A GLOBALLY AND SUPERLINEARLY CONVERGENT
GAUSS–NEWTON-BASED BFGS METHOD FOR SYMMETRIC
NONLINEAR EQUATIONS

DONGHUI LI† AND MASAO FUKUSHIMA‡

Abstract. In this paper, we present a Gauss–Newton-based BFGS method for solving symmetric nonlinear equations which contain, as a special case, an unconstrained optimization problem, a saddle point problem, and an equality constrained optimization problem. A suitable line search is proposed with which the presented BFGS method exhibits an approximate norm descent property. Under appropriate conditions, global convergence and superlinear convergence of the method are established. The numerical results show that the proposed method is successful.

Key words. BFGS method, global convergence, superlinear convergence, symmetric equations

AMS subject classifications. 65H10, 90C26

VI. S0036142998335704

1. Introduction. During the past two decades, much effort has been made to establish global convergence of quasi-Newton methods, especially for convex unconstrained minimization problems (e.g., [2], [3], [11], [15], [16], [17], [19]). We refer to [7] for a recent review. For nonconvex minimization problems, the authors [12] recently proposed a modified BFGS method with global and superlinear convergence. For nonlinear equations, to the authors’ knowledge, the only global convergence result is due to Griewank [10] for Broyden’s rank one method. However, a potential problem with this method is that the line search may not be executed finitely in a certain special situation [10, pp. 81–82].

The objective of this paper is to present a well-defined BFGS method for symmetric nonlinear equations, which is based on Gauss–Newton method and converges globally and superlinearly under suitable conditions. Unlike traditional quasi-Newton methods for minimization problems, the proposed algorithm is a nonlinear equations-based method. Therefore, it may be used to solve such problems as those of finding a stationary point of an unconstrained optimization problem and a KKT point of an equality constrained optimization problem.

The paper is organized as follows. We present a Gauss–Newton-based BFGS method for symmetric equations and give some useful properties in the next section. In section 3, we establish global and superlinear convergence of the proposed method under the assumption of Jacobian uniform nonsingularity. In section 4, we report numerical results for the proposed method.

2. Algorithm and its properties. Let \( g : \mathbb{R}^n \rightarrow \mathbb{R}^n \) be differentiable. Consider the following system of nonlinear equations:

\[
g(x) = 0, \quad x \in \mathbb{R}^n.
\]
We will focus our attention on the case where the Jacobian $\nabla g(x)$ of $g$ is symmetric for all $x \in \mathbb{R}^n$. This problem comes, for example, from unconstrained optimization problems and equality constrained problems. When $g$ is the gradient mapping of some function $f : \mathbb{R}^n \to \mathbb{R}$, (1) is just the first order necessary condition for the unconstrained optimization problem

$$\min f(x), \quad x \in \mathbb{R}^n.$$  

For the equality constrained problem

$$\min f(z) \quad \text{s.t.} \quad h(z) = 0,$$  

where $h$ is a vector-valued function, the KKT conditions can be represented as the system (1) with $x = (z, v)$, and

$$g(z, v) = \left( \begin{array}{c} \nabla f(z) + \nabla h(z) v \\ h(z) \end{array} \right),$$

where $v$ is the vector of Lagrange multipliers. Notice that the Jacobian $\nabla g(z, v)$ is symmetric for all $(z, v)$. There are some other practical problems, such as the saddle point problem, the discretized two-point boundary value problem, and the discretized elliptic boundary value problem, that take the form of (1) with symmetric Jacobian (see, e.g., [14, Chapter 1]).

To begin with, we briefly review Newton’s method for solving (1). In Newton’s method, the subproblem to be solved on each iteration is the following linear equation in $p$:

$$\nabla g_k^T p + g_k = 0,$$

where $g_k = g(x_k)$, $\nabla g_k = \nabla g(x_k)$, and $x_k$ is the current iterate. Define

$$\theta(x) = \frac{1}{2} \|g(x)\|^2.$$  

Suppose $g_k \neq 0$ and let (5) have a solution $p_k$. If $\nabla g_k$ is nonsingular, then we have

$$p_k^T \nabla \theta(x_k) = -g_k^T \nabla g_k^{-1} \nabla g_k g_k = -\|g_k\|^2 < 0,$$

which implies that $p_k$ is a descent direction of $\theta$ at $x_k$. In this case, one can choose a steplength $\lambda_k > 0$ satisfying

$$\theta(x_k + \lambda_k p_k) \leq \theta(x_k) + \sigma \lambda_k g_k^T \nabla g_k p_k,$$

where $\sigma \in (0, 1)$ is a given constant. The next iterate is then determined as $x_{k+1} = x_k + \lambda_k p_k$. Newton’s method is a norm descent method in the sense that $\|g_{k+1}\| \leq \|g_k\|$ holds on every iteration. It converges globally and quadratically under suitable conditions.

Conventional quasi-Newton methods use a matrix $B_k$ instead of $\nabla g_k$ in (5). However, if the vector $p_k$ is determined from $B_k$, then $p_k$ is not necessarily a descent direction of $\theta$ at $x_k$. Therefore, the line search (7) is no longer usable. One way to globalize such quasi-Newton methods is to employ a new line search rule such as the one given by Griewank [10] for Broyden’s method. Another way is to modify a quasi-Newton direction so that it is a descent direction of $\theta$ at $x_k$. In this paper, we adopt the latter strategy.
Notice that, if $\nabla g_k$ is nonsingular, (5) is equivalent to the system
\[
\nabla g_k \nabla g_k^T p + \nabla g_k g_k = 0.
\]
This means that the Newton direction coincides with Gauss–Newton direction
\[
p_k = -(\nabla g_k \nabla g_k^T)^{-1} \nabla g_k g_k,
\]
which as mentioned above is a descent direction of $\theta$ at $x_k$. We will construct a BFGS method that generates at each iteration an approximate Gauss–Newton direction having an almost descent property. To do this, notice that if $\nabla g_k^2 = \nabla g_k \nabla g_k^T$ in (8) is replaced by a matrix $B_k$, then the solution $p_k$ of (8) will be a descent direction of $\theta$ at $x_k$ provided that $B_k$ is positive definite. However, even though $\nabla g_k \nabla g_k^T$ is replaced with $B_k$, (8) still contains $\nabla g_k$, the calculation of which should be avoided. To cope with this problem, it will be helpful to observe that if $\nabla g_k$ is symmetric, the following relationship holds:
\[
g(x_k + \lambda_k g_k) - g_k = \lambda_k \int_0^1 \nabla g(x_k + t\lambda_k g_k) g_k dt,
\]
where $\lambda_k$ is an arbitrary scalar. Denote $G_k = \int_0^1 \nabla g(x_k + t\lambda_k g_k) dt$. If $||\lambda_k g_k||$ is small, then $G_k \approx \nabla g_k$ and hence
\[
g(x_k + \lambda_k g_k) - g_k \approx \lambda_k \nabla g_k g_k.
\]
Taking this into account, we get the following linear equation as an approximation of (8):
\[
B_k p + \lambda_k^{-1} [g(x_k + \lambda_k g_k) - g_k] = 0.
\]
It is not difficult to see that if $||\lambda_k g_k||$ is sufficiently small, then the vector $p_k$ obtained by solving (9) turns out to be a descent direction of $\theta$ at $x_k$ provided that $B_k$ is positive definite. If, in addition, $B_k$ is an approximation of $\nabla g_k \nabla g_k^T$, then the solution of (9) is an approximate Gauss–Newton direction. However, it is not desirable to use a line search rule like (7) because we want to avoid using $\nabla g_k$. Therefore, a new line search technique is needed. In the following, we deal with this problem.

Let $\{\epsilon_k\}$ be a positive sequence satisfying
\[
\sum_{k=0}^{\infty} \epsilon_k < \infty.
\]
We determine a positive steplength $\lambda_k$ so that the following inequality holds for $\lambda = \lambda_k$:
\[
||g(x_k + \lambda p_k)||^2 - ||g_k||^2 \leq -\sigma_1 ||\lambda g_k||^2 - \sigma_2 ||\lambda p_k||^2 + \epsilon_k ||g_k||^2,
\]
where $\sigma_1$ and $\sigma_2$ are some positive constants. It is straightforward to see that as $\lambda \to 0^+$, the left-hand side of (11) goes to zero, while the right-hand side tends to the positive constant $\epsilon_k ||g_k||^2$. Thus, (11) is satisfied for all sufficiently small $\lambda > 0$. Hence, we can obtain $\lambda_k$ by means of a backtracking process.
Now we state a Gauss–Newton-based BFGS method for solving (1).

**Algorithm 1.**

**Step 0.** Choose an initial point \( x_0 \in \mathbb{R}^n \), an initial symmetric positive definite matrix \( B_0 \in \mathbb{R}^{n \times n} \), a positive sequence \( \{\epsilon_k\} \) satisfying (10), and constants \( r, \rho \in (0,1), \sigma_1, \sigma_2 > 0, \lambda_{-1} > 0 \). Let \( k := 0 \).

**Step 1.** Stop if \( g_k = 0 \). Otherwise, solve the following linear equation to get \( p_k \):

\[
(12) \quad B_k p + \lambda_{k-1}^{-1}(g(x_k + \lambda_{k-1} g_k) - g_k) = 0.
\]

**Step 2.** If

\[
(13) \quad \|g(x_k + p_k)\| \leq \rho \|g_k\|,
\]

then take \( \lambda_k = 1 \) and go to Step 4. Otherwise go to Step 3.

**Step 3.** Let \( i_k \) be the smallest nonnegative integer \( i \) such that (11) holds for \( \lambda = r^i \). Let \( \lambda_k = r^{i_k} \).

**Step 4.** Let the next iterate be \( x_{k+1} = x_k + \lambda_k p_k \).

**Step 5.** Put \( s_k = x_{k+1} - x_k = \lambda_k p_k \), \( \delta_k = g_{k+1} - g_k \), and \( y_k = g(x_k + \delta_k) - g(x_k) \). If \( y_k^T s_k \leq 0 \), then \( B_{k+1} = B_k \) and go to Step 6. Otherwise, update \( B_k \) by the BFGS formula:

\[
(14) \quad B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}.
\]

**Step 6.** Let \( k := k + 1 \). Go to Step 1.

*Remark.* (1) In the update formula (14), we use \( y_k = g(x_k + \delta_k) - g(x_k) \). This differs from the standard update formula where \( y_k \) is the difference of the gradients \( g_{k+1} - g_k \), which is denoted by \( \delta_k \) in Algorithm 1. By doing so, we have the approximate relations

\[
y_k \approx \nabla g_{k+1} \delta_k \approx \nabla g_{k+1} \nabla g_{k+1} s_k.
\]

Since \( B_{k+1} \) satisfies the secant equation \( B_{k+1} s_k = y_k \) and \( \nabla g_k \) is symmetric, we have approximately

\[
B_{k+1} s_k \approx \nabla g_{k+1} \nabla g_{k+1} s_k = \nabla g_{k+1} \nabla g_{k+1}^T s_k.
\]

This means that \( B_{k+1} \) approximates \( \nabla g_{k+1} \nabla g_{k+1}^T \) along direction \( s_k \). We also have approximately

\[
\lambda_{k-1}^{-1}(g(x_k + \lambda_{k-1} g_k) - g_k) \approx \nabla g_{k} g_k.
\]

Therefore, the subproblem (12) can be regarded as an approximation to the subproblem (8) of the Gauss–Newton method. In this sense, we call Algorithm 1 a Gauss–Newton-based BFGS method.

(2) Since Step 5 ensures that \( B_k \) is always symmetric and positive definite, (12) has a unique solution for each \( k \). Moreover, for every \( k \), Step 3 can be executed in finite steps. Therefore, the method is well defined.

(3) If (13) does not hold, then the steplength \( \lambda_k \) satisfies

\[
(15) \quad \|g_{k+1}\|^2 - \|g_k\|^2 \leq -\sigma_1 \|\lambda_k g_k\|^2 - \sigma_2 \|\lambda_k p_k\|^2 + \epsilon_k \|g_k\|^2.
\]
(4) Since \( \{ \epsilon_k \} \) satisfies (10), the inequalities (13) and (15) indicate that \( \{ g_k \} \) is at least approximately norm descent. Moreover, as we will see in section 3, (13) holds for all \( k \) sufficiently large. In other words, \( \{ g_k \} \) is norm descent when \( k \) is sufficiently large.

Let \( \Omega \) be the level set defined by
\[
\Omega = \{ x \mid \| g(x) \| \leq \epsilon \| g(x_0) \| \},
\]
where \( \epsilon \) is a positive constant such that
\[
\sum_{k=0}^{\infty} \epsilon_k \leq \epsilon. \tag{17}
\]

Then we have the following lemma.

**Lemma 2.1.** Let \( \{ x_k \} \) be generated by Algorithm 1. Then \( \{ x_k \} \subset \Omega \). Moreover, \( \{ \| g_k \| \} \) converges.

**Proof.** For every \( k \), we have either (13) or (15). In either case, we have
\[
\| g_{k+1} \| \leq (1 + \epsilon_k)^{\frac{k+1}{2}} \| g_k \| \leq (1 + \epsilon_k)^{\frac{k+1}{2}} \| g_k \|.
\]
Since \( \epsilon_k \) satisfies (10), we conclude from Lemma 3.3 in [4] that \( \{ \| g_k \| \} \) converges. Moreover, we have for all \( k \)
\[
\| g_{k+1} \| \leq (1 + \epsilon_k)^{\frac{k+1}{2}} \| g_k \| \\
\vdots \\
\leq \prod_{i=0}^{k} (1 + \epsilon_i)^{\frac{1}{2}} \| g(x_0) \| \\
\leq \| g(x_0) \| \left[ \frac{1}{k+1} \sum_{i=0}^{k} (1 + \epsilon_i) \right]^{\frac{k+1}{2}} \\
= \| g(x_0) \| \left[ 1 + \frac{1}{k+1} \sum_{i=0}^{k} \epsilon_i \right]^{\frac{k+1}{2}} \\
\leq \| g(x_0) \| \left( 1 + \epsilon \frac{k+1}{k+1} \right)^{\frac{k+1}{2}} \\
\leq \epsilon^{\frac{1}{2}} \| g(x_0) \|,
\]
where \( \epsilon \) is a constant satisfying (17). This implies that \( \{ x_k \} \subset \Omega \). \( \square \)

We make the following assumptions which will be frequently used in the rest of this section and in the next section.

**Assumption A.** (i) \( g \) is continuously differentiable on an open convex set \( \Omega_1 \) containing \( \Omega \).

(ii) The Jacobian of \( g \) is symmetric and bounded on \( \Omega_1 \), i.e., \( \nabla g(x)^T = \nabla g(x) \) for every \( x \in \Omega_1 \) and there exists a positive constant \( M \) such that
\[
\| \nabla g(x) \| \leq M \quad \forall x \in \Omega_1.
\]

(iii) \( \nabla g \) is uniformly nonsingular on \( \Omega_1 \); i.e., there is a constant \( m > 0 \) such that
\[
m\| p \| \leq \| \nabla g(x)p \| \quad \forall x \in \Omega_1, p \in \mathbb{R}^n.
\]
Remark. Conditions (ii) and (iii) in Assumption A imply that there exist constants \( M \geq m > 0 \) such that

\[
m\|p\| \leq \|\nabla g(x)p\| \leq M\|p\| \quad \forall x \in \Omega_1, p \in \mathbb{R}^n,
\]

\[
\frac{1}{M}\|p\| \leq \|\nabla g(x)^{-1}p\| \leq \frac{1}{m}\|p\| \quad \forall x \in \Omega_1, p \in \mathbb{R}^n,
\]

and

\[
m\|x - y\| \leq \|g(x) - g(y)\| \leq M\|x - y\| \quad \forall x, y \in \Omega_1.
\]

In particular, for all \( x \in \Omega_1 \), we have

\[
m\|x - x^\ast\| \leq \|g(x)\| = \|g(x) - g(x^\ast)\| \leq M\|x - x^\ast\|,
\]

where \( x^\ast \) stands for the unique solution of (1) in \( \Omega_1 \). Moreover, the level set \( \Omega \) is bounded.

Under Assumption A, we can prove some useful properties pertaining to Algorithm 1. The next lemma follows immediately from Assumption A, so that the proof is omitted.

**Lemma 2.2.** Let conditions (i) and (ii) in Assumption A be satisfied. Then the following inequalities hold for every \( k \):

\[
\|s_k\| \leq M\|y_k\| \quad \text{and} \quad \|y_k\| \leq M\|\delta_k\| \leq M^2\|s_k\|.
\]

Denote

\[
G_k = \int_0^1 \nabla g(x_k + \tau\lambda_{k-1}g_k)d\tau.
\]

Then we have \( \lambda_{k-1}^{-1}[g(x_k + \lambda_{k-1}g_k) - g_k] = G_kg_k \). Hence (12) can be rewritten as

\[
B_kp + G_kg_k = 0.
\]

**Lemma 2.3.** Let Assumption A be satisfied. Then the following statements hold.

(i) If \( s_k \to 0 \), then there is a constant \( m_1 > 0 \) such that for all \( k \) sufficiently large

\[
y_k^Ts_k \geq m_1\|s_k\|^2.
\]

(ii) Suppose that (13) holds only for a finite number of \( k \)’s. Then we have

\[
\sum_{k=0}^{\infty} \|\lambda_kg_k\|^2 < \infty
\]

and

\[
\sum_{k=0}^{\infty} \|\lambda_kp_k\|^2 = \sum_{k=0}^{\infty} \|s_k\|^2 < \infty.
\]

Moreover, (25) holds for all \( k \) sufficiently large.
Proof. (i) By the mean-value theorem we have
\[
g_k^T s_k = s_k^T [g(x_k + \delta_k) - g_k]
= s_k^T \int_0^1 \nabla g(x_k + \tau \delta_k) d\tau \delta_k
= s_k^T \int_0^1 \nabla g(x_k + \tau \delta_k) d\tau \int_0^1 \nabla g(x_k + \tau s_k) d\tau s_k
= s_k^T \left[ \int_0^1 \nabla g(x_k + \tau s_k) d\tau \right]^2 s_k
+ s_k^T \int_0^1 [\nabla g(x_k + \tau \delta_k) - \nabla g(x_k + \tau s_k)] d\tau \int_0^1 \nabla g(x_k + \tau s_k) d\tau s_k
\geq \left\| \int_0^1 \nabla g(x_k + \tau s_k) d\tau s_k \right\|^2
- \left\| \int_0^1 [\nabla g(x_k + \tau \delta_k) - \nabla g(x_k + \tau s_k)] d\tau \int_0^1 \nabla g(x_k + \tau s_k) d\tau s_k \right\|
= \|g_{k+1} - g_k\|^2
- \left\| \int_0^1 [\nabla g(x_k + \tau \delta_k) - \nabla g(x_k + \tau s_k)] d\tau \int_0^1 \nabla g(x_k + \tau s_k) d\tau s_k \right\|
\geq m^2 \|s_k\|^2 - M \|s_k\|^2 \int_0^1 \|\nabla g(x_k + \tau \delta_k) - \nabla g(x_k + \tau s_k)\| d\tau
= \left[ m^2 - M \int_0^1 \|\nabla g(x_k + \tau \delta_k) - \nabla g(x_k + \tau s_k)\| d\tau \right] \|s_k\|^2,
\]
where the last inequality follows from (18) and (20). If \(s_k \to 0\), then \(\delta_k = g_{k+1} - g_k \to 0\). By the continuity of \(\nabla g\), we get (25).

(ii) If (13) holds for only finitely many \(k\)'s, then Step 3 is used to determine a steplength \(\lambda_k\) for all \(k\) sufficiently large. By (15) we have \(\sigma_1 \|\lambda_k g_k\|^2 + \sigma_2 \|s_k\|^2 \leq \|g_k\|^2 - \|g_{k+1}\|^2 + \epsilon_k \|g_k\|^2\). Since \{\|g_k\|\} is bounded and \{\{\epsilon_k\} satisfies (10), we get (26) and (27) by summing these inequalities. In particular, \(\|s_k\| \to 0\), which also implies that (25) holds for all \(k\) sufficiently large. 

3. Global and superlinear convergence. In this section, we establish global and superlinear convergence for Algorithm 1. First, we prove some useful lemmas.

Lemma 3.1. Let Assumption A hold. Then there are a positive integer \(k'\) and positive constants \(\beta_j\), \(j = 1, 2, 3\), such that, for any \(k \geq k'\), the inequalities
\[
\beta_2 \|s_i\|^2 \leq s_i^T B_i s_i \leq \beta_3 \|s_i\|^2 \quad \text{and} \quad \|B_i s_i\| \leq \beta_1 \|s_i\|
\]
hold for at least half of indices \(i \in \{0, 1, 2, \ldots, k\}\).

Proof. By Lemma 2.2 and 2.3, (22) and (25) hold for all \(k\) sufficiently large, say \(k \geq k'\). From Theorem 2.1 in [2], conditions (25) and (22) imply that (28) holds for at least \([(k - k')/2]\) of indices \(i\) such that \(k' \leq i \leq k\). Since \(k'\) is a fixed integer and \(B_i\) are positive definite, we may take smaller \(\beta_2\), and larger \(\beta_1\) and \(\beta_3\) if necessary so that (28) holds for all \(i < k'\). Therefore (28) holds for at least half of indices \(i \in \{0, 1, 2, \ldots, k\}\).

The following lemma is important in the analysis of global convergence of Algorithm 1.
LEMMA 3.2. Let conditions (i) and (ii) in Assumption A hold. If \( \lambda_k \neq 1 \), then we have the following estimate for \( \lambda_k \):

\[
\lambda_k \geq \frac{2(p_k^T B_k p_k - t_k \|p_k\| \|g_k\|)r}{\sigma_1 \|g_k\|^2 + (\sigma_2 + M^2) \|p_k\|^2},
\]

where \( t_k = \int_0^1 \|\nabla g(x_k + \tau \lambda_k^{-1} p_k) - \nabla g(x_k + \tau \lambda_k^{-1} g_k)\| d\tau \).

Proof. The fact that \( \lambda_k \neq 1 \) indicates that \( \lambda_k \) was determined in Step 3 and that \( \lambda_k' = \lambda_k/r \) did not satisfy (11), i.e.,

\[
\|g(x_k + \lambda_k' p_k)\|^2 - \|g_k\|^2 > -\sigma_1 \|\lambda_k' g_k\|^2 - \sigma_2 \|\lambda_k' p_k\|^2 + \epsilon_k \|g_k\|^2
\]

or equivalently

\[
(\lambda_k')^2 (\sigma_1 \|g_k\|^2 + \sigma_2 \|p_k\|^2) \geq -\|g(x_k + \lambda_k' p_k)\|^2 - \|g_k\|^2.
\]

By an elementary deduction we get

\[
\|g(x_k + \lambda_k' p_k)\|^2 - \|g_k\|^2 = (g(x_k + \lambda_k' p_k) + g_k)^T (g(x_k + \lambda_k' p_k) - g_k)
\]

\[
= 2g_k^T (g(x_k + \lambda_k' p_k) - g_k) + \|g(x_k + \lambda_k' p_k) - g_k\|^2
\]

\[
\leq 2g_k^T (g(x_k + \lambda_k' p_k) - g_k) + M^2 \|\lambda_k' p_k\|^2. \tag{31}
\]

For the first term on the right-hand side of (31), we have

\[
g_k^T (g(x_k + \lambda_k' p_k) - g_k)
\]

\[
= \lambda_k' g_k^T \int_0^1 \nabla g(x_k + \tau \lambda_k' p_k)p_k d\tau
\]

\[
= \lambda_k' g_k^T \int_0^1 \nabla g(x_k + \tau \lambda_k^{-1} p_k)p_k d\tau
\]

\[
+ \lambda_k' g_k^T \int_0^1 [\nabla g(x_k + \tau \lambda_k' p_k) - \nabla g(x_k + \tau \lambda_k^{-1} g_k)] d\tau p_k
\]

\[
= -\lambda_k' p_k^T B_k p_k + \lambda_k' g_k^T \int_0^1 [\nabla g(x_k + \tau \lambda_k' p_k) - \nabla g(x_k + \tau \lambda_k^{-1} g_k)] d\tau p_k
\]

\[
\leq -\lambda_k' p_k^T B_k p_k + \lambda_k' \|g_k\| \|p_k\| \int_0^1 \|\nabla g(x_k + \tau \lambda_k' p_k) - \nabla g(x_k + \tau \lambda_k^{-1} g_k)\| d\tau
\]

\[
= -\lambda_k' p_k^T B_k p_k + \lambda_k' t_k \|g_k\| \|p_k\|, \tag{32}
\]

where the third equality follows from (23) and (24). Applying (32) to (31), we get from (30)

\[
(\lambda_k')^2 (\sigma_1 \|g_k\|^2 + \sigma_2 \|p_k\|^2) \geq 2\lambda_k' p_k^T B_k p_k - 2\lambda_k' t_k \|p_k\| \|g_k\| - M^2 \|\lambda_k' p_k\|^2,
\]

that is,

\[
\lambda_k' \geq \frac{2(p_k^T B_k p_k - t_k \|p_k\| \|g_k\|)}{\sigma_1 \|g_k\|^2 + (\sigma_2 + M^2) \|p_k\|^2},
\]

Since \( \lambda_k = \lambda_k' r \), we obtain (29). \( \square \)
Remark. It is not difficult to see from the above proof that (29) holds if $B_k$ is updated by any other update formula. In other words, the lemma is independent of the update formula (14). We will use this result again in section 4.

Now we establish a global convergence theorem for Algorithm 1. Let the constants $\beta_j, j = 1, 2, 3$, be as specified in Lemma 3.1 and define the index set

$$J = \{i \mid (28) \text{ holds}\}.$$  

THEOREM 3.3. Let Assumption A hold. Then the sequence $\{x_k\}$ generated by Algorithm 1 converges to the unique solution $x^*$ of (1).

Proof. By Lemma 2.1, $\{\|g_k\|\}$ converges. So, if

$$\liminf_{k \to \infty} \|g_k\| = 0,$$

then every accumulation point of $\{x_k\}$ is a solution of (1). Since $\nabla g(x)$ is uniformly nonsingular on $\Omega$, (1) has only one solution. Moreover, since $\Omega$ is bounded, $\{x_k\} \subset \Omega$ has at least one accumulation point. Therefore $\{x_k\}$ itself converges to the unique solution of (1). It thus suffices to verify (34).

If (13) holds for infinitely many $k$’s, then (34) is trivial. Consider the case where (13) holds for only finitely many $k$’s, so that Step 3 is executed for all $k$ sufficiently large. Since (26) holds, we need only to show that there is an infinite subsequence of $\{\lambda_k\}$ with a positive lower bound, i.e., $\limsup_{k \to \infty} \lambda_k > 0$.

We assume $\lim_{k \to \infty} \lambda_k = 0$ for contradiction. Notice that $s_k = \lambda_k p_k$ and $B_k s_k = \lambda_k B_k p_k = -\lambda_k G_k g_k$ by (24). From (28) and (33), it is clear that when $k \in J$ is sufficiently large, $p_k^T B_k p_k \geq \beta_2 \|p_k\|^2$ and $\|g_k\| = \|G_k^{-1} B_k p_k\| \leq \frac{1}{m} \|B_k p_k\| \leq \frac{1}{m} \beta_1 \|p_k\|$, where the last inequality follows from (19). Since $s_k \to 0$ and $\lambda_k \to 0$, we have $\lambda_k p_k = r^{-1} s_k \to 0$ and $\lambda_{k-1} g_k \to 0$. Therefore $t_k$ defined in Lemma 3.2 goes to zero.

Consequently, we can deduce from (29) that $\{\lambda_k\}_{k \in J}$ is bounded away from zero. This contradicts the assumption $\lim_{k \to \infty} \lambda_k = 0$. The proof is complete. \[\square\]

As a corollary of Theorem 3.3, we have the following result.

COROLLARY 3.4. Let Assumption A hold. Then there is a positive constant $\tilde{\lambda}$ such that for every $k \in J$, if (13) does not hold, then $\lambda_k \geq \tilde{\lambda}$.

Proof. It follows from (24) and (28) that for any $k \in J$,

$$\|g_k\| = \|G_k^{-1} B_k p_k\| \leq \frac{1}{m} \|B_k p_k\| \leq \frac{\beta_1}{m} \|p_k\|,$$

where $G_k$ is defined by (23) and the first inequality follows from (19). By Theorem 3.3 we have that $t_k$ defined by Lemma 3.2 tends to zero as $k \to \infty$. Therefore, there is a positive integer $k$ such that $t_k \leq \frac{1}{2} \beta_1^{-1} m \beta_2$, where $\beta_2$ is the constant in (28). Thus, from (28), (29), and (35), we have for any $k \in J$ sufficiently large

$$\lambda_k \geq \frac{2(\beta^2 \sigma_1 p_k^T B_k p_k - t_k \|p_k\| \|g_k\|)}{\sigma_1 \|g_k\|^2 + (\sigma_2 + M^2) \|p_k\|^2} \geq \frac{2(\beta_2 \|p_k\|^2 - m^{-1} \beta_1 t_k \|p_k\|^2) r}{\sigma_1 \|g_k\|^2 + (\sigma_2 + M^2) \|p_k\|^2} \geq \frac{\beta_2 \|p_k\|^2 r}{\sigma_1 \|g_k\|^2 + (\sigma_2 + M^2) \|p_k\|^2} \geq \frac{\beta_2 \|p_k\|^2 r}{m^{-2} \beta_1^2 \sigma_1 \|p_k\|^2 + (\sigma_2 + M^2) \|p_k\|^2} \geq \frac{\beta_2 r}{m^{-2} \beta_1^2 \sigma_1 + \sigma_2 + M^2}.$$
Letting \( \hat{\lambda} = \beta_{2\tau}/(m^-2\beta_{1}\sigma_1 + \sigma_2 + M^2) \), we get the conclusion. \( \square \)

Now we turn to proving the superlinear convergence of Algorithm 1. In the rest of the paper, we abbreviate \( g(x^*) \) and \( \nabla g(x^*) \) as \( g_* \) and \( \nabla g_* \), respectively. The following lemma shows that, like the ordinary BFGS method, the Dennis–Moré condition \([4]\), \([5]\) ensures superlinear convergence of Algorithm 1. Recall again that \( B_k \) is updated so as to approximate \( \nabla g^2_k = \nabla g_k \nabla g_k^T \) in Algorithm 1.

**Lemma 3.5.** Let Assumption A hold. If

\[
\lim_{k \to \infty} \frac{||(B_k - \nabla g^2_k)p_k||}{||p_k||} = 0,
\]

then \( \lambda_k \equiv 1 \) for all \( k \) sufficiently large. Moreover, \( \{x_k\} \) converges superlinearly.

**Proof.** Denote

\[
\eta_k = \frac{||(B_k - \nabla g^2_k)p_k||}{||p_k||}.
\]

Then by (24),

\[
||\nabla g^2_k p_k|| = ||(B_k - \nabla g^2_k)p_k + G_k g_k|| \\
\leq ||(B_k - \nabla g^2_k)p_k|| + ||G_k g_k|| \\
\leq \eta_k ||p_k|| + M ||g_k||,
\]

(38)

where \( G_k \) is defined by (23). Since \( \eta_k \to 0 \) and \( \nabla g_* \) is nonsingular, the inequality (38) implies that there is a constant \( M_2 > 0 \) such that for all \( k \) sufficiently large

\[
||p_k|| \leq M_2 ||g_k||.
\]

Again by (24), we have

\[
\nabla g^2_k(x_k + p_k - x^*) = \nabla g^2_k(x_k - x^*) + \nabla g^2_k p_k \\
= \nabla g^2_k(x_k - x^*) - G_k g_k + (\nabla g^2_k - B_k)p_k \\
= \nabla g^2_k(x_k - x^*) - G_k (g_k - g_*) + (\nabla g^2_k - B_k)p_k \\
= (\nabla g^2_k - G_k \overline{G}_k)(x_k - x^*) + (\nabla g^2_k - B_k)p_k,
\]

where \( \overline{G}_k = \int_0^1 \nabla g(x^* + \tau(x_k - x^*))d\tau \). It then follows that

\[
||\nabla g^2_k(x_k + p_k - x^*)|| \leq ||(\nabla g^2_k - G_k \overline{G}_k)(x_k - x^*)|| + ||(\nabla g^2_k - B_k)p_k|| \\
= ||(\nabla g^2_k - G_k \overline{G}_k)(x_k - x^*)|| + \eta_k ||p_k|| \\
\leq ||\nabla g^2_k - G_k \overline{G}_k|| ||x_k - x^*|| + \eta_k M_2 ||g_k|| \\
\leq ||\nabla g^2_k - G_k \overline{G}_k|| ||x_k - x^*|| + \eta_k M_2 M ||x_k - x^*|| \\
= \eta_k M_2 M ||x_k - x^*|| + o(||x_k - x^*||),
\]

(40)

where the second inequality follows from (39) and the last inequality follows again from (21). Since \( \eta_k \to 0 \) and \( \nabla g^2_k \) is nonsingular, the inequality (40) implies

\[
\frac{||x_k + p_k - x^*||}{||x_k - x^*||} \to 0.
\]

(41)
Moreover, we have
\[
\|g(x_k + p_k)\| = \|g(x_k + p_k) - g_*\| \\
\leq M\|x_k + p_k - x^*\| \\
= \frac{M}{m}\|x_k + p_k - x^*\| m\|x_k - x^*\| \\
\leq \frac{M}{m}\|x_k + p_k - x^*\|\|g_k\|,
\]
where the last inequality follows from (21). This together with (41) indicates that (13) is satisfied for all \(k\) sufficiently large. In other words, the unit steplength is always accepted for all \(k\) sufficiently large. Moreover, (41) implies the superlinear convergence of \(\{x_k\}\).

Lemma 3.5 shows that to establish superlinear convergence of Algorithm 1, it suffices to verify that \(\{x_k\}\) satisfies the Dennis–Moré condition (36).

**Lemma 3.6.** Let Assumption A hold. Then, for any fixed \(\gamma > 0\), we have
\[
\sum_{k=0}^{\infty} \|x_k - x^*\|^{\gamma} < \infty.
\]
Moreover, we have
\[
\sum_{k=0}^{\infty} \nu_k(\gamma) < \infty,
\]
where \(\nu_k(\gamma) = \max\{\|x_k - x^*\|^{\gamma}, \|x_k+1 - x^*\|^{\gamma}\}\).

**Proof.** We first show that there exists an index \(i'\) and a constant \(\bar{\rho} \in (0,1)\) such that
\[
\|g_{i+1}\|^2 \leq \bar{\rho}\|g_i\|^2 \quad \forall i \in J \text{ with } i \geq i'.
\]

If the steplength \(\lambda_i\) is determined by Step 2, then
\[
\|g_{i+1}\|^2 \leq \rho^2\|g_i\|^2.
\]
On the other hand, if \(\lambda_i\) is determined by Step 3, then it satisfies (11) with \(k = i\) and hence
\[
\|g_{i+1}\|^2 \leq (1 - \sigma_1\lambda_i^2 + \epsilon_i)\|g_i\|^2 - \sigma_2\|s_i\|^2 \leq (1 - \sigma_1\lambda_i^2 + \epsilon_i)\|g_i\|^2.
\]
In the latter case, by Corollary 3.4, there is some constant \(\tilde{\lambda} > 0\) such that \(\lambda_i \geq \tilde{\lambda}\) for all \(i \in J\). It then follows from (46) that
\[
\|g(x_{i+1})\|^2 \leq (1 + \epsilon_i - \sigma_1\tilde{\lambda}^2)\|g(x_i)\|^2.
\]
Since \(\epsilon_i \to 0\) as \(i \to \infty\), there are an index \(i'\) and a constant \(\rho_1 \in (0,1)\) such that \(1 + \epsilon_i - \sigma_1\lambda_i^2 \leq \rho_1\) holds for all \(i \geq i'\). Let \(\bar{\rho} = \min\{\rho^2, \rho_1\} < 1\). Then (44) follows from (45) and (47).

Now we verify (42). Notice that by Lemma 3.1, for all \(k \geq k'\), the number of elements in \(J\) is at least \(\lceil \frac{k}{2} \rceil\). Therefore, for any \(k > \max\{k', 2i'\}\), there are at least \(\lceil \frac{k}{2} \rceil - i'\) of indices \(i\) such that (44) holds. Let \(K\) denote the set of indices \(i\) for which
(44) holds. Also, let \( r_k \) denote the number of indices in \( K \) not exceeding \( k \). Then we have \( r_k \geq \left[ \frac{k}{2} \right] - i' \geq 0 \) for each \( k \). Multiplying (44) for \( i \in K \) and (46) for \( i \notin K \) from \( i = i' \) to \( i = k \) yields

\[
\|g_{k+1}\|^2 \leq \left[ \prod_{i=i'}^k (1 + \epsilon_i - \sigma_1 \lambda_i^2) \right] \tilde{\rho}^{r_k} \|g(x_{i'})\|^2
\]

\[
\leq \left[ \prod_{i=i'}^k (1 + \epsilon_i) \right] \tilde{\rho}^{r_k} \|g(x_{i'})\|^2
\]

\[
\leq \left[ \prod_{i=0}^k (1 + \epsilon_i) \right] \tilde{\rho}^{r_k} \|g(x_{i'})\|^2
\]

\[
\leq e^{\tilde{\rho} \frac{k}{2} - (i' + 1)} \|g(x_{i'})\|^2
\]

\[
= e^{\tilde{\rho} \frac{k}{2}} \|g(x_{i'})\|^2
\]

where \( c' = e^{\tilde{\rho} \frac{k}{2} - (i' + 1)} \|g(x_{i'})\|^2 \) and \( \tilde{\rho} = \rho^{\frac{k}{2}} \in (0, 1) \). This together with (21) shows that \( \|x_{k+1} - x^*\|^2 \leq m^{-2} c' \tilde{\rho}^k \) holds for all \( k \) large enough. Hence we have (42) for any \( \gamma \).

Since \( \nu_k(\gamma) \leq \|x_{k+1} - x^*\|^\gamma + \|x_k - x^*\|^\gamma \), (43) follows from (42).

To obtain superlinear convergence of Algorithm 1, we need a further assumption on \( g \).

Assumption B. \( \nabla g \) is Hölder continuous at \( x^* \); i.e., there are positive constants \( M_3 \) and \( \alpha \) such that for every \( x \) in a neighborhood of \( x^* \)

\[
\|\nabla g(x) - \nabla g(x^*)\| \leq M_3 \|x - x^*\|^\alpha.
\]

Lemma 3.7. Let Assumptions A and B hold. Then there exist positive constants \( M_4 \) and \( m_2 \) such that for all \( k \) sufficiently large

\[
\|y_k - \nabla g^2 s_k\| \leq M_4 \nu_k \|s_k\| \quad \text{and} \quad \|y_k\| \geq m_2 \|s_k\|,
\]

where \( \nu_k = \max\{\|x_k - x^*\|^\alpha, \|x_{k+1} - x^*\|^\alpha\} \).

Proof. Since \( x_k \to x^* \), (48) holds for all \( k \) large enough. By the mean value theorem we have for all \( k \) sufficiently large

\[
\|y_k - \nabla g^2 s_k\| = \|g(x_k + \delta_k) - g_k - \nabla g^2 s_k\|
\]

\[
= \left\| \int_0^1 \nabla g(x_k + \tau \delta_k) d\tau \delta_k - \nabla g^2 s_k \right\|
\]

\[
= \left\| \left[ \int_0^1 \nabla g(x_k + \tau \delta_k) d\tau \right] \left[ \int_0^1 \nabla g(x_k + t \delta_k) dt - \nabla g^2 s_k \right] \right\|
\]

\[
\leq \left\| \int_0^1 (\nabla g(x_k + \tau \delta_k) - \nabla g_s) d\tau \int_0^1 \nabla g(x_k + t \delta_k) dt \right\| \|s_k\|
\]

\[
+ \left\| \nabla g_s \int_0^1 (\nabla g(x_k + t \delta_k) - \nabla g_s) dt \right\| \|s_k\|
\]
\[
\begin{align*}
M\|s_k\| \left[ \int_0^1 & \|\nabla g(x_k + \tau \delta_k) - \nabla g_x\| d\tau \\
& + \int_0^1 \|\nabla g(x_k + ts_k) - \nabla g_x\| dt \right]
\leq MM_3\|s_k\| \left[ \int_0^1 \|x_k - x^* + \tau \delta_k\|^\alpha d\tau + \int_0^1 \|x_k - x^* + ts_k\|^\alpha dt \right]
\leq MM_3\|s_k\| \left[ \int_0^1 (\|x_k - x^*\| + \|\delta_k\|)^\alpha d\tau \\
& + \int_0^1 ((1-t)x_k + tx_{k+1} - x^*)^\alpha dt \right]
\leq MM_3\|s_k\| \left\{ (\|x_k - x^*\| + \|\delta_k\|)^\alpha \\
& + \int_0^1 ((1-t)||x_k - x^*|| + t||x_{k+1} - x^*||)^\alpha dt \right\}
\leq MM_3\|s_k\| \{ (\|x_k - x^*\| + M\|s_k\|)^\alpha + \nu_k \}
\leq MM_3\|s_k\| \{ (\|x_k - x^*\| + M(||x_k - x^*|| + ||x_{k+1} - x^*||))^\alpha + \nu_k \}
\leq M_4 \nu_k \|s_k\|,
\end{align*}
\]

where \(M_4 = MM_3[(2M + 1)^\alpha + 1]\). This yields the first inequality of (49). Moreover, by (50) we have

\[
\|y_k\| \geq \|\nabla g^2 s_k\| - \|y_k - \nabla g^2 s_k\| \geq \|\nabla g^2 s_k\| - M_4 \nu_k \|s_k\|.
\]

Since \(\nabla g_x\) is nonsingular and \(\nu_k \to 0\), the second inequality of (49) follows from the last inequality. \(\square\)

Denote \(P = \nabla g^{-1}\). For an \(n \times n\) matrix \(A\), define a matrix norm \(\|A\|_F = \|PA\|_F\), where \(\| \cdot \|_F\) denotes the Frobenius norm of a matrix. We let \(H_k\) and \(H_{k+1}\) stand for the inverse matrices of \(B_k\) and \(B_{k+1}\), respectively.

We notice that Theorem 3.3 ensures that \(\{x_k\}\) converges. In particular, \(s_k \to 0\). Therefore, Lemma 2.3(i) yields that \(y_k^T s_k > 0\) for all \(k\) sufficiently large. Hence we see from Step 5 in Algorithm 1 that for all \(k\) large enough, \(B_{k+1}\) is always generated by the update formula (14).

The following lemma shows that the BFGS formula (14) exhibits a property similar to that of the ordinary BFGS formula.

**Lemma 3.8.** Under Assumptions A and B, there exist positive constants \(M_5, M_6, M_7, \) and \(\omega \in (0, 1)\) such that for all \(k\) sufficiently large

\[
|B_{k+1} - \nabla g_x^2|_P \leq |B_k - \nabla g_x^2|_P + M_5 \nu_k,
\]

\[
|H_{k+1} - \nabla g_x^{-2}|_{P^{-1}} \leq (1 - \frac{1}{2} \omega \mu_k^2 + M_6 \nu_k) |H_k - \nabla g_x^{-2}|_{P^{-1}} + M_7 \nu_k,
\]

where \(\nu_k = \max\{||x_k - x^*||^\alpha, ||x_{k+1} - x^*||^\alpha\}\) and \(\mu_k\) is given by

\[
\mu_k = \frac{||P^{-1}[H_k - \nabla g_x^{-2}]y_k||}{||H_k - \nabla g_x^{-2}|_{P^{-1}}\|P y_k||}.
\]
In particular, \(\{\|B_k\|\}\) and \(\{\|H_k\|\}\) are bounded.

Proof. From the update formula (14), we get

\[
P(B_{k+1} - \nabla g_k^2)P = P(B_k - \nabla g_k^2)P - \frac{[(PB_kP)(P^{-1}s_k)][(PB_kP)(P^{-1}s_k)]^T}{(P^{-1}s_k)^T(PB_kP)(P^{-1}s_k)} + \frac{(Py_k)(Py_k)^T}{(Py_k)^T(P^{-1}s_k)}.
\]

(54)

Denote \(\tilde{B}_k = PB_kP\), \(\tilde{s}_k = P^{-1}s_k\), \(\tilde{y}_k = Py_k, Q_{k+1} = P(B_{k+1} - \nabla g_k^2)P = PB_{k+1}P - I\), and \(Q_k = P(B_k - \nabla g_k^2)P = \tilde{B}_k - I\). Then \(Q_k, Q_{k+1}\), and \(\tilde{B}_k\) are symmetric, and (54) is rewritten as

\[
Q_{k+1} = Q_k - \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{y}_k \tilde{y}_k^T}{\tilde{y}_k^T \tilde{s}_k}.
\]

Taking the norm operation on both sides, we get

\[
\|Q_{k+1}\|_F \leq \left\|Q_k - \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{s}_k \tilde{s}_k^T}{\|\tilde{s}_k\|^2} + \frac{\tilde{s}_k \tilde{s}_k^T}{\|\tilde{s}_k\|^2} \right\|_F.
\]

(55)

We estimate the two terms on the right-hand side of (55). For the first term, we have

\[
\left\|Q_k - \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{s}_k \tilde{s}_k^T}{\|\tilde{s}_k\|^2} \right\|_F^2 = \text{trace} \left\{ \left( Q_k - \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{s}_k \tilde{s}_k^T}{\|\tilde{s}_k\|^2} \right)^T \left( Q_k - \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{s}_k \tilde{s}_k^T}{\|\tilde{s}_k\|^2} \right) \right\}
\]

\[
= \text{trace} \left\{ Q_k^2 + \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k + \tilde{s}_k \tilde{s}_k^T \|\tilde{s}_k\|^2}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} - \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k + \tilde{s}_k \tilde{s}_k^T \|\tilde{s}_k\|^2}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} \right\}
\]

\[
= \|Q_k\|_F^2 + \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T \tilde{B}_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k} + 2 \frac{\tilde{B}_k \tilde{s}_k \tilde{s}_k^T Q_k}{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k}
\]

(56)

By the definition of \(Q_k\) and \(\tilde{B}_k\), we have

\[
\tilde{s}_k^T \tilde{B}_k Q_k \tilde{B}_k \tilde{s}_k = \tilde{s}_k^T \tilde{B}_k (\tilde{B}_k - I) \tilde{B}_k \tilde{s}_k
\]

\[
= \tilde{s}_k^T \tilde{B}_k^2 \tilde{s}_k - \tilde{s}_k^T \tilde{B}_k^2 \tilde{s}_k
\]

\[
= \tilde{s}_k^T \tilde{B}_k^2 \tilde{s}_k - \|\tilde{B}_k \tilde{s}_k\|^2
\]

and

\[
\tilde{s}_k^T Q_k \tilde{s}_k = \tilde{s}_k^T \tilde{B}_k \tilde{s}_k - \|\tilde{s}_k\|^2.
\]
Thus, we get from (56)

\[
\frac{1}{s_k^T B_k s_k} \|s_k\|^2 F \leq \frac{1}{s_k^T B_k s_k} \left( \frac{\|B_k s_k\|^4}{(s_k^T B_k s_k)^2} + 1 - 2 \frac{s_k^T B_k^3 s_k}{s_k^T B_k s_k} \right)
\]

\[
+ 2 \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} - 2 \frac{s_k^T B_k^3 s_k}{\|s_k\|^2} + 2 \frac{s_k^T B_k^3 s_k}{\|s_k\|^2} - 2
\]

\[
= \frac{\|Q_k\|^2}{F} + 2 \left[ \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} - \frac{s_k^T B_k^3 s_k}{s_k^T B_k s_k} \right] - \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} - \frac{1}{s_k^T B_k s_k}
\]

\[
\leq \frac{\|Q_k\|^2}{F}
\]

(57)

where the inequality holds because

\[
\|B_k s_k\|^2 = [B_k^2 s_k]^T [B_k^2 s_k] \leq \|B_k^2 s_k\| \|B_k^2 s_k\| = [s_k^T B_k^2 s_k]^2 [s_k^T B_k s_k]^2.
\]

For the second term of (55) we have

\[
\left\| \frac{s_k s_k^T}{\|s_k\|^2} \frac{\tilde{y}_k y_k^T}{\tilde{y}_k^T s_k} \right\|_F \leq \left\| \left( \frac{1}{\|s_k\|^2} - \frac{1}{\tilde{y}_k^T s_k} \right) \frac{s_k s_k^T}{\|s_k\|^2} \right\|_F + \left\| \frac{s_k s_k^T - \tilde{y}_k \tilde{y}_k^T}{\tilde{y}_k^T s_k} \right\|_F
\]

\[
\leq \frac{\|\tilde{y}_k - s_k\| \|\tilde{y}_k\|}{\tilde{y}_k^T s_k} + \frac{\|s_k - \tilde{y}_k\|^2}{\tilde{y}_k^T s_k} + \frac{(\tilde{y}_k - \tilde{y}_k) \tilde{y}_k^T}{\tilde{y}_k^T s_k}
\]

\[
\leq \frac{2\|P^{-1} s_k\| + \|P y_k\| \|P y_k - P^{-1} s_k\|}{\tilde{y}_k^T s_k}
\]

\[
\leq \frac{(2\|P^{-1}\| + \|P\|\|y_k\|) \|P\| \|y_k - \nabla g^2 s_k\|}{\tilde{y}_k^T s_k}
\]

\[
\leq \frac{1}{m_k} (2\|P^{-1}\| + \|P\| M^2) \|P\| M_{\delta_k}
\]

(58)

where \( M_{\delta} = \frac{1}{m_k} (2\|P^{-1}\| + \|P\| M^2) \|P\| M_{\delta_k} \) and the last inequality follows from (22), (25), and (49). Since \( \|Q_k\|_F = \|B_k - \nabla g^2\|_F \) and \( \|Q_k+1\|_F = \|B_{k+1} - \nabla g^2\|_F \), (51) follows from (55), (57), and (58).

Now we verify (52). The inverse update formula of the BFGS method is repre-
sent as

\[ H_{k+1} = H_k + \frac{(s_k - H_k y_k) s_k^T}{y_k^T s_k} + s_k (s_k - H_k y_k)^T y_k^T s_k \]

\[ = \left( I - \frac{s_k y_k}{y_k^T s_k} \right) H_k \left( I - \frac{y_k s_k^T}{y_k^T s_k} \right) + \frac{s_k s_k^T}{y_k^T s_k}, \]

which is the dual form of the DFP update formula in the sense that \( H_k \leftrightarrow B_k \), \( H_{k+1} \leftrightarrow B_{k+1} \), and \( s_k \leftrightarrow y_k \). Also, from (19) and (49) we deduce that

\[ \| P y_k \| \geq \frac{1}{M} \| y_k \| \geq \frac{m_2}{M} \| s_k \|, \]

(59)

\[ \| P y_k - P^{-1} s_k \| \leq \| P \| \| y_k - \nabla g^2 s_k \| \leq M_1 \nu_k \| P \| \| s_k \|, \]

and

(60)

\[ \| s_k - \nabla g^{-2} y_k \| \leq \| P^2 \| \| y_k - \nabla g^2 s_k \| \leq M_1 \nu_k \| P^2 \| \| s_k \|. \]

Since \( \nu_k \to 0 \), the condition of Lemma 3.1 in [4] is satisfied (with the identification \( s \leftrightarrow y_k \), \( y \leftrightarrow s_k \), \( B \leftrightarrow H_k \), \( A \leftrightarrow \nabla g^{-2} \), and \( M \leftrightarrow P^{-1} \)). Therefore, there are constants \( \alpha_1 > 0, \alpha_2 > 0 \), and \( \omega \in (0, 1) \) such that

\[ \| H_{k+1} - \nabla g^{-2} \|_{P^{-1}} \leq \left( \sqrt{1 - \omega \mu_k^2} + \alpha_1 \frac{\| P^{-1} s_k - P y_k \|}{\| P y_k \|} \right) \| H_k - \nabla g^{-2} \|_{P^{-1}} \]

(62)

\[ + \alpha_2 \frac{\| s_k - \nabla g^{-2} y_k \|}{\| P y_k \|}, \]

where \( \mu_k \) is defined by (53). Notice that \( \sqrt{1 - \omega \mu_k^2} \leq 1 - \frac{1}{2} \omega \mu_k^2 \). In view of (59), (60), and (61), we get (52).

Finally, by Lemma 3.4 in [4] and (42), we see that \( \{ \| B_k - \nabla g^2 \|_P \} \) and \( \{ \| H_k - \nabla g^{-2} \|_{P^{-1}} \} \) converge. In particular, \( \{ \| B_k \| \} \) and \( \{ \| H_k \| \} \) are bounded. \( \square \)

Now we prove the superlinear convergence theorem for Algorithm 1.

**THEOREM 3.9.** Let Assumptions A and B hold. Then

\[ \lim_{k \to \infty} \frac{\| (B_k - \nabla g^2) s_k \|}{\| s_k \|} = 0. \]

Moreover, \( \{ x_k \} \) converges superlinearly and \( \lambda_k \equiv 1 \) for all \( k \) sufficiently large.

**Proof.** We rewrite (52) as

\[ \frac{1}{2} \omega \mu_k^2 \| H_k - \nabla g^{-2} \|_{P^{-1}} \]

\[ \leq \| H_k - \nabla g^{-2} \|_{P^{-1}} - \| H_{k+1} - \nabla g^{-2} \|_{P^{-1}} + M_0 \| H_k - \nabla g^{-2} \|_{P^{-1}} \nu_k + M_1 \nu_k. \]

Notice that \( \{ \| H_k - \nabla g^{-2} \|_{P^{-1}} \} \) is bounded and \( \nu_k = \nu_k(\alpha) \) satisfies (43). Summing the above inequalities, we get

\[ \frac{1}{2} \omega \sum_{k=0}^{\infty} \mu_k^2 \| H_k - \nabla g^{-2} \|_{P^{-1}} < \infty. \]
In particular, we have by the definition (53) of \( \mu_k \)

\[
\lim_{k \to \infty} \mu_k^2 \| H_k - \nabla g_*^{-2} \|_{p-1} = \lim_{k \to \infty} \frac{\| P^{-1}(H_k - \nabla g_*^{-2})y_k \|}{\| H_k - \nabla g_*^{-2} \|_{p-1} \| Py_k \|} = 0.
\]

However, since \( \| H_k - \nabla g_*^{-2} \|_{p-1} \) is bounded, it follows that

\[
\lim_{k \to \infty} \frac{\| P^{-1}(H_k - \nabla g_*^{-2})y_k \|}{\| Py_k \|} = 0.
\]

(64)

By (19) and (22), we have

\[
\| Py_k \| = \| \nabla g_*^{-1} y_k \| \leq \frac{1}{m} \| y_k \| \leq \frac{M^2}{m} \| s_k \|.
\]

By (18) we have

\[
\| P^{-1}(H_k - \nabla g_*^{-2})y_k \| = \| \nabla g_*(H_k - \nabla g_*^{-2})y_k \| \geq m \| (H_k - \nabla g_*^{-2})y_k \|.
\]

Thus, (64) implies

(65)

\[
\lim_{k \to \infty} \frac{\| (H_k - \nabla g_*^{-2})y_k \|}{\| s_k \|} = 0.
\]

On the other hand, we have

\[
\| (H_k - \nabla g_*^{-2})y_k \| = \| H_k(\nabla g_*^2 - B_k)\nabla g_*^{-2} y_k \|
\geq \| H_k(\nabla g_*^2 - B_k)s_k \| - \| H_k(\nabla g_*^2 - B_k)(s_k - \nabla g_*^{-2} y_k) \|
= \| H_k(\nabla g_*^2 - B_k)s_k \| - \| H_k(\nabla g_*^2 - B_k)\nabla g_*^{-2}(y_k - \nabla g_*^2 s_k) \|
\geq \| H_k(\nabla g_*^2 - B_k)s_k \| - M \| H_k(\nabla g_*^2 - B_k)\nabla g_*^{-2} \| s_k \|
= \| H_k(\nabla g_*^2 - B_k)s_k \| - o(\| s_k \|),
\]

where the last inequality follows from (49). Notice that \( \{ \| B_k \| \} \) and \( \{ \| H_k \| \} \) are bounded and \( \{ H_k \} \) is uniformly nonsingular. Therefore, there is a constant \( m_3 > 0 \) such that \( \| H_k(\nabla g_*^2 - B_k)s_k \| \geq m_3 \| (\nabla g_*^2 - B_k) s_k \| \) for all \( k \). Thus, we have

\[
\| (H_k - \nabla g_*^{-2})y_k \| \geq m_3 \| (\nabla g_*^2 - B_k) s_k \| - o(\| s_k \|),
\]

and hence (65) yields (63). In view of Lemma 3.5, the proof is complete.

Lemma 3.5 and Theorem 3.9 show that when Assumptions A and B are met, (13) holds for all \( k \) sufficiently large. This means that when \( k \) is sufficiently large, Algorithm 1 has a norm descent property.

4. Numerical experiments. In this section, we report results of some preliminary numerical experiments with the proposed method. We solved the following two problems with various sizes.

Problem 1. The discretized two-point boundary value problem [14]

\[
g(x) \triangleq Ax + \frac{1}{(n + 1)^2} F(x) = 0,
\]
where $A$ is the $n \times n$ tridiagonal matrix given by

$$
A = \begin{pmatrix}
2 & -1 & & & \\
-1 & 2 & -1 & & \\
& \ddots & \ddots & \ddots & \\
& & \ddots & \ddots & -1 \\
& & & -1 & 2 \\
\end{pmatrix},
$$

and $F(x) = (F_1(x), F_2(x), \ldots, F_n(x))^T$ with

$$
F_i(x) = \sin x_i - 1, \quad i = 1, 2, \ldots, n.
$$

**Problem 2.** Unconstrained optimization problem

$$
\min f(x), \quad x \in \mathbb{R}^n,
$$

with Engval function [18] $f : \mathbb{R}^n \to \mathbb{R}$ defined by

$$
f(x) = \sum_{i=2}^n \{(x_{i-1}^2 + x_i^2)^2 - 4x_{i-1} + 3\}.
$$

The related symmetric nonlinear equation is

$$
g(x) \triangleq \frac{1}{4} \nabla f(x) = 0,
$$

where $g(x) = (g_1(x), g_2(x), \ldots, g_n(x))^T$ with

$$
g_1(x) = x_1(x_1^2 + x_2^2) - 1, \\
g_i(x) = x_i(x_{i-1}^2 + 2x_i^2 + x_{i+1}^2) - 1, \quad i = 2, 3, \ldots, n-1, \\
g_n(x) = x_n(x_{n-1}^2 + x_n^2).
$$

In the experiments, the parameters in Algorithm 1 were chosen as $r = 0.1$, $\rho = \sqrt{0.9}$, $\sigma_1 = \sigma_2 = 10^{-5}$, $\lambda_1 = 0.01$, and $\epsilon_k = k^{-2}$, and the initial matrix $B_0$ was always set to be the unit matrix. The program was coded in FORTRAN77. We stopped the iteration when the condition $\|g(x_k)\| \leq 10^{-5}$ was satisfied. Tables 1 and 2 show the number of iterations the method needed to solve Problems 1 and 2, respectively.

The numerical results indicate that the proposed method performs quite well for Problem 1. Moreover, the initial points do not influence the number of iterations very much. The method is also successful for Problem 2. We also tested the method on some other problems. For example, we use the method to solve the nonlinear equation derived from the unconstrained optimization problem with Powell function [18]. Although the method converged to a solution of the equation for some starting points, it failed to locate a solution for many other starting points. We observed that when the method failed, the matrix $B_k$ became very small and hence $B_k^{-1}$ became large as the iteration proceeded. We also observed that when the method stalled, Jacobian matrices of $g$ were usually almost singular. This indicates that the conditions for global convergence given in Theorem 3.3 might be violated.
<table>
<thead>
<tr>
<th>$n=9$</th>
<th>$n=13$</th>
<th>$n=12$</th>
<th>$n=14$</th>
<th>$n=16$</th>
<th>$n=15$</th>
<th>$n=16$</th>
<th>$n=16$</th>
<th>$n=16$</th>
</tr>
</thead>
<tbody>
<tr>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
</tr>
<tr>
<td>91</td>
<td>91</td>
<td>91</td>
<td>91</td>
<td>91</td>
<td>91</td>
<td>91</td>
<td>91</td>
<td>91</td>
</tr>
<tr>
<td>00001...00001-</td>
<td>00001...00001</td>
<td>...</td>
<td>00001...00001</td>
<td>00001...00001</td>
<td>...</td>
<td>00001...00001</td>
<td>00001...00001</td>
<td>...</td>
</tr>
<tr>
<td>329</td>
<td>329</td>
<td>329</td>
<td>329</td>
<td>329</td>
<td>329</td>
<td>329</td>
<td>329</td>
<td>329</td>
</tr>
<tr>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
</tr>
<tr>
<td>31</td>
<td>31</td>
<td>31</td>
<td>31</td>
<td>31</td>
<td>31</td>
<td>31</td>
<td>31</td>
<td>31</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 1**

Computational results for Problem 1.
5. Concluding remarks. We have presented a Gauss–Newton-based BFGS method for solving symmetric nonlinear equations. We have shown that the proposed method converges globally if Jacobian matrices are uniformly nonsingular, and the convergence rate is superlinear if in addition the Jacobian is Hölder continuous. However, the uniform nonsingularity of Jacobian matrices may be a restrictive requirement in practice. In [13], we modified the method in a way similar to [12] and established its global convergence under the weaker assumption that the Jacobian matrix is nonsingular at an accumulation point of the generated sequence.

Acknowledgments. The authors would like to thank Dr. N. Yamashita for his helpful suggestions on the paper. We also thank the anonymous referees for their constructive suggestions.

REFERENCES

[12] D. Li and M. Fukushima, A Modified BFGS Method and Its Global Convergence in Nonconvex Minimization, Technical report 98003, Department of Applied Mathematics and Physics,


