Journal of Applied Mathematics and Computational Mechanics 2021, 20(4), 29-41
www.amcm.pcz.pl p-ISSN 2299-9965
DOI: 10.17512/jamcm.2021.4.03 e-ISSN 2353-0588

A NEW OPTIMIZATION METHOD BASED ON PERRY'S IDEA THROUGH THE USE OF THE MATRIX POWER

Imane Hafaidia[1], Noureddine Benrabia[2], Mourad Ghiat[1], Hamza Guebbai[1]

[1] Laboratoire de Mathématiques Appliquées et de Modélisation, Université 8 Mai 1945 Guelma B.P. 401 Guelma 24000 Algérie
[2] Département de Mathématiques, Université Mohamed-Chérif Messaadia Souk Ahras B.P. 1553, 41000, Souk Ahras, Algeria
hafaidia.imane@yahoo.com; hafaidia.imane@univ-guelma.dz, noureddinebenrabia@yahoo.fr, mourad.ghi24@gmail.com; ghiat.mourad@univ-guelma.dz, guebaihamza@yahoo.fr; guebbai.hamza@univ-guelma.dz

Received: 6 August 2021; Accepted: 12 December 2021

Abstract. The purpose of this paper is to present a new conjugate gradient method for solving unconstrained nonlinear optimization problems, based on Perry's idea. An accelerated adaptive algorithm is proposed, where our search direction satisfies the sufficient descent condition. The global convergence is analyzed using the spectral analysis. The numerical results are described for a set of standard test problems, and it is shown that the performance of the proposed method is better than that of the CG-DESCENT, the mBFGS and the SPDOC.

MSC 2010: *90C30, 90C26, 90C06*
Keywords: *conjugate gradient method, symmetrical technique, spectral analysis, global convergence, unconstrained optimization*

1. Introduction

It is well known that the nonlinear conjugate gradient method is characterized by low memory requirements and strong local and global convergence properties and is more practical than other methods because it minimizes the large-scale unconstrained optimization problem [1–6]

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

where f is a sufficiently smooth function. This method generates a sequence $\{x_k\}_{k\in\mathbb{N}}$, where the starting is some $x_0 \in \mathbb{R}^n$, using the following recurrence relation

$$x_{k+1} = x_k + \alpha_k d_k, \tag{2}$$

where, α_k is the step length of the line search, and the directions d_k are given by

$$\begin{cases} d_1 = -g_1 \\ d_{k+1} = -g_{k+1} + \beta_k d_k, \quad \forall k \geq 1, \end{cases} \tag{3}$$

where, $g_k = g(x_k) = \nabla f(x_k)$, and β_k is a conjugate gradient parameter. Well known formulas for β_k include the Hestenes-Stiefel (HS) [7], the Fletcher-Reeves (FR) [8], the Polak-Ribiére-Polyak (PRP) [9], the Liu-Storey (LS) [10] and the Hager-Zhang (HZ) [11].

We know that to obtain the global convergence results of the said conjugate gradient methods, it is usually required that the step size α_k should satisfy some line search conditions, such as the strong Wolfe line search

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^\top d_k, \tag{4}$$

$$\sigma g_k^\top d_k \leq g(x_k + \alpha_k d_k)^\top d_k \leq -\sigma g_k^\top d_k, \tag{5}$$

with, $0 < \delta < \dfrac{1}{2}$ and $\delta < \sigma < 1$. The search direction d_{k+1} is required to satisfy the sufficient descent condition

$$d_{k+1}^\top g_{k+1} \leq -c\|g_{k+1}\|^2, \quad c > 0. \tag{6}$$

In addition to what we mentioned, quasi-Newton methods [12,13] are shown to be sometimes effective methods for solving (1). The search direction of Quasi-Newton methods is given by

$$d_{k+1} = -H_{k+1} g_{k+1},$$

where H_{k+1} is an approximation to the inverse of the Hessian matrix $\bigtriangledown^2 f(x_k)^{-1}$. Some authors use this technique in the conjugate gradient method. By using it, Perry [14] has proposed the following formula in order to compute the parameter β_k

$$\beta_k^p = \frac{y_k^\top g_{k+1} - s_k^\top g_{k+1}}{d_k^\top y_k}, \tag{7}$$

where, $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$. By substituting (7) in (3) and applying some simple algebraic manipulations, we obtain the corresponding Perry's search direction as follows:

$$d_{k+1}^p = -\left(I - \frac{s_k y_k^\top}{y_k^\top s_k} + \frac{s_k s_k^\top}{y_k^\top s_k} \right) g_{k+1} = -P_{k+1} g_{k+1}. \tag{8}$$

In Perry's method, the matrix P_{k+1} is used to estimate the approximation of the inverse of the Hessian matrix. If the line search is exact, ($d_k^\top g_{k+1} = 0$), then (7) is identical to the Hestenes and Stiefel [7] conjugate gradient algorithm. Observing that P_{k+1} is not symmetric, then some authors have modified this matrix to meet the previous requirements using different techniques (see for example [15–17]), as Andrei [18] who presented a symmetric matrix to estimate the inverse of the Hessian matrix approximation as follows

$$P_{k+1}^{N} = I - \frac{s_k y_k^\top + y_k s_k^\top}{y_k^\top s_k} + \eta_k \frac{s_k s_k^\top}{y_k^\top s_k}, \tag{9}$$

by computing the parameter η_k in some different manner [19].

In this paper, we focus our attention on Perry's [14] observation using the (HS) choice, in which the direction d_{k+1} in (3) can be rewritten as

$$d_{k+1} = -D_{k+1} g_{k+1} = -\left(I - \frac{s_k y_k^\top}{s_k^\top y_k} \right) g_{k+1}. \tag{10}$$

The matrix D_{k+1} is a conjugate gradient iteration matrix and represents an inverse of a Hessian matrix approximation but is not symmetric. In literature, there have several symmetry procedures have been proposed like the Powell symmetrical technique [20]. Thus, based on the HS method, many variants have been developed because it has better computing performance, some of these variants are widely used in practice.

Moreover, it is very important to choose a good iteration matrix for a general nonlinear conjugate gradient method. Starting from D_{k+1}, we propose a new symmetric and positive definite matrix which always satisfies the sufficient descent condition for any line search. We also use a certain technique of an accelerated adaptation on our conjugate gradient algorithm and show that the proposed method converges globally using the spectral analysis [Spectral analysis refers to us studying the eigenvalues of the matrix M_{k+1} that comes after (for instance, check Section 2)]. Finally, we describe the numerical results.

2. The new method

In this section, we present a new algorithm, developed and adapted for solving large-scale problems, at any iteration.

The matrix D_{k+1} in (10) is not symmetric; so, we propose the following symmetric one instead

$$D_{k+1}^{sym} = \frac{D_{k+1}^\top + D_{k+1}}{2} = I - \frac{1}{2} \frac{s_k y_k^\top + y_k s_k^\top}{s_k^\top y_k}. \tag{11}$$

As we can see, our proposed matrix is symmetric, and we move to the next step that is the study of its spectra (consequently its positive definiteness). Thus, one gains an always-true sufficient condition of the descent.

Theorem 1 *Let D_{k+1}^{sym} be defined by (11). If s_k and y_k are independent linear vectors and $s_k^\top y_k \neq 0$. Then, D_{k+1}^{sym} has 1 for an eigenvalue with multiplicity $(n-2)$; and the* *remaining two eigenvalues are the maximum and the minimum ones $\mu_{\max}^{k+1}$ and $\mu_{\min}^{k+1}$ respectively and are formulated by*

$$\mu_{\max}^{k+1} = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{(s_k^\top s_k)(y_k^\top y_k)}{(s_k^\top y_k)^2}}, \tag{12}$$

$$\mu_{\min}^{k+1} = \frac{1}{2} - \frac{1}{2}\sqrt{\frac{(s_k^\top s_k)(y_k^\top y_k)}{(s_k^\top y_k)^2}}. \tag{13}$$

□

PROOF Using the following algebraic formula

$$det(I + xy^\top + uv^\top) = (1 + y^\top x)(1 + v^\top u) - (x^\top v)(y^\top u),$$

we get

$$\begin{aligned} det(D_{k+1}^{sym}) &= det\left(I - \frac{1}{2}\frac{s_k y_k^\top + y_k s_k^\top}{s_k^\top y_k}\right) \\ &= \left(1 - \frac{1}{2}\frac{s_k^\top y_k}{s_k^\top y_k}\right)\left(1 - \frac{1}{2}\frac{y_k^\top s_k}{s_k^\top y_k}\right) - \frac{1}{4}\frac{(s_k^\top s_k)(y_k^\top y_k)}{(s_k^\top y_k)^2} \\ &= \frac{1}{4} - \frac{(s_k^\top s_k)(y_k^\top y_k)}{4(s_k^\top y_k)^2}. \end{aligned} \tag{14}$$

Therefore, the matrix D_{k+1}^{sym} is nonsingular when $\frac{(s_k^\top s_k)(y_k^\top y_k)}{(s_k^\top y_k)^2} > 1$.

The matrix D_{k+1}^{sym} has the eigenvalue 1 (with multiplicity $(n-2)$).

Since $\forall \zeta \in span\{y_k, s_k\}^\perp$

$$D_{k+1}^{sym}\zeta = \left(I - \frac{1}{2}\frac{s_k y_k^\top + y_k s_k^\top}{s_k^\top y_k}\right)\zeta = \zeta - \frac{y_k^\top \zeta}{2 s_k^\top y_k} s_k - \frac{s_k^\top \zeta}{2 s_k^\top y_k} y_k = \zeta.$$

From formula (11), we can get the trace of D_{k+1}^{sym} as follows

$$\begin{aligned} tr(D_{k+1}^{sym}) &= tr\left(I - \frac{1}{2}\frac{s_k y_k^\top + y_k s_k^\top}{s_k^\top y_k}\right) \\ &= n - \frac{1}{2}\frac{y_k^\top s_k + s_k^\top y_k}{s_k^\top y_k} \\ &= n - 1 \\ &= \underbrace{1 + ... + 1}_{(n-2)times} + \mu_{\max}^{k+1} + \mu_{\min}^{k+1}. \end{aligned} \tag{15}$$

Therefore, we obtain that

$$\mu_{\max}^{k+1} + \mu_{\min}^{k+1} = 1. \tag{16}$$

Also, for the determinant relations

$$det(D_{k+1}^{sym}) = \mu_{\max}^{k+1}\mu_{\min}^{k+1},$$

and thus

$$\mu_{\max}^{k+1}\mu_{\min}^{k+1} = \frac{1}{4} - \frac{(s_k^\top s_k)(y_k^\top y_k)}{4(s_k^\top y_k)^2}. \tag{17}$$

From (16) and (17), we construct the following quadratic equation

$$\mu^2 - \mu + \frac{1}{4} - \frac{(s_k^\top s_k)(y_k^\top y_k)}{4(s_k^\top y_k)^2} = 0.$$

Thus, the other two eigenvalues are determined by (12) and (13), respectively. ∎

From the previous theorem and $\frac{(s_k^\top s_k)(y_k^\top y_k)}{(s_k^\top y_k)^2} > 1$, we get $\mu_{\min}^{k+1} \leq 0$. So, in conclusion, the matrix D_{k+1}^{sym} is not positive definite.

To render the matrix D_{k+1}^{sym} positive definite, we need to raise its power to $2p$, $p \in \mathbb{N}^\star$. To this end, let the formula be

$$M_{k+1} = (D_{k+1}^{sym})^{2p} = \left(I - \frac{1}{2}\frac{s_k y_k^\top + y_k s_k^\top}{s_k^\top y_k}\right)^{2p}. \tag{18}$$

Then, applying the previous theorem, the eigenvalues of M_{k+1} are similar to those of D_{k+1}^{sym} only that its maximum and minimum ones (λ_{k+1}^+ and λ_{k+1}^-, respectively) are given by $\lambda_{k+1}^+ = \left(\mu_{\max}^{k+1}\right)^{2p}, \quad \lambda_{k+1}^- = \left(\mu_{\min}^{k+1}\right)^{2p}$.

However, if s_k and y_k were linearly dependent vectors, i.e $s_k = \sigma y_k$, then M_{k+1} would be reformulated as

$$M_{k+1} = \left(D_{k+1}^{sym} + \left(1 - \mu_{\min}^{k+1}\right) I\right)^{2p},$$

that yields

$$M_{k+1} = \left(2I - \frac{y_k y_k^\top}{y_k^\top y_k}\right)^{2p}. \tag{19}$$

Then, the eigenvalues of the matrix M_{k+1} consist of 2^{2p} with ($n-1$ multiplicity), and $\lambda_{k+1}^- = 1$, because D_{k+1}^{sym} is diagonalizable for any function H defined on $Sp(D_{k+1}^{sym}) \subset D(H)$, then $sp(H(D_{k+1}^{sym})) = H(Sp(D_{k+1}^{sym}))$. In the previous case, we took $H(a) = (a + (1 - \mu_{\min}^{k+1}))^{2p}$, to obtain the above eigenvalues.

From the simple adaptive strategy applied to the matrix D_{k+1}, we have the following search direction

$$d_{k+1} = -M_{k+1} g_{k+1}, \tag{20}$$

where

$$M_{k+1} = \begin{cases} (19) & \text{if} \quad s_k = \sigma y_k \\ (18) & \text{otherwise.} \end{cases} \tag{21}$$

The next theorem implies that our method satisfies the sufficient descent condition.

Theorem 2 *Let the sequence* $\{d_{k+1}\}_{k\in\mathbb{N}}$ *be generated by (20). Then the search direction satisfies the sufficient descent condition*

$$d_{k+1}^\top g_{k+1} \le -c\|g_{k+1}\|^2, \quad c>0.$$

PROOF For all $k \ge 1$, we have from (20) and the fact that M_{k+1} is a symmetric, positive definite matrix,

$$d_{k+1}^\top g_{k+1} = -g_{k+1}^\top M_{k+1} g_{k+1} \le -\lambda_{k+1}^- \|g_{k+1}\|^2. \tag{22}$$

This shows that the descent condition is satisfied. ∎

3. The acceleration of the new conjugate gradient algorithm

We know that the best features of the conjugate gradient methods are their simple iterations and low memory requirements. However, the proposed matrix in the previous section requires a large storage space which is not easy to apply in this form to a large scale unconstrained optimization problem. In order to overcome this difficulty, we propose an accelerated formula to calculate M_{k+1} more efficiently.

The next theorem shows this new reformulation of M_{k+1}.

Theorem 3 *Let* M_{k+1} *be defined by (18). Then,*

$$\left(I - \frac{1}{2}\frac{s_k y_k^\top + y_k s_k^\top}{s_k^\top y_k}\right)^{2p} = I + \eta_{2p}\left(s_k y_k^\top + y_k s_k^\top\right) + \delta_{2p} s_k s_k^\top + \gamma_{2p} y_k y_k^\top, \tag{23}$$

where

$$\begin{aligned}
\eta_{2p} &= \frac{1}{2}\left(\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p-1} + \left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p-1}\right)\left(\frac{-1}{2(s_k^\top y_k)} - \frac{1}{(4a_k b_k - 1)(s_k^\top y_k)}\right)\\
&- \frac{2b_k}{(4a_k b_k - 1)(s_k^\top y_k)}\left(\frac{-a_k}{2\sqrt{a_k b_k}}\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p-1} + \frac{a_k}{2\sqrt{a_k b_k}}\left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p-1}\right)\\
&+ \frac{1}{(4a_k b_k - 1)(s_k^\top y_k)},
\end{aligned}$$

$$\begin{aligned}
\delta_{2p} &= \frac{-\sqrt{a_k b_k}}{2a_k}\left(\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p-1} - \left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p-1}\right)\left(\frac{-4a_k b_k - 1}{2(4a_k b_k - 1)(s_k^\top y_k)}\right)\\
&- \frac{2b_k}{(4a_k b_k - 1)(s_k^\top y_k)}\left(\frac{1}{2}\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p-1} + \frac{1}{2}\left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p-1} - 1\right),
\end{aligned}$$

$$\begin{aligned}
\gamma_{2p} &= \frac{-\sqrt{a_k b_k}}{2b_k}\left(\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p-1}-\left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p-1}\right)\left(\frac{-4a_k b_k-1}{2(4a_k b_k-1)(s_k^\top y_k)}\right)\\
&- \frac{2a_k}{(4a_k b_k-1)(s_k^\top y_k)}\left(\frac{1}{2}\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p-1}+\frac{1}{2}\left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p-1}-1\right),
\end{aligned}$$

with $a_k=-\dfrac{s_k^\top s_k}{2(s_k^\top y_k)}$ *and* $b_k=-\dfrac{y_k^\top y_k}{2(s_k^\top y_k)}$. □

PROOF We give a proof by induction. For $p=1$,

$$\left(I-\frac{1}{2}\frac{s_k y_k^\top+y_k s_k^\top}{s_k^\top y_k}\right)^2 = I-\frac{3}{4(s_k^\top y_k)}s_k y_k^\top-\frac{3}{4(s_k^\top y_k)}y_k s_k^\top+\frac{y_k^\top y_k}{4(s_k^\top y_k)^2}s_k s_k^\top+\frac{s_k^\top s_k}{4(s_k^\top y_k)^2}y_k y_k^\top.$$

We, then assume that for any $p\geqslant 1$, $\left(I-\dfrac{1}{2}\dfrac{s_k y_k^\top+y_k s_k^\top}{s_k^\top y_k}\right)^{2p}$ verifies (23), and we show that $\left(I-\dfrac{1}{2}\dfrac{s_k y_k^\top+y_k s_k^\top}{s_k^\top y_k}\right)^{2p+1}$ also holds.

$$\begin{aligned}
&\left(I-\frac{1}{2}\frac{s_k y_k^\top+y_k s_k^\top}{s_k^\top y_k}\right)^{2p+1} =\\
&= \left(I+\eta_{2p}s_k y_k^\top+\eta_{2p}y_k s_k^\top+\delta_{2p}s_k s_k^\top+\gamma_{2p}y_k y_k^\top\right)\left(I-\frac{s_k y_k^\top+y_k s_k^\top}{2(s_k^\top y_k)}\right) =\\
&= I+\left(\frac{-1}{(s_k^\top y_k)}+\frac{\eta_{2p}}{2}-\frac{\delta_{2p}s_k^\top s_k}{2(s_k^\top y_k)}\right)s_k y_k^\top+\left(\frac{-1}{(s_k^\top y_k)}+\frac{\eta_{2p}}{2}-\frac{\gamma_{2p}y_k^\top y_k}{2(s_k^\top y_k)}\right)y_k s_k^\top\\
&\quad +\left(\frac{\delta_{2p}}{2}-\frac{\eta_{2p}y_k^\top y_k}{2(s_k^\top y_k)}\right)s_k s_k^\top+\left(\frac{\gamma_{2p}}{2}-\frac{\eta_{2p}s_k^\top s_k}{2(s_k^\top y_k)}\right)y_k y_k^\top,
\end{aligned}$$

where, similarly, we get

$$\begin{aligned}
\eta_{2p+1} &= \left(\frac{-1}{(s_k^\top y_k)}+\frac{\eta_{2p}}{2}-\frac{\delta_{2p}s_k^\top s_k}{2(s_k^\top y_k)}\right)=\left(\frac{-1}{(s_k^\top y_k)}+\frac{\eta_{2p}}{2}-\frac{\gamma_{2p}y_k^\top y_k}{2(s_k^\top y_k)}\right)\\
&= \frac{1}{2}\left(\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p}+\left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p}\right)\left(\frac{-1}{2(s_k^\top y_k)}-\frac{1}{(4a_k b_k-1)(s_k^\top y_k)}\right)\\
&- \frac{2b_k}{(4a_k b_k-1)(s_k^\top y_k)}\left(\frac{-a_k}{2\sqrt{a_k b_k}}\left(\frac{1}{2}-\sqrt{a_k b_k}\right)^{2p}+\frac{a_k}{2\sqrt{a_k b_k}}\left(\frac{1}{2}+\sqrt{a_k b_k}\right)^{2p}\right)\\
&+ \frac{1}{(4a_k b_k-1)(s_k^\top y_k)},
\end{aligned}$$

$$
\begin{aligned}
\delta_{2p+1} &= \left(\frac{\delta_{2p}}{2} - \frac{\eta_{2p} y_k^\top y_k}{2(s_k^\top y_k)}\right) \\
&= \frac{-\sqrt{a_k b_k}}{2a_k}\left(\left(\frac{1}{2} - \sqrt{a_k b_k}\right)^{2p} - \left(\frac{1}{2} + \sqrt{a_k b_k}\right)^{2P}\right)\left(\frac{-4a_k b_k - 1}{2(4a_k b_k - 1)(s_k^\top y_k)}\right) \\
&- \frac{2b_k}{(4a_k b_k - 1)(s_k^\top y_k)}\left(\frac{1}{2}\left(\frac{1}{2} - \sqrt{a_k b_k}\right)^{2p} + \frac{1}{2}\left(\frac{1}{2} + \sqrt{a_k b_k}\right)^{2p} - 1\right),
\end{aligned}
$$

$$
\begin{aligned}
\gamma_{2p+1} &= \left(\frac{\gamma_{2p}}{2} - \frac{\eta_{2p} s_k^\top s_k}{2(s_k^\top y_k)}\right) \\
&= \frac{-\sqrt{a_k b_k}}{2b_k}\left(\left(\frac{1}{2} - \sqrt{a_k b_k}\right)^{2p} - \left(\frac{1}{2} + \sqrt{a_k b_k}\right)^{2p}\right)\left(\frac{-4a_k b_k - 1}{2(4a_k b_k - 1)(s_k^\top y_k)}\right) \\
&- \frac{2a_k}{(4a_k b_k - 1)(s_k^\top y_k)}\left(\frac{1}{2}\left(\frac{1}{2} - \sqrt{a_k b_k}\right)^{2p} + \frac{1}{2}\left(\frac{1}{2} + \sqrt{a_k b_k}\right)^{2p} - 1\right),
\end{aligned}
$$

which completes the proof. ■

The previous theorem allows us to rewrite M_{k+1} as

$$
M_{k+1} = I + \eta_{2p}\left(s_k y_k^\top + y_k s_k^\top\right) + \delta_{2p} s_k s_k^\top + \gamma_{2p} y_k y_k^\top, \tag{24}
$$

which makes it more suitable for numerical programming. According to Theorem 3, M_{k+1} can be modified as

$$
M_{k+1} = \begin{cases} (19) & \text{if} \quad s_k = \sigma y_k \\ (24) & \text{otherwise.} \end{cases} \tag{25}
$$

Additionally, we obtain the descent conjugate gradient algorithm as

Algorithm 1 Generalization of Perry's Powers (GPP)

Step 1. Give an initial point x_0 and $\varepsilon \geq 0$. Set $k = 0$.
Step 2. Calculate $g_0 = g(x_0)$. If $\|g_k\| \leq \varepsilon$, then stop, otherwise let $d_0 = -g_0$ and continue with **Step 3**.
Step 3. Calculate the step length α_k with strong Wolfe line search conditions (4) and (5).
Step 4. Set $x_{k+1} = x_k + \alpha_k d_k$.
Step 5. Calculate $g_{k+1} = g(x_{k+1})$.
Step 6. If $\|g_{k+1}\| \leq \varepsilon$, then stop.
Step 7. Calculate the direction d_{k+1} via (20) where M_{k+1} is computed by (25). Set $k = k + 1$, then go to **Step 3**.

4. Global convergence result

In this section, we analyze the our algorithm's global convergence of our algorithm using spectral theory tools. Earlier, we introduced the following hypotheses about the objective function $f(x)$.

H1 f is bounded below in $\mathbb{R}^n$ and is continuously differentiable in a neighborhood N of the level set $S = \{x \in \mathbb{R}^n \quad f(x) \leq f(x_0)\}$, where x_0 is the starting point of the iteration.

H2 The gradient of f is Lipschitz continuous over N, i.e. there is a constant $L > 0$ such that

$$\|\nabla f(\tilde{x}) - \nabla f(x)\| \leq L\|\tilde{x} - x\|.$$

Lemma 1 *Supposing that the hypotheses* **H1** *and* **H2** *are satisfied and the sequence* $\{x_k\}_k$ *is generated by (2) and* α_k *are determined such that the Wolfe conditions hold, the Zoutendijk condition is*

$$\sum_{k=0}^{\infty} \frac{(g_k^\top d_k)^2}{\|d_k\|^2} < +\infty.$$

PROOF See [21]. ■

Next, for an objective function satisfying **H1** and **H2**, the spectral condition theorem of the global convergence in [20] is introduced as:

Theorem 4 *Let the objective function* $f(x)$ *satisfy* **H1** *and* **H2**. *For the nonlinear conjugate gradient method, its iterative sequence is generated by (2) and its line search directions are calculated by*

$$\begin{cases} d_1 = -g_1 \\ d_{k+1} = -M_{k+1} g_{k+1}, \quad \forall k \geq 1, \end{cases} \tag{26}$$

such that the sufficient descent condition (6) holds, and that α_k *are determined in a way such that the Wolfe conditions hold and*

$$\sum_{k=1}^{\infty} (\lambda_{k+1}^{+})^{-2} = +\infty, \tag{27}$$

where, λ_{k+1}^{+} *is the maximum eigenvalue of* M_{k+1}. *Then*

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{28}$$

Moreover, if $\lambda_{k+1}^{+} \leq \Lambda$, *where* Λ *is a positive constant, then*

$$\lim_{k\to\infty} \|g_k\| = 0. \tag{29}$$

Remark 1 If M_{k+1} is an symmetric positive definite matrix, then the spectral condition (27) can be rewritten as

$$\sum_{k=1}^{\infty} (\kappa_2(M_{k+1}))^{-2} = +\infty, \tag{30}$$

where κ_2 is the spectral condition number of M_{k+1}. □

PROOF Supposing that, by contradiction, there exists $\gamma > 0$ such that $\|g_k\| \geq \gamma$ for all $k \geq 1$. Then, from (26) and the fact that M_{k+1} is an symmetric positive definite matrix, it follows that

$$\|d_{k+1}\|^2 = g_{k+1}^\top M_{k+1}^\top M_{k+1} g_{k+1} \leq \left(\lambda_{k+1}^+\right)^2 \|g_k\|^2, \tag{31}$$

and that,

$$\cos^2\theta_k = \frac{(g_{k+1}^\top d_{k+1})^2}{\|d_{k+1}\|^2 \|g_{k+1}\|^2} \geq \frac{\left(\lambda_{k+1}^-\right)^2 \|g_{k+1}\|^4}{\left(\lambda_{k+1}^+\right)^2 \|g_{k+1}\|^4} = (\kappa_2(M_{k+1}))^{-2},$$

where, θ_k is the angle between d_{k+1} and $(-g_{k+1})$. Thus,

$$\sum_{k\geq 0} \|g_{k+1}\|^2 cos^2\theta_k \geq \gamma^2 \sum_{k=0}^{\infty} (\kappa_2(M_{k+1})^{-2} = +\infty, \tag{32}$$

which contradicts the Zoutendijk's condition

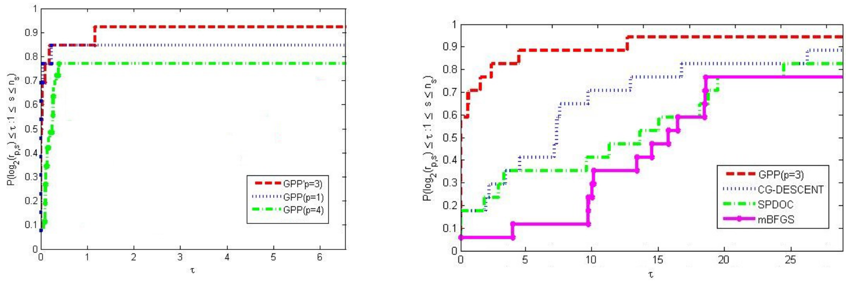
$$\sum_{k\geq 0} \|g_{k+1}\|^2 cos^2\theta_k \leq \sum_{k=0}^{\infty} \frac{(g_{k+1}^\top d_{k+1})^2}{\|d_{k+1}\|^2} < +\infty \tag{33}$$

This latter contradiction implies that the results of Lemma 1 are true.
Hence, $\liminf_{k\to\infty} \|g_k\| = 0$. ■

5. Numerical results

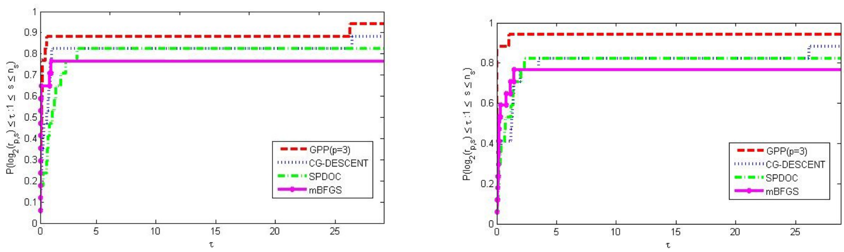
In this section, we discuss the efficiency of our new version of GPP algorithm by comparing it with the CG-DESCENT algorithm of Hager and Zhang [22], the mBFGS algorithm [23] and the SPDOC algorithm [24]. To determine the performance of all algorithms on a set of unconstrained optimization test problems [25], each problem is tested for a number of variables: 2, 10, 50, 100, 1000, 1500, 2000, 5000, and 10000 so that the total number of test problems is the 80 unconstrained problems. We run them on a PC with the next specifications Intel(R) core (TM)i5 CPU 650 @ 3.20 GHz, 3.00 Go RAM. Using the strong Wolfe line search conditions with $\delta = 0.0001$, $\sigma = 0.1$ and the termination criterion for all the algorithms $\|g_k\|^2 \leq 10^{-6}$, we adopt the performance profiles given by Dolan and Moré [26] to compare the performance.

Before doing so, we choose the best value of p. As it is shown in Figure 1a, the new method with $p = 3$ performs better than $p = 1$, $p = 4$. Figures 1b and 2a,b represent the performance profile measured by CPU time, the number of iterations and the number of functions and gradient evaluations, respectively. All figures show that the proposed algorithm in this paper performs substantially better than that of the CG-DESCENT, the mBFGS and SPDOC.



(a) (b)

Fig. 1. Performance profile for CPU time



(a) (b)

Fig. 2. Performance profile for the number of iterations (a), functions and gradient evaluations (b)

6. Conclusion

In this paper, we have proposed a conjugate gradient method based on Perry's idea with modification. An important property of our proposed method is to ensure the sufficient descent using any line search, and we showed that it is globally convergent for general functions. We confirmed the effectiveness of our method using the performance profile. By varying the exponent $p \in \mathbb{N}^*$, we had found that $p = 3$ was seemingly the optimal one for bettering the performance on, virtually, all the scales (CPU time, the number of iterations and the number of functions and gradient evaluations).

In our future works, we will consider more thoroughly exploring and investigating the possible reasons why $p = 3$ is (seemingly) the one exponent choice that renders our method's performance as well as possible.

Acknowledgements

The authors are grateful to the Editor and reviewer for their valuable comments and suggestions on the original version of this paper. We are particularly grateful for the assistance given by PhD student MAHCENE MOUHAMED GHAITH of the laboratory of the Laboratoire de Mathématiques Appliquées et de Modélisation, Université 8 Mai 1945 Guelma. B.P. 401 Guelma 24000 Algérie.

References

[1] Baluch, B., Salleh, Z., & Alhawarat, A. (2018). A new modified three-term Hestenes-Stiefel conjugate gradient method with sufficient descent property and its global convergence. *Journal of Optimization*, *6*, 1-13.

[2] Sabi'u, J., Muangchoo, K., Shah, A., Abubakar, A.B., & Aremu, K.O. (2021). An inexact optimal hybrid conjugate gradient method for solving symmetric nonlinear equations. *Symmetry*, *13*(10), 1829.

[3] Salleh, Z., Alhamzi, G., Masmali, I., & Alhawarat, A. (2021). A modified Liu and Storey conjugate gradient method for large scale unconstrained optimization problems. *Algorithms*, *14*(8), 227.

[4] Sabi'u, J., Shah, A., & Waziri, M.Y. (2020). Two optimal Hager-Zhang conjugate gradient methods for solving monotone nonlinear equations. *Applied Numerical Mathematics*, *153*, 217-233.

[5] Waziri, M.Y., Ahmed, K., & Sabi'u, J. (2019). A family of Hager-Zhang conjugate gradient methods for system of monotone nonlinear equations. *Applied Mathematics and Computation*, *361*, 645-660.

[6] Sabi'u, J., Shah, A., & Waziri, M.Y. (2021). A modified Hager-Zhang conjugate gradient method with optimal choices for solving monotone nonlinear equations. *International Journal of Computer Mathematics*, 1-23.

[7] Hestenes, M.R., & Stiefel, E. (1952). *Methods of Conjugate Gradients for Solving Linear Systems*. (Vol. 49, No. 1), Washington, DC: NBS.

[8] Fletcher, R., & Powell, M.J. (1963). A rapidly convergent descent method for minimization. *The Computer Journal*, *6*(2), 163-168.

[9] Polak, E., & Ribiere, G. (1969). Note sur la convergence de méthodes de directions conjuguées. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, *3*(R1), 35-43.

[10] Cardenas, S. (1991). Efficient generalized conjugate gradient algorithms. I. Theory. *Journal of Optimization Theory and Applications*, *69*(1), 129-137.

[11] Hager, W.W., & Zhang, H. (2005). A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on Optimization*, *16*(1), 170-192.

[12] Abubakar, A.B., Sabi'u, J., Kumam, P., & Shah, A. (2021). Solving nonlinear monotone operator equations via modified sr1 update. *Journal of Applied Mathematics and Computing*, 1-31.

[13] Zhou, W. (2020). A modified BFGS type quasi-Newton method with line search for symmetric nonlinear equations problems. *Journal of Computational and Applied Mathematics*, *367*, 112454.

[14] Perry, A. (1978). A modified conjugate gradient algorithm. *Operations Research*, *26*(6), 1073-1078.

[15] Livieris, I.E., & Pintelas, P. (2015). A modified Perry conjugate gradient method and its global convergence. *Optimization Letters*, *9*(5), 999-1015.

[16] Andrei, N. (2017). Accelerated adaptive Perry conjugate gradient algorithms based on the self-scaling memoryless BFGS update. *Journal of Computational and Applied Mathematics*, *325*, 149-164.

[17] Waziri, M.Y., Hungu, K.A., & Sabi'u, J. (2020). Descent Perry conjugate gradient methods for systems of monotone nonlinear equations. *Numerical Algorithms*, *85*(3), 763-785.

[18] Andrei, N.(2007). Scaled conjugate gradient algorithms for unconstrained optimization. *Computational Optimization and Applications*, *38*, 3, 401-416.

[19] Yao, S., He, D., & Shi, L. (2018). An improved Perry conjugate gradient method with adaptive parameter choice. *Numerical Algorithms*, *78*(4), 1255-1269.

[20] Dongyi, L., & Genqi, X. (2011). Applying Powell's symmetrical technique to conjugate gradient methods. *Computational Optimization and Applications*, *49*(2), 319-334.

[21] Andrei, N. (2010). Another accelerated conjugate gradient algorithm with guaranteed descent and conjugacy conditions for large-scale unconstrained optimization. No. 7. ICI Technical Report.

[22] Hager, W.W., & Zhang, H. (2006). Algorithm 851: CG-DESCENT, a conjugate gradient method with guaranteed descent. *ACM Transactions on Mathematical Software (TOMS)*, *32*(1), 113-137.

[23] Shanno, D.F. (1978). Conjugate gradient methods with inexact searches. *Mathematics of Operations Research*, *3*(3), 244-256.

[24] Liu, D., & Xu, G. (2013). Symmetric Perry conjugate gradient method. *Computational Optimization and Applications*, *56*(2), 317-341.

[25] Andrei, N. (2008). An unconstrained optimization test functions collection. *Advanced Modeling and Optimization*, *10*, 1, 147-161.

[26] Dolan, E.D., & Moré, J.J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, *91*(2), 201-213.

[27] Zoutendijk, G.(1970). Nonlinear programming, computational methods. *Integer and Nonlinear Programming*, 37-86.