (2.18) instead of (2.19).

Lemma 2.5 Suppose that assumptions (A1), (A2) and (A3) hold. Then there exists k_0 such that for all $k \geq k_0$, the iterates generated by Algorithm 1.2 satisfy

$$\text{pred}_k \geq \alpha \|s_k\|^2$$

where α is a positive constant.

Proof. From assumption (A2) on $f(x)$, there exist positive constants, μ_1 and μ_2, and a neighborhood of x_*, $\hat{D} \subset D$ such that

$$\mu_1 \|z\|^2 \leq z^T \nabla^2 f(x) z \leq \mu_2 \|z\|^2 \tag{2.27}$$

for all $z \in \hat{D}$ and $z \in R^n$. By assumption (A1), we may assume that $x_k \in \hat{D}$, for all k. Therefore, using Lemma 2.4 there exists k_0 such that for all $k \geq k_0$,

$$|s_k^T (y_k - B_k s_k)| \leq \frac{1}{2} \mu_1 \|s_k\|^2. \tag{2.28}$$

Using (2.27) we have that

$$\begin{aligned} s_k^T y_k &= \int_0^1 s_k^T \nabla^2 f(x_k + \tau s_k) s_k d\tau \\ &\geq \mu_1 \|s_k\|^2. \end{aligned}$$

Therefore,

$$\begin{aligned} f(x_k) - m(x_k + s_k) &= -s_k^T \nabla f(x_k) - \frac{1}{2} s_k^T B_k s_k \\ &= \frac{1}{2} s_k^T (y_k - B_k s_k) - \frac{1}{2} s_k^T y_k - s_k^T \nabla f(x_k) \\ &\leq \frac{1}{4} \mu_1 \|s_k\|^2 - \frac{1}{2} \mu_1 \|s_k\|^2 - s_k^T \nabla f(x_k) \\ &= -s_k^T \nabla f(x_k) - \frac{1}{4} \mu_1 \|s_k\|^2. \end{aligned}$$

But, since $f(x_k) - m(x_k + s_k) > 0$, this implies that

$$-s_k^T \nabla f(x_k) \quad \geq \quad \frac{1}{4} \mu_1 \|s_k\|^2,$$

and, hence, $\|\nabla f(x_k)\| \geq \frac{1}{4} \mu_1 \|s_k\|$. Therefore, since (2.17) and Lemma 2.2 implies that $\|B_k\| \leq \xi$, where $\xi > 0$, we have from assumption (A3) that

$$\begin{aligned} \text{pred}_k &\geq \eta \|\nabla f(x_k)\| \min\{\|s_k\|, \|\nabla f(x_k)\| / \|B_k\|\} \\ &\geq (\eta \mu_1 / 4) \|s_k\| \min\{\|s_k\|, (1/4) \mu_1 \|s_k\| / \|B_k\|\} \\ &\geq (\eta \mu_1 / 4) \|s_k\| \min\{\|s_k\|, (1/4) \mu_1 \|s_k\| / \xi\} \\ &\geq \alpha \|s_k\|^2 \end{aligned}$$

where α is a positive constant. $\square$

Lemma 2.6 Suppose that assumptions (A1), (A2) and (A3) hold. Then the iterates generated by Algorithm 1.2 satisfy

$$\lim_{k\to\infty} \frac{\text{ared}_k}{\text{pred}_k} = 1. \tag{2.29}$$

Proof. Since

$$\begin{aligned} \text{ared}_k &= f(x_k) - f(x_k + s_k) \\ &= -s_k^T \nabla f(x_k) - \frac{1}{2} s_k^T \nabla^2 f(x_k) s_k + O(\|s_k\|^3) \\ &= -s_k^T \nabla f(x_k) - \frac{1}{2} s_k^T y_k + O(\|s_k\|^3), \end{aligned}$$

and $\text{pred}_k = -s_k^T \nabla f(x_k) - \frac{1}{2} s_k^T B_k s_k$, we have that

$$|\text{ared}_k - \text{pred}_k| \le |s_k^T (y_k - B_k s_k)| + O(\|s_k\|^3). \tag{2.30}$$

Therefore, by Lemma 2.5 this implies that

$$\left| \frac{\text{ared}_k}{\text{pred}_k} - 1 \right| \le \frac{|s_k^T (y_k - B_k s_k)|}{\alpha \|s_k\|^2} + O(\|s_k\|) \tag{2.31}$$

where α is a positive constant. By Lemma 2.4, the limit of the first term on the right is 0, and so (2.29) follows. □

We now show that Algorithm 1.2 is q-superlinearly convergent.

Theorem 2.7 Suppose that assumptions (A1), (A2) and (A3) hold. Assume also, that in Algorithm 1.2 $s_k < \Delta_k$ only if $s_k = -B_k^{-1} \nabla f(x_k)$. Then the sequence $\{x_k\}$ generated by Algorithm 1.2 converges q-superlinearly to x_*.

Proof. By Lemma 2.6 we have that

$$\frac{\text{ared}_k}{\text{pred}_k} \to 1.$$

Therefore, there exists k_1 such that for all $k \ge k_1$, $\Delta_k \ge \Delta_{k_1}$. Since $\sum_{k=1}^{\infty} \|s_k\| < \infty$, there exists $k_2 \ge k_1$ such that

$$\|s_k\| < \Delta_k$$

for all $k \ge k_2$. Therefore, for all $k \ge k_2$, $s_k = -B_k^{-1} \nabla f(x_k)$. That is, the quasi-Newton step is accepted for all $k \ge k_2$, and thus the algorithm reduces to the standard PSB method. Therefore, using (2.19), q-superlinear convergence follows from Theorem 3 of Powell[1975]. □

3. Discussion

The previous two sections show that it is possible to modify a trust region PSB method so that it has strong convergence properties, *without* requiring an extra gradient evaluation at rejected trial points. Unfortunately, however, the PSB method is not regarded as a leading computational method in comparison to methods in the Broyden class, including the BFGS and the SR1. The main drawback of the PSB appears to be its lack of the invariance property under linear transformations of the variable space that the entire Broyden class has. This seems to lead to computational difficulties, especially on badly scaled problems. (For an example of the computational difficulties of the PSB method relative to the BFGS, see p. 207 of Dennis and Schnabel [1983].) We should note, however, that in tests in a trust region context by Conn, Gould, and Toint [1992], the PSB method was competitive with the BFGS method. However, Conn, Gould, and Toint [1992] speculate in this paper that the good performance of the PSB may be due to the fact that the test set is mainly well-scaled.

An obvious question, therefore, is whether the scalar updating technique that is studied in this paper can be applied to members of the Broyden class. There appears to be a fundamental problem in doing so. To illustrate this, consider the DFP update,

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)y_k^T + y_k(y_k - B_k s_k)^T}{y_k^T s_k} - \frac{(y_k - B_k s_k)^T s_k y_k y_k^T}{(y_k^T s_k)^2}.$$

Using the analysis technique introduced in Broyden, Dennis, and Moré [1973], it is straightforward to show that the analysis of Powell [1975] applies to this method. The key change is to use the weighted Frobenius form,

$$\|E\|_F = \|\nabla^2 f(x_*)^{-\frac{1}{2}} E \nabla^2 f(x_*)^{-\frac{1}{2}}\|_F,$$

instead of the unweighted Frobenius norm, to prove an analogous result to (2.6). Using the same technique, it is straightforward to show that if one uses the DFP analog of the scalar update (1.5),

$$B_{k+1} = B_k + (2(f(x_+ s_k) - f(x_k) - s_k^T \nabla f(x_k)) - s_k^T B_k s_k)\frac{y_k y_k^T}{(y_k^T s_k)^2} \tag{3.1}$$

at rejected trial points, then the theoretical analysis and results of Section 2 go through.

But even though this DFP analog of the method analyzed in this paper uses a scalar update at rejected trial points and has the same theoretical properties, it does not have the same computational advantage as the PSB version. This is because, although the scalar version of the DFP update, (3.1), only causes the new model to interpolate the function value at the rejected trial point, it still requires the evaluation of the gradient at the rejected trial point to compute the vector y. This problem is inherent to a DFP analog of the scalar update, because the scale-invariance is achieved by changing the subspace that the update affects from that spanned by the step direction s to that spanned by the difference in gradients y.

This DFP example appears to illustrate a fundamental difficulty in applying the scalar updating technique to the Broyden class: although the interpolation condition for the scalar update only requires the evaluation of the function at the trial point, the scale invariance property appears to require the evaluation of the gradient. There are scale-invariant formulas that do not have this difficulty, for example the formula

$$B_{k+1} = B_k + (2(f(x_k + s_k) - f(x_k) - s_k^T \nabla f(x_k)) - s_k^T B_k s_k)\frac{B_k s_k s_k^T B_k}{(s_k^T B_k s_k)^2}$$

which is related to the inverse of the inverse-Hessian form of the scalar BFGS update. But it appears very difficult to prove bounded deterioration results for formulas like this, due to difficulties in analyzing the term $B_k s_k$. Thus it is not clear to us how to obtain results like those in Section 2 for a method that uses a scale-invariant scalar update at rejected trial points. This appears to be a challenging and potentially fruitful topic for future research.

References

C. G. Broyden, J. E. Dennis, Jr. and J. J. Moré, (1973), *On the local and superlinear convergence of quasi-Newton methods,* J. Inst. Math. Appl. 12, pp. 223-246.

R. H. Byrd, H. Fayez Khalfan and R. B. Schnabel, (1993), *Analysis of a symmetric rank-one trust region method,* Technical Report CU-CS-657-93, Department of Computer Science, University of Colorado at Boulder.

A. R. Conn, N. I. M. Gould and Ph. Toint, (1988), *Testing a class of methods for solving minimization problems with simple bounds on the variables,* Math. Comp. 50, No. 182, pp. 399-430.

A. R. Conn, N. I. M. Gould and Ph. Toint, (1992), *Numerical experiments with the LANCELOT package (Release A) for large-scale nonlinear optimization,* Report 92/16, Department of Mathematics, University of Namur, Belgium.

J. E. Dennis, Jr. and R. B. Schnabel, (1983) *Numerical Methods for Nonlinear Equations and Unconstrained Optimization,* Prentice-Hall, Englewood Cliffs, New Jersy, 1983.

R. Fletcher, (1972), *An Algorithm for Solving Linearly Constrained Optimization Problems,* Mathematical Programming, 2, pp.133-165.

H. Fayez Khalfan, R. H. Byrd and R. B. Schnabel, (1993), *A Theoretical and experimental study of the symmetric rank one update,* SIAM Journal on Optimization, 3, pp. 1-24.

J. J. Moré, D. C. Sorensen, (1981), *Computing a trust region step,* Argonne National Laboratory Report, Argonne, Illinois.

M. J. D. Powell, (1975), *Convergence properties of a class of minimization algorithms,* in Nonlinear Programming 2, O. Mangasarian, R. Meyer, and S. Robinson, eds., Academic Press, New York, pp. 1-27.

G. A. Shultz, R. B. Schnabel and R.H. Byrd, (1985), *A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties,* SIAM Journal on Numerical Analysis, 22, pp. 47-67.