International Conference on Advances in Computational Modeling and Simulation

Global Convergence of a Modified PRP Conjugate Gradient Method

Yueqin Zhang, Hao Zheng, Chuanlin Zhang*

College of Information Science and Technology, Jinan University, Guangzhou 510632, China

Abstract

In this paper, we propose a modified PRP conjugate gradient method which develops a new formula for parameter and possesses the following properties: (1) the sufficient descent property holds without any line searches; (2) this method inherits an important property of Polak-Ribière-Polyak (PRP) method; (3) under some assumable conditions, the method is globally convergent. Preliminary numerical results show that this method is very efficient.

© 2011 Published by Elsevier Ltd. Selection and/or peer-review under responsibility of Kunming University of Science and Technology Open access under CC BY-NC-ND license.

Keywords: sufficient descent; conjugate gradient method; line search; global convergence

1. Introduction

Consider the unconstrained nonlinear optimization problem

\[ \min_{x \in \mathbb{R}^n} f(x) \]  \hspace{1cm} (1.1)

where \( f: \mathbb{R}^n \to \mathbb{R} \) is a smooth, nonlinear function and its gradient \( g \) is available. Nonlinear conjugate gradient method is well suited for solving large scale problem, its iterative formula is given by

\[ x_{k+1} = x_k + \alpha_k d_k \]  \hspace{1cm} (1.2)

\[ d_k = \begin{cases} -g_k, k = 1 \\ -g_k + \beta_k d_{k-1}, k \geq 2 \end{cases} \]  \hspace{1cm} (1.3)

* Corresponding author. Tel: +86-13640697733.
E-mail address: tclzhang@jnu.edu.cn

Open access under CC BY-NC-ND license.

doi:10.1016/j.proeng.2012.01.1131
where \( g_k = \nabla f(x_k) \), \( d_k \) is the search direction, \( \alpha_k \) is a step-size obtained by a one-dimensional line search and \( \beta_k \) is a scalar. There are many formulas have been proposed to compute the scalar \( \beta_k \).

Among them, four famous formulas for \( \beta_k \) are called Fletcher-Reeves (FR)(1964), Polak-Ribière-Polyak (PRP)(1969), Hestense-Stiefel (HS)(1952), and Dai-Yuan(1999) methods are given by

\[
\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \quad (1.4)
\]
\[
\beta_k^{PRP} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2} \quad (1.5)
\]
\[
\beta_k^{HS} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} \quad (1.6)
\]
\[
\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}} \quad (1.7)
\]

respectively, where \( y_{k-1} = g_k - g_{k-1} \) and \( \| \cdot \| \) means the Euclidean norm.

In the already-existing convergence analysis and implementations of the conjugate gradient method, the Wolfe conditions, namely

\[
f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k \quad (1.8)
\]
\[
g_k(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k \quad (1.9)
\]

where \( \delta \in (0,1) \), \( \sigma \in (\delta,1) \), are often imposed on the line search, \( g_k = g(x_k) \).

Furthermore, the sufficient descent property, namely,

\[
-g_k^T d_k \geq c \|g_k\|^2 \quad (1.10)
\]

has often been used in the literature to analyze the global convergence of conjugate gradient method with inexact line searches, where \( c \) is a positive constant. The convergence behavior of (1.4), (1.5), (1.6) and (1.7) with some line search conditions has been studied by many authors for many years. The PRP method with the exact line search is not globally convergent, see Powell’s counterexample[1], but the \( \text{PRP}+ (\beta_k^{\text{PRP}+} = \max \{0, \beta_k^{\text{PRP}} \} \) method with Wolfe line search was proved globally convergent under the sufficient descent condition, the HS method is very familiar with the PRP method. The DY method is globally convergent without the descent condition, but its numerical results are not better than the PRP method.

In [2], the authors proposed a Modified FR and CD Conjugate Gradient Method, and a Modified HS Conjugate Gradient Method also was proposed in [3].

Enlightened by the above ideas, a Modified PRP Conjugate Gradient Method was proposed as follows:

\[
\beta_k^{\text{MPRP}} = \frac{\mu_1 (\|g_k\|^2 + \|g_k^T g_{k-1}\| g_k^T g_{k-1})}{\mu_1 \|g_k^T d_{k-1}\| + \mu_2 \|g_{k-1}\|^2} \quad (1.11)
\]

where \( \mu_1 \in (0, +\infty) \), \( \mu_2 \in (2\mu_1, +\infty) \), \( \mu_3 \in [\epsilon, +\infty) \), and \( \epsilon \) is a positive constant. Clearly, an important feature of \( \beta_k^{\text{MPRP}} \) is that its value is greater than zero without line search, from (1.11) we can get
\[
\| g_k \|^2 - \frac{(g_k^T g_{k-1})^2}{\| g_{k-1} \|^2} g_k^T g_{k-1} \geq \| g_k \|^2 - \frac{(g_k^T g_{k-1})^2}{\| g_{k-1} \|^2} \| g_k \|^2 g_{k-1} = 0 \quad (1.12)
\]

which, along with (1.11), gives
\[
\beta_k^{MPRP} = \frac{\mu_g (\| g_k \|^2 - \| g_{k-1} \|^2 g_k^T g_{k-1})}{\mu_1 |g_k^T d_{k-1}| + \mu_2 \| g_{k-1} \|^2} \geq \frac{\mu_1 |g_k^T d_{k-1}| + \mu_2 \| g_{k-1} \|^2}{\mu_1 |g_k^T d_{k-1}| + \mu_2 \| g_{k-1} \|^2} \geq 0 \quad (1.13)
\]

Based on this formula we also established a global convergent nonlinear conjugate gradient method with the strong Wolfe conditions, namely (1.8) and
\[
\frac{\mu_1 |g_k^T d_{k-1}| + \mu_2 \| g_{k-1} \|^2}{\mu_1 |g_k^T d_{k-1}| + \mu_2 \| g_{k-1} \|^2} \geq 0
\]

where \( \delta \in (0, 1), \sigma \in (\delta, 1) \). This paper is organized as follows. We will present a new algorithm (Algorithm 2.2), and the sufficient descent property (1.10) of Algorithm 2.2 is also given in the next section. In section 3, the global convergence results of the MPRP method are established. At last, the preliminary numerical results are reported.

2. Preliminaries and Algorithm

Throughout this paper, we assume that \( g_k \neq 0 \) for all \( k \), for otherwise a stationary point has been found.

First we give the following theorem, which illustrates that the formula (1.11) possesses the sufficient descent property for any line search.

**Theorem 2.1** Consider any method (1.2) and (1.3), where \( \beta_k = \beta_k^{MPRP} \). Then for all \( k \geq 1 \),
\[
-g_k^T d_k \geq \left( 1 - \frac{1}{\mu} \right) \| g_k \|^2 \quad (2.1)
\]

where \( \frac{\mu_2}{2\mu_1} \geq \mu > 1 \).

**Proof** If \( g_k^T d_{k-1} = 0 \) for all \( \frac{\mu_2}{2\mu_1} \geq \mu > 1 \), then we have
\[
-g_k^T d_k = \| g_k \|^2 \geq \left( 1 - \frac{1}{\mu} \right) \| g_k \|^2 \quad (2.2)
\]

Otherwise, since \( d_i = -g_i \), we have \( g_i^T d_i = -\| g_i \|^2 \), which satisfies (2.1). From (1.12) and the definition of \( \beta_k^{MPRP} \), we have
\[
-g_k^T d_k = \| g_k \|^2 - \beta_k^{MPRP} d_{k-1} \geq \left( 1 - \frac{1}{\mu} \right) \| g_k \|^2 \quad (2.3)
\]

Therefore we can deduce that (2.1) is true.
Now we can present a new descent conjugate gradient method as follows:

**Algorithm 2.2:**

- **step1:** Given \( x_0 \in \mathbb{R}^n, \epsilon \geq 0 \), set \( d_1 = -g_1, \ k = 1 \), if \( \| g_1 \| \leq \epsilon \), then stop.
- **step2:** Take \( \alpha_k \geq 0 \) by one of the above inexact line searches.
- **step3:** Let \( x_{k+1} = x_k + \alpha_k d_k \) and \( g_{k+1} = g(x_{k+1}) \). If \( g_{k+1} = g(x_{k+1}) \), then stop.
- **step4:** Compute \( \beta_k = \beta_k^{(MIP)} \) by the formula (1.11) and generate \( d_{k+1} \) by (1.3).
- **step5:** Set \( k := k + 1 \), go to step2.

Furthermore, in order to establish the global convergence results of the algorithm 2.2, the following basic assumption will be given on the objective function.

**Assumption 2.3**

(i) The level set \( \Omega = \{ x \in \mathbb{R}^n \mid f(x) \leq f(x_i) \} \) is bounded, where \( x_i \) is the starting point. (ii) In the neighborhood \( N \) of \( \Omega \), \( f \) is continuously differentiable and its gradient \( g \) is Lipschitz continuous, namely, there exists a constant \( L > 0 \), such that for any \( \forall x, x' \in N \),

\[
\| g(x) - g(x') \| \leq L \| x - x' \|
\]  

(2.4)

If \( f \) satisfies Assumption 2.3, we can get that

\[
\| g(x) \| \leq \gamma', \text{ for all } x \in \Omega
\]  

(2.5)

And

\[
\| x \| \leq \xi, \text{ for all } x \in \Omega
\]  

(2.6)

where \( \gamma' \) and \( \xi \) are some positive constants.

### 3. Convergence analysis

The following important result was given by Zoutendijk[4] and Wolfe[5].

**Lemma 3.1** Suppose that \( x_1 \) is a starting point for which Assumption 2.3 holds. Consider any iterative method (1.2), where \( d_k \) is a descent direction, and \( \alpha_k \) is obtained by Wolfe line search (1.8) and (1.9). Then

\[
\sum_{k=1}^{\infty} \frac{(g_i^T d_i)^2}{\| d_i \|^2} \leq +\infty
\]  

(3.1)

Since the level set \( \Omega \) is bounded and \( f(x_i) \) is a decreasing sequence, we can get this Lemma easily when \( \alpha_k \) is obtained by strong Wolfe line search (1.8) and (1.10).

The following property was given by Gilbert and Nocedal[6], called Property(*). This property means that the next research direction approaches to the steepest direction automatically when a small step-size generated, and the step-sizes are not produced successively.

**Property (*)** Consider any iteration method (1.2) and (1.3), and assume that the following inequality holds for all \( k \),

\[
0 < \gamma \leq \| g_i \| \leq \gamma'
\]  

(3.2)

where \( \gamma \) and \( \gamma' \) are constants. If there exist \( b > 1 \) and \( \lambda > 0 \) such that for all \( k \),

\[
|\beta_k| \leq b
\]  

(3.3)

and
\[ \|s_{k-1}\| \leq \lambda \Rightarrow |\beta_k| \leq \frac{1}{2b} \]  

(3.4)

where \( s_{k-1} = x_k - x_{k-1} \).

The following Lemma shows that the MPRP method has Property (*).

**Lemma 3.2** Suppose that the Assumption 2.3 holds. Consider any iteration method of the form (1.2) and (1.3), where \( \beta_k = \beta_k^{MPRP} \) and \( \alpha_k \) is obtained by any line search. Then the Property (*) holds.

**Proof** Consider any constant \( \gamma \) and \( y \) which satisfy (3.2).

Let \( b = \max\{1 + \varepsilon, \frac{2\mu_1\gamma^2}{\mu_3\gamma^2}\} > 1 \) and \( \lambda = \min\{\frac{(\mu_3\gamma^2)^2}{8\mu_1^2\gamma^2 L}, \frac{\mu_3\gamma^2}{4(1 + \varepsilon)\mu_1\gamma L}\} > 0 \), where \( \mu_1 \in (0, +\infty) \), \( \mu_2 \in (2\mu_1, +\infty) \), \( \mu_3 \in [\varepsilon, +\infty) \), \( \mu_2 > \mu_1 \), and \( \varepsilon \) is a positive constant.

\[ |\beta_k^{MPRP}| \leq \frac{\mu_1\|g_k\|^2 - \|g_k^T g_{k-1}\|}{\|g_{k-1}\|^2} \leq \frac{2\mu_1\|g_k\|^2}{\mu_1\|g_{k-1}\|^2} \leq \frac{2\mu_1\gamma^2}{\mu_1\gamma^2} \leq b \]  

(3.5)

When \( \|s_{k-1}\| \leq \lambda \), we can get from (2.4) that

\[ |\beta_k^{MPRP}| \leq \frac{\mu_1\|g_k\|}{\mu_1\|g_{k-1}\|^2} \leq \frac{1}{2b} \]  

(3.6)

The proof is completed.

**Lemma 3.3** [7] Suppose that the Assumption 2.3 holds. Consider any iteration method of the form (1.2) and (1.3), where \( \beta_k \geq 0 \), \( \alpha_k \) is obtained by the Wolfe line search and (1.10) is true. If \( \|g_k\| \geq \gamma \) for all \( k > 0 \), we have \( d_k \neq 0 \), we can denote \( \mu_k = \frac{\mu_1}{\|g_k\|^2} \), then

\[ \sum_{k \geq 2} \mu_k \|g_{k-2}\|^2 < \infty \]  

(3.7)

We can deduce that (1.10) cannot show the convergence of the sequence \( \mu_k \), but it implies that the vector \( \mu_k \) changes slowly.

**Lemma 3.4** [7] Suppose that the Assumption 2.3 holds. Consider any iteration method of the form (1.2) and (1.3), where \( \beta_k \geq 0 \), \( \alpha_k \) is obtained by the Wolfe line search and (1.10) is true. If \( \|g_k\| \geq \gamma \) and \( \beta_k \) satisfies Property (*), then there exists \( \lambda > 0 \), for all \( \Delta \in \mathbb{Z}^+, k \geq k_0 \) such that

\[ \left| K_{k,\Delta} \right| \geq \frac{\Delta}{2} \]  

(3.8)

where \( K_{k,\Delta} = \{i \in \mathbb{Z}^+: k \leq i \leq k + \Delta - 1, \|s_{i-1}\| > \lambda\} \), \( K_{k,\Delta} \) means the number of \( K_{k,\Delta} \), \( k_0 \) is an index.
When the method has Property(*), this lemma can show that if the gradients are bounded and away from zero, then a fraction of the steps can not be too small. Otherwise we can prove that \( \|d_k\| \) increases linearly at most.

Finally we can get the following theorem.

**Theorem 3.5** Suppose that \( \{x_i\} \) is generated by Algorithm 2.2, where \( \alpha_k \) is obtained by the Wolfe line research (1.8) and (1.9). Assume that the Assumption 2.3 holds. Then we have

\[
\lim_{k \to \infty} \inf \|g_k\| = 0
\]  

(3.9)

**Proof** Assume that (3.9) is not true, then there exists a constant \( m > 0 \) such that

\[
\|g_k\| \geq m > 0
\]  

(3.10)

From the above relation and Lemma 3.2, we can know that Lemma 3.3 and Lemma 3.4 hold. Let

\[
\mu_k = \frac{d_k}{\|d_k\|},
\]

then for all \( l, k \in \mathbb{Z}^+ \) and \( l \geq k \), we have

\[
x_i - x_{k+1} = \sum_{i=k}^{l} \|s_{i-1}\| \mu_{i-1} = \sum_{i=k}^{l} \|s_{i-1}\| \mu_{i-1} + \sum_{i=k}^{l} \|s_{i-1}\| (\mu_{i-1} - \mu_{k-1})
\]  

(3.11)

From (2.6), taking norms of the above equation, we can get that

\[
\sum_{i=k}^{l} \|s_{i-1}\| \leq 2\xi + \sum_{i=k}^{l} \|s_{i-1}\| \|\mu_{i-1} - \mu_{k-1}\|
\]  

(3.12)

Let \( \lambda \) be given by Lemma 3.4, \( \Delta = \left[ \frac{8\xi}{\lambda} \right] \) which means that \( \Delta \) is an integer and no less than \( \frac{8\xi}{\lambda} \).

Then from Lemma 3.4, there exists an index \( k_0 \), such that

\[
\sum_{i=k_0}^{l} \|\mu_{i+1} - \mu_i\|^2 \leq \frac{1}{4\Delta}
\]  

(3.13)

But from Lemma 3.4, there exists an index \( k_0 \), such that

\[
|K_{k_0, \Delta}| > \frac{\Delta}{2}
\]  

(3.14)

From (3.13) and Cauchy-Schwarz inequality, we have

\[
\|\mu_{i-1} - \mu_{k-1}\| \leq \sum_{j=k}^{i-1} \|\mu_j - \mu_{j-1}\| \leq (i-k) \frac{1}{2} \left( \sum_{j=k}^{i-1} \|\mu_j - \mu_{j-1}\|^2 \right)^{\frac{1}{2}} \leq \Delta \frac{1}{2} \left( \frac{1}{4\Delta} \right)^{\frac{1}{2}} = \frac{1}{2}
\]  

(3.15)

From above relation (2.6) and (3.14), we get that

\[
2\xi \geq \frac{1}{2} \sum_{j=k}^{i-1} \|s_{j-1}\| > \frac{\lambda}{2} |K_{k_0, \Delta}| > \frac{\lambda \Delta}{4}
\]  

(3.16)

So \( \Delta < \frac{8\xi}{\lambda} \). By the definition of \( \Delta \), we can obtain a contradiction. Therefore (3.9) is true.
4. Numerical experiments

In this section, we will test PRP, PRP⁺ and MPRP conjugate methods with Wolfe line search. For each method, ε = 10⁻⁶, δ = 0.01, σ = 0.1, μ₁ = 1.0, μ₂ = 2.4 and μ₃ = 1.0. The problem we test are from [8].

In order to rank the iterative numerical methods, one can compute the total number of function and gradient evaluation by the formula  \( N_{total} = NF + 5 \times NG \) (4.1).

Table 1 shows the computation results, where the columns of the table have the following meanings: Problem (the name of the test problem), Dim (the dimension of the problem), NI (the total number of iterations), NF (the number of the function evaluations), NG (the number of the gradient evaluations) and the star * denotes that this result is the best among these three methods.

Table 1. Test results for the PRP method/PRP⁺ method/MPRP method

<table>
<thead>
<tr>
<th>Problem</th>
<th>Dim</th>
<th>PRP NI/NF/NG</th>
<th>PRP⁺ NI/NF/NG</th>
<th>MPRP NI/NF/NG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Helical valley function</td>
<td>3</td>
<td>72/235/86</td>
<td>49/1577/60</td>
<td>47/161/61*</td>
</tr>
<tr>
<td>Biggs EXP6 function</td>
<td>6</td>
<td>137/278/199</td>
<td>108/217/150</td>
<td>105/206/137*</td>
</tr>
<tr>
<td>Gaussian function</td>
<td>3</td>
<td>4/7/5*</td>
<td>4/7/5*</td>
<td>4/7/5*</td>
</tr>
<tr>
<td>Powell badly scaled</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Box three-dimensional function</td>
<td>3</td>
<td>19/44/31</td>
<td>19/44/31</td>
<td>9/26/18*</td>
</tr>
<tr>
<td>Variably dimensional function</td>
<td>10</td>
<td>6/21/9</td>
<td>6/21/8*</td>
<td>6/22/8*</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>6/21/9</td>
<td>6/21/8*</td>
<td>6/22/8*</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>6/21/9</td>
<td>6/21/8*</td>
<td>6/22/8*</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>6/21/9</td>
<td>6/21/8*</td>
<td>6/22/8*</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>6/21/9</td>
<td>6/21/8*</td>
<td>6/22/8*</td>
</tr>
<tr>
<td>Watson function</td>
<td>31</td>
<td>2409/10002/2918</td>
<td>22947/10000/2972</td>
<td>2613/10001/2625*</td>
</tr>
<tr>
<td>Penalty I function</td>
<td>50</td>
<td>105/285/217</td>
<td>29/79/56*</td>
<td>47/111/90</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>35/91/65</td>
<td>89/210/165</td>
<td>41/113/87</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>34/108/65</td>
<td>28/86/53*</td>
<td>41/115/77</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>34/108/65</td>
<td>329/421/362</td>
<td>31/85/53</td>
</tr>
<tr>
<td>Penalty II function</td>
<td>10</td>
<td>3744/7742/3952</td>
<td>4994/10002/5000</td>
<td>525/1429/819*</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>1118/3364/1193</td>
<td>407/1216/480</td>
<td>314/953/374*</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>90/276/121</td>
<td>90/276/121</td>
<td>72/211/95</td>
</tr>
<tr>
<td>Brown badly scaled</td>
<td>2</td>
<td>18/107/26</td>
<td>11/58/17*</td>
<td>44/245/54</td>
</tr>
<tr>
<td>Brown and Dennis function</td>
<td>4</td>
<td>38/191/62</td>
<td>37/167/50*</td>
<td>64/293/89</td>
</tr>
<tr>
<td>Gulf research and development function</td>
<td>3</td>
<td>-</td>
<td>9/53/22*</td>
<td>27/118/59</td>
</tr>
<tr>
<td>Trigonometric function</td>
<td>20</td>
<td>68/104/95</td>
<td>73/114/102</td>
<td>65/104/102*</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>59/93/85</td>
<td>54/85/78*</td>
<td>46/68/68*</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>64/107/97</td>
<td>61/101/93</td>
<td>53/87/87*</td>
</tr>
<tr>
<td>Extended Rosenbrock function</td>
<td>2</td>
<td>31/124/58</td>
<td>31/116/55*</td>
<td>37/132/61</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---</td>
<td>-----------</td>
<td>------------</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>32/126/59</td>
<td>31/116/55*</td>
<td>37/132/61</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>29/118/56</td>
<td>31/116/55*</td>
<td>37/132/61</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>31/124/58</td>
<td>33/121/57*</td>
<td>42/146/66</td>
</tr>
<tr>
<td>Extended Powell singular function</td>
<td>40</td>
<td>116/323/133</td>
<td>94/277/118*</td>
<td>156/433/165</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>81/227/95</td>
<td>103/295/122*</td>
<td>193/530/205</td>
</tr>
<tr>
<td>Beale function</td>
<td>2</td>
<td>19/42/22</td>
<td>18/42/21*</td>
<td>20/48/23</td>
</tr>
<tr>
<td>Wood function</td>
<td>4</td>
<td>169/600/234</td>
<td>79/273/103</td>
<td>74/266/103*</td>
</tr>
<tr>
<td>Chebyquad function</td>
<td>50</td>
<td>420/1351/467</td>
<td>420/1351/467</td>
<td>403/1294/450*</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>2/22/1*</td>
<td>2/22/1*</td>
<td>2/22/1*</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>2/22/1*</td>
<td>2/22/1*</td>
<td>2/22/1*</td>
</tr>
<tr>
<td>Brown almost-linear function</td>
<td>20</td>
<td>13/58/31</td>
<td>7/42/24*</td>
<td>7/43/24</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>7/50/27</td>
<td>7/50/26*</td>
<td>7/50/26*</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>7/66/35</td>
<td>6/58/32</td>
<td>6/58/29*</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>6/70/27</td>
<td>5/61/33*</td>
<td>5/61/33*</td>
</tr>
<tr>
<td>Linear-full rank function</td>
<td>100</td>
<td>4/26/8*</td>
<td>4/26/8*</td>
<td>5/28/7</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>4/26/7*</td>
<td>4/26/7*</td>
<td>5/28/8</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>7/26/8*</td>
<td>7/62/8*</td>
<td>8/72/10</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>4/34/15*</td>
<td>4/34/15*</td>
<td>4/34/15*</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>6/58/27*</td>
<td>6/58/27*</td>
<td>6/48/19*</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>5/50/7*</td>
<td>5/51/7</td>
<td>5/51/7</td>
</tr>
<tr>
<td>Broden tridiagonal function</td>
<td>10</td>
<td>27/84/30</td>
<td>29/90/32</td>
<td>26/78/26*</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>33/103/37</td>
<td>33/103/37</td>
<td>30/90/30*</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>51/159/58</td>
<td>51/159/58</td>
<td>52/159/57*</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>65/197/69</td>
<td>63/192/67</td>
<td>58/176/62*</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>66/206/77</td>
<td>65/203/76*</td>
<td>93/284/102</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>82/263/100</td>
<td>87/283/108</td>
<td>63/202/78*</td>
</tr>
<tr>
<td>Broden banded function</td>
<td>10</td>
<td>151/516/190</td>
<td>234/741/272</td>
<td>20/75/34*</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>142/451/170</td>
<td>142/451/170</td>
<td>21/75/31*</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>42/156/67</td>
<td>42/143/57</td>
<td>29/103/43*</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>38/136/60</td>
<td>24/75/28</td>
<td>24/74/27*</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>40/153/63</td>
<td>24/74/27*</td>
<td>24/74/27*</td>
</tr>
<tr>
<td></td>
<td>10000</td>
<td>42/157/68</td>
<td>24/124/27</td>
<td>24/74/27*</td>
</tr>
<tr>
<td>Discrete boundary value function</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>775/2354/776*</td>
</tr>
</tbody>
</table>
Similarly, we compare PRP method, MPRP method with \( PRP^+ \) method as follows: for each problem \( i \), compute the total numbers of function evaluations and gradient evaluations required by the evaluated methods (EM) and \( PRP^+ \) method by formula (4.1), and denote them by \( N_{\text{total},j}(EM) \) (EM donates PRP, \( PRP^+ \) or MPRP) and \( N_{\text{total},j}(PRP^+) \); then calculate the ratio

\[
\gamma_i(EM(j)) = \frac{N_{\text{total},j}(EM(j))}{N_{\text{total},j}(PRP^+)} \tag{4.2}
\]

If \( \gamma_i > 1 \), then \( PRP^+ \) method is better than EM method; if \( \gamma_i = 1 \), \( PRP^+ \) method is as well as EM method; if \( \gamma_i < 1 \), \( PRP^+ \) method is worse than EM method. Finally we use the geometric mean of these ratios for \( EM(j) \) method over all the test problems which is defined by

\[
\gamma(EM(j)) = \left( \prod_{i=1}^{\text{\#S}} \gamma_i(EM(j)) \right)^{1/\text{\#S}} \tag{4.3}
\]

where \( S \) denotes the set of the test problems and \( \text{\#S} \) the number of elements in \( S \).

According to the above rule, it’s clear that \( \gamma(PRP^+) = 1 \). The values of \( \gamma(PRP) \) and \( \gamma(MPRP) \) are listed in Table 2.

<table>
<thead>
<tr>
<th>( PRP )</th>
<th>( PRP^+ )</th>
<th>MPRP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.075300</td>
<td>1.0</td>
<td>0.911050</td>
</tr>
</tbody>
</table>

From Table 2, we can see that the average performances of the MPRP method is the best among the three conjugate gradient methods, and the average performances of the \( PRP^+ \) method is a little better than the PRP method. Furthermore, MPRP method is globally convergence for the general nonconvex unconstrained optimization problem, whereas the PRP method with exact line searches may not converge.

Acknowledgements

The authors gratefully acknowledge the support from the National Natural Science Foundations of China (No. 61070165) and R. & D. of Ministry of Education and Guangdong (2011B090400458).

References


