



**Laboratoire d'Arithmétique,
Calcul formel et d'Optimisation**

UMR CNRS 6090

Laco

**Modification of the Wolfe line search rules to satisfy
the descent condition in the Polak-Ribière-Polyak
conjugate gradient method**

Paul Armand

**Rapport de recherche n° 2005-04
Déposé le 29 septembre 2005
Version révisée et augmentée du rapport LACO 2003-10**

Modification of the Wolfe line search rules to satisfy the descent condition in the Polak-Ribière-Polyak conjugate gradient method

Paul ARMAND[†]

April 7, 2005 (revised version)

Abstract. The Polak-Ribière-Polyak (PRP) conjugate gradient algorithm is a useful tool of unconstrained numerical optimization. Efficient implementations of the algorithm usually perform line searches satisfying the strong Wolfe conditions. It is well known that these conditions do not guarantee that the successive computed directions are descent directions. This paper proposes a line search technique that guarantees the realization of a relaxed form of the strong Wolfe conditions and also guarantees the descent condition at each iteration. It is proved that this line search algorithm preserves the usual convergence properties of any descent algorithm. In particular, it is shown that the Zoutendijk condition holds under mild assumptions. It is also proved that the resulting PRP algorithm is convergent under a strong convexity assumption. For nonlinear nonconvex function, we propose a modification of the PRP algorithm and prove its global convergence. Numerical tests are presented to validate the different approaches.

Key words. unconstrained optimization, conjugate gradient method, line search algorithm

AMS subject classification. 90C06, 90C30, 65K05

1 Introduction

The Polak-Ribière-Polyak [23, 24] (PRP) conjugate gradient algorithm is an iterative method for computing an unconstrained minimizer of a continuously differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$. The method is particularly useful for solving problems when n is large, because it requires very few memory storage. The algorithm generates a sequence of iterates according to

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \dots \quad (1.1)$$

where $d_k \in \mathbb{R}^n$ is the line search direction and $\alpha_k > 0$ is a step length. The direction is defined by

$$d_k = \begin{cases} -g_k & \text{for } k = 0, \\ -g_k + \beta_k d_{k-1} & \text{for } k \geq 1, \end{cases} \quad (1.2)$$

where g_k denotes the gradient $\nabla f(x_k)$ and β_k is a scalar defined by

$$\beta_k = \frac{(g_k - g_{k-1})^\top g_k}{\|g_{k-1}\|^2}. \quad (1.3)$$

[†]LACO-CNRS, Université de Limoges, Faculté des Sciences, 123, avenue Albert Thomas, 87060 Limoges (France); e-mail: armand@unilim.fr.

The step length is computed by performing a line search along d_k . In practice, a relevant choice is to compute α_k according to the realization of the strong Wolfe conditions, namely

$$f(x_{k+1}) \leq f(x_k) + \omega_1 \alpha_k g_k^\top d_k \quad (1.4)$$

and

$$|g_{k+1}^\top d_k| \leq -\omega_2 g_k^\top d_k. \quad (1.5)$$

where $0 < \omega_1 < \omega_2 < 1$, see [7, 22].

The PRP method belongs to the class of descent algorithms. Each new direction d_{k+1} must satisfy the descent condition

$$g_{k+1}^\top d_{k+1} < 0. \quad (1.6)$$

However, and contrary to other formulas of β_k , such as Fletcher-Reeves [8], conjugate descent [7] or Dai-Yuan [5], the realization of the strong Wolfe conditions does not guarantee that (1.6) is satisfied at each iteration. The following example is given in [4] (see [18] for a similar example). Consider $f(x) = ax^2$, where $a = \min\{(1+\omega_2)/2, 1-\omega_1\}$, with $0 < \omega_2 \leq 1/2$. From the starting point $x_0 = 1$, the step length $\alpha_0 = 1$ satisfies (1.4) and (1.5), but $g_1^\top d_1 = 4a^2(2a-1)^3 > 0$. In particular, this example shows that the algorithm defined by (1.1)–(1.5), does not converge, even if the function is strongly convex. Note that the same behavior occurs when formula (1.3) is replaced by the Hestenes-Stiefel [16] formula.

The lack of the descent property has some drawbacks. From a theoretical point of view, the convergence analyses related to the PRP algorithm with a Wolfe line search have been done with the additional assumption that the descent condition is satisfied at each iteration, see [3, 6, 12]. A purpose of the present paper is to show that the realization of a relaxed form of the Wolfe line search conditions, leads to guarantee the descent property at each iteration, while maintaining the known convergence properties. From a practical point of view, several line search algorithms have been proposed to satisfy the Wolfe conditions [1, 7, 17, 20, 21, 26, 27], but, and as emphasized by Gilbert and Nocedal in [12, p. 39], none of them is guaranteed to satisfy the strong Wolfe conditions and also provide a descent direction for the PRP method. To force the realization of (1.6) in the numerical tests, the strategy proposed in [12] is the following. At first, a step length that satisfies (1.4) and (1.5) is computed by means of the Moré-Thuente [21] algorithm, then the line search iterations are pursued until (1.6) is also satisfied. The code CG+, developed by Liu, Nocedal and Waltz [19], uses this technique. Most the time, this technique succeeds in satisfying the conditions (1.4)–(1.6) at each iteration, except for a problem of the CUTER collection [13], for which we observed a failure due to the non realization of the descent condition. We will come back to this example at the end of the paper. More recently, Grippo and Lucidi [15] formalized the line search technique suggested in [12] and proposed an implementable trust-region version of the PRP algorithm and proved its global convergence.

These considerations motivated us to propose a modification of the strong Wolfe conditions to satisfy the descent condition in the PRP conjugate gradient algorithm, while preserving the convergence properties and the numerical efficiency. Our line search technique, described in Section 2, is based on a relaxation of (1.4) whenever it is necessary during the line search procedure, allowing the trial steps to go in a neighborhood of a minimizer of the line search function, where both conditions (1.5) and (1.6) are satisfied. In fact, this technique has its roots in previous works on line searches for quasi-Newton methods in the framework of SQP algorithms, see [2, 9, 10, 11]. The general idea is to use the sufficient decrease condition (1.4) as a guide to produce trial steps that converge to a stationary point of the line search function, without supposing the existence of a step satisfying all the line search conditions.

In Section 3, we will show that any descent algorithm, equipped of this line search, satisfies the Zoutendijk condition. Then we prove that the resulting PRP algorithm is globally convergent when the function f is strongly convex. We then propose a modified version of the PRP iteration and prove its global convergence in the nonlinear nonconvex case. Next we present some numerical experiments in Section 4, to show the feasibility and the efficiency of this new line search technique.

2 The line search algorithm

Using (1.2) and (1.6), the direction d_{k+1} is a descent direction provided that

$$\beta_{k+1} g_{k+1}^\top d_k < \|g_{k+1}\|^2.$$

To find a step length that satisfies this inequality, a possible strategy would be to reduce enough the size of the step. Since the gradient is supposed to be continuous, β_{k+1} goes to zero with α_k . Such an approach has been used by Grippo and Lucidi [14, 15]. They proposed line search conditions that accept short steps and proved a strong convergence result for the resulting PRP algorithm. From a practical point of view, the computation of short steps is not really efficient. Moreover, this strategy can not be adopted for the realization of the Wolfe conditions, because (1.5) is not satisfied for small steps. A second strategy would be to choose a step length near a local minimizer of the line search function (note that in this case a global convergence result cannot be expected because of the counterexample of Powell [25]). In that case the scalar product $g_{k+1}^\top d_k$ goes to zero, and unless g_{k+1} goes also to zero, the descent condition can be satisfied. Note that (1.5) is also satisfied in a neighborhood of a minimum, but this is not necessarily the case for condition (1.4). To solve this kind of conflict, we propose to relax the decrease condition (1.4) in the following way. Starting from an initial trial step, we first use a backtracking technique until (1.4) is satisfied. Let $\alpha_{k,1}$ the corresponding step. Suppose that either (1.5) or (1.6) is not satisfied at the new point, say $x_{k,1} = x_k + \alpha_{k,1} d_k$. If $g_{k,1}^\top d_k > 0$, then a minimum of the line search function $\alpha \rightarrow f(x_k + \alpha d_k)$ has been bracketed. In this case it is easy to find a step near this minimum while satisfying the decrease

condition, see for example the sectioning phase described in [7, Section 2.6] or the modified updating algorithm described in [21, Section 3]. We adopt the same strategy here. Suppose now that $g_{k,1}^\top d_k < 0$. In this case, the decrease condition is reformulated as follows:

$$f(x_k + \alpha d_k) \leq f(x_k) + \alpha_{k,1} \omega_1 g_k^\top d_k + (\alpha - \alpha_{k,1}) \omega_1 \max\{g_k^\top d_k, g_{k,1}^\top d_k\}.$$

From $\alpha_{k,1}$, a forward (or extrapolation) step is computed and it is followed by a sequence of backtracking steps until a new step, say $\alpha_{k,2}$, satisfies the above condition. If $\alpha_{k,2}$ satisfies the other line search conditions, then it is an acceptable step, otherwise the decrease condition is reformulated as previously and so on. By repeating this procedure, it is only asked to the function value $f(x_{k+1})$, to be not greater than the value of a decreasing and convex piecewise linear function. We will show that it is sufficient to obtain global convergence properties for the descent algorithm. Note that this algorithmic scheme operates without assuming the existence of a step length which satisfies all conditions (1.4) and (1.5)–(1.6). In the sequel, the latter conditions will be replaced by only one set of conditions, called the stopping criterion of the line search. The only assumption is that the stopping criterion is satisfied at a stationary point of the line search function.

To give a description of the line search algorithm, we denote by ϕ the line search function. With the previous notation we would have $\phi(\alpha) = f(x_k + \alpha d_k)$.

Assumptions 2.1 (i) The function $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is continuously differentiable on $[0, \infty)$ and $\phi'(0) < 0$. (ii) If $\phi'(\alpha) = 0$, then the step length α satisfies the line search stopping criterion.

The algorithm generates a sequence $\{\alpha_i\}$ of steps in the following way. While $\phi'(\alpha_i) < 0$ the algorithm stays in Phase I. From α_i , a first extrapolation trial step is computed and is followed by a sequence of backtracking trial steps, until a sufficient decrease of ϕ is obtained. This defines the next step α_{i+1} . As soon as a new step satisfies $\phi'(\alpha_{i+1}) > 0$, this means that a minimizer has been bracketed in $[\alpha_i, \alpha_{i+1}]$, then the algorithm enters in Phase II. In this second phase, the algorithm generates a sequence of intervals of the form $[\alpha_i, \alpha_{i+1}]$ or $[\alpha_{i+1}, \alpha_i]$ that contain the minimizer and whose lengths tend to zero. It is assumed that the following constants are given: $\omega_1 \in (0, 1)$, $0 < \tau_E < \tau'_E$ and $\tau_1 \in (0, \frac{1}{2})$.

LINE SEARCH ALGORITHM

PHASE I

1. Set $i = 0$, $\alpha_0 = 0$ and $b_0 = \infty$.
2. Find a step length such that a sufficient decrease of ϕ is obtained:
 - 2.1. Set $j = 0$, $b_{i,0} = b_i$ and $s_i = \max\{\phi'(\alpha_l) : 0 \leq l \leq i\}$.

- 2.2. If $b_i = \infty$, choose $\tau_{i,0} \in [\tau_E, \tau'_E]$ and set $\alpha_{i,1} = (1 + \tau_{i,0})\alpha_i$,
else choose $\tau_{i,0} \in [\tau_1, (1 - \tau_1)]$ and set $\alpha_{i,1} = \alpha_i + \tau_{i,0}(b_i - \alpha_i)$.
Go to Step 2.4.
- 2.3. Choose $\tau_{i,j} \in [\tau_1, (1 - \tau_1)]$ and set $\alpha_{i,j+1} = \alpha_i + \tau_{i,j}(\alpha_{i,j} - \alpha_i)$.
- 2.4. If $\phi(\alpha_{i,j+1}) \leq \phi(0) + \omega_1 \sum_{l=0}^{j-1} (\alpha_{l+1} - \alpha_l)s_l + \omega_1(\alpha_{i,j+1} - \alpha_i)s_i$,
go to Step 3.
- 2.5. If $\phi'(\alpha_{i,j+1}) > 0$, set $b_{i,j+1} = \alpha_{i,j+1}$, else set $b_{i,j+1} = b_{i,j}$.
- 2.6. Set $j = j + 1$ and go to Step 2.3.
3. Set $\alpha_{i+1} = \alpha_{i,j+1}$. If the stopping criterion holds, stop.
4. If $\phi'(\alpha_{i+1}) < 0$, set $b_{i+1} = b_{i,j}$, $i = i + 1$ and go to Step 2,
else set $b_{i+1} = \alpha_i$, $i = i + 1$ and go to Step 5.

PHASE II

5. Find a step length such that a decrease of ϕ is obtained:
 - 5.1. Set $j = 0$ and $b_{i,0} = b_i$.
 - 5.2. Choose $\tau_{i,0} \in [\tau_1, (1 - \tau_1)]$ and set $\alpha_{i,1} = \alpha_i + \tau_{i,0}(b_i - \alpha_i)$.
Go to Step 5.4.
 - 5.3. Choose $\tau_{i,j} \in [\tau_1, (1 - \tau_1)]$ and set $\alpha_{i,j+1} = \alpha_i + \tau_{i,j}(\alpha_{i,j} - \alpha_i)$.
 - 5.4. If $\phi(\alpha_{i,j+1}) \leq \phi(\alpha_i)$, go to Step 6.
 - 5.5. Set $b_{i,j+1} = \alpha_{i,j+1}$.
 - 5.6. Set $j = j + 1$ and go to Step 5.3.
6. Set $\alpha_{i+1} = \alpha_{i,j+1}$. If the stopping criterion holds, stop.
7. If $\phi'(\alpha_{i+1})(\alpha_{i+1} - \alpha_i) < 0$, set $b_{i+1} = b_{i,j}$, else set $b_{i+1} = \alpha_i$.
8. Set $i = i + 1$ and go to Step 5.

The following lemma gives some elementary properties of the line search algorithm.

Lemma 2.2 *Suppose that Assumptions 2.1 hold. For any index $i \geq 0$, if $\phi'(\alpha_i) \neq 0$, then the following properties hold:*

- (i) $\phi'(\alpha_i)(b_i - \alpha_i) < 0$.
- (ii) If $b_i < \infty$, then $\phi(\alpha_i) < \phi(b_i)$ or $\phi'(\alpha_i)\phi'(b_i) < 0$.
- (iii) If $b_i < \infty$, then the interval bounded by α_i and b_i contains at least one local minimizer of ϕ .
- (iv) For sufficiently large j , the decrease conditions at Step 2.4 and Step 5.4 are satisfied. In particular, there exists $j_i \geq 0$ such that

$$\begin{cases} \alpha_i < \alpha_{i+1} = \alpha_{i,j_i+1} < \cdots < \alpha_{i,1} < b_i & \text{if } \phi'(\alpha_i) < 0, \\ b_i < \alpha_{i,1} < \cdots < \alpha_{i,j_i+1} = \alpha_{i+1} < \alpha_i & \text{otherwise.} \end{cases}$$

- (v) If the algorithm entered in Phase II, then $|\alpha_{i+1} - b_{i+1}| \leq (1 - \tau_1)|\alpha_i - b_i|$.

Proof. To prove (i), consider first that during Phase I one has $\phi'(\alpha_i) < 0$ (Step 4) and $\alpha_i < b_i$ (Step 2). During Phase II, the first time the algorithm

goes to Step 5, one has $\phi'(\alpha_i) > 0$ and $b_i = \alpha_{i-1} < \alpha_i$ (Step 4). For the next iterations, the property follows that at the end of Step 7 one has either $\phi'(\alpha_{i+1})(\alpha_{i+1} - \alpha_i) > 0$ and $b_{i+1} = \alpha_i$, or $\phi'(\alpha_{i+1})(\alpha_{i+1} - \alpha_i) < 0$ and $b_{i+1} = b_{i,j} < \alpha_{i+1} < \alpha_i$ or $\alpha_i < \alpha_{i+1} < b_{i,j} = b_{i+1}$.

To prove property (ii), suppose that $b_i < \infty$. During Phase I one has $\phi'(\alpha_i) < 0$ (Step 4). The value of b_i depends on steps 2.1, 2.5 and 4. As long as $\phi'(\alpha_i) < 0$ one has $\phi'(b_i) > 0$. Consider now the sequence of values generated during Phase II. The first time the algorithm goes to Step 5, one has $\phi'(\alpha_i) > 0$ and $\phi'(b_i) < 0$ (Step 4). For the next iterations, suppose that Property (ii) is satisfied at rank i . By Assumption 2.1 (ii), one has $\phi'(\alpha_{i+1}) \neq 0$. Suppose at first that $\phi'(\alpha_{i+1})(\alpha_{i+1} - \alpha_i) > 0$. One has $b_{i+1} = \alpha_i$ (Step 7) and $\phi'(\alpha_i)(b_i - \alpha_i) < 0$ (Property (i)). But $(\alpha_{i+1} - \alpha_i)$ and $(b_i - \alpha_i)$ are of the same sign, it follows that $\phi'(\alpha_{i+1})\phi'(b_{i+1}) < 0$. Suppose now that $\phi'(\alpha_{i+1})(\alpha_{i+1} - \alpha_i) < 0$. One has either $b_{i+1} = \alpha_{i,j}$, for some $j \geq 1$ or $b_{i+1} = b_i$. In the former case, one has $\phi(\alpha_{i,j}) > \phi(\alpha_i)$ and since $\phi(\alpha_i) \geq \phi(\alpha_{i+1})$, we obtain $\phi(b_{i+1}) > \phi(\alpha_{i+1})$. In the latter case, from the induction hypothesis one has either $\phi(b_{i+1}) = \phi(b_i) > \phi(\alpha_i) \geq \phi(\alpha_{i+1})$ or $\phi'(\alpha_i)\phi'(b_{i+1}) < 0$. In this last case, since $\phi'(\alpha_i)(b_i - \alpha_i) < 0$, $\phi'(\alpha_{i+1})(\alpha_{i+1} - \alpha_i) < 0$ and $(b_i - \alpha_i)$ is of the same sign that $(\alpha_{i+1} - \alpha_i)$, we obtain $\phi'(\alpha_i)\phi'(\alpha_{i+1}) > 0$, and thus $\phi'(\alpha_{i+1})\phi'(b_{i+1}) < 0$.

Property (iii) is a straightforward consequence of the first two properties.

The choices of $\alpha_{i,j}$ (Steps 2.3 and 5.3), imply that $|\alpha_{i,j+1} - \alpha_i| \leq (1 - \tau_1)^j |\alpha_{i,1} - \alpha_i|$ for $j \geq 1$, and thus $\alpha_{i,j} \rightarrow \alpha_i$ when $j \rightarrow \infty$. Since $\phi'(\alpha_i)(b_i - \alpha_i) < 0$ and $0 < \omega_1 < 1$, the tests at Steps 2.4 and 5.4 are satisfied for sufficiently large j , from which Property (iv) follows.

To prove (v), suppose that the algorithm entered is Phase II. At Step 7, if $b_{i+1} = b_{i,j}$, then from Step 5.5 one has $b_{i,j} = \alpha_{i,j}$ and thus $\alpha_{i+1} = \alpha_i + \tau_{i,j_i}(b_{i+1} - \alpha_i)$. Since b_{i+1} is in the interval bounded by α_i and b_i , one has $|\alpha_{i+1} - b_{i+1}| \leq (1 - \tau_1)|b_{i+1} - \alpha_i| \leq (1 - \tau_1)|b_i - \alpha_i|$. On the other hand, if $b_{i+1} = \alpha_i$, then $|\alpha_{i+1} - b_{i+1}| = |\alpha_{i+1} - \alpha_i| \leq (1 - \tau_1)|b_i - \alpha_i|$. \square

Suppose that α_{i+1} was computed during Phase I. One has $\alpha_i < \alpha_{i+1}$ and

$$\phi(\alpha_{i+1}) \leq \phi(0) + \omega_1 \sum_{j=0}^i (\alpha_{j+1} - \alpha_j) \max\{\phi'(\alpha_l) : 0 \leq l \leq j\}. \quad (2.1)$$

On the other hand, if α_{i+1} was computed during Phase II, one has $\phi(\alpha_{i+1}) \leq \phi(\alpha_i)$.

Proposition 2.3 *Suppose that Assumptions 2.1 hold. If the algorithm does not terminate, then an infinite sequence of step lengths $\{\alpha_i\}$ is built and either the sequence $\{\phi(\alpha_i)\}$ tends to $-\infty$ or $\liminf_{i \rightarrow \infty} \phi'(\alpha_i) = 0$.*

Proof. While the stopping criterion at Step 3 or Step 6 is not satisfied, Assumption 2.1 (ii) implies that $\phi'(\alpha_i) \neq 0$ and Property (iv) of Lemma 2.2 implies that the next step α_{i+1} is well defined.

Suppose at first that $\phi'(\alpha_i) > 0$ for some $i \geq 1$, then the algorithm entered in Phase II. Properties (iii) and (v) of Lemma 2.2 imply that the sequences $\{\alpha_i\}$ and $\{b_i\}$ converge to a common limit point $\bar{\alpha}$ such that $\phi'(\bar{\alpha}) = 0$.

Suppose now that $\phi'(\alpha_i) < 0$ for all $i \geq 1$. We proceed by contradiction, by supposing that ϕ is bounded from below and that there exists $\varepsilon > 0$ such that

$$\phi'(\alpha_i) \leq -\varepsilon, \quad \text{for all } i \geq 0. \quad (2.2)$$

Since the algorithm stays in Phase I, the sequence $\{\alpha_i\}$ is increasing and Property (2.1) implies that

$$\begin{aligned} \phi(\alpha_i) &\leq \phi(0) + \omega_1 \sum_{j=0}^{i-1} (\alpha_{j+1} - \alpha_j) \max\{\phi'(\alpha_l) : 0 \leq l \leq j\}, \\ &\leq \phi(0) - \omega_1 \sum_{j=0}^{i-1} (\alpha_{j+1} - \alpha_j) \varepsilon, \\ &= \phi(0) - \omega_1 \varepsilon \alpha_i. \end{aligned} \quad (2.3)$$

Since ϕ is bounded from below, the sequence $\{\alpha_i\}$ converges to some limit point $\bar{\alpha} > 0$.

The distance between the first trial step $\alpha_{i,1}$ and α_i is bounded away from zero. Indeed, either $b_i = \infty$ for all i and then $\alpha_{i,1} - \alpha_i \geq \tau_E \alpha_i$, or $b_i < \infty$ for some iteration i . In the first case, one has $\alpha_{i,1} - \alpha_i \geq \frac{\tau_E}{2} \bar{\alpha}$ for large i . In the latter case, the choice of $\alpha_{i,1}$ at Step 2.2 implies that $\alpha_{i,1} - \alpha_i \geq \tau_I (b_i - \alpha_i)$. But we know from Lemma 2.2 (iii) that the interval $[\alpha_i, b_i]$ contains a local minimizer of ϕ , therefore $(b_i - \alpha_i)$ is bounded away from zero because of (2.2). Since $(\alpha_{i+1} - \alpha_i) \rightarrow 0$ when $i \rightarrow \infty$, for sufficiently large i the decrease condition at Step 2.4 is not satisfied at $\alpha_{i,1}$. It follows that the sequence $\{\alpha_{i,j_i}\}$ is such that $\alpha_{i+1} = \alpha_i + \tau_{i,j_i} (\alpha_{i,j_i} - \alpha_i)$ and

$$\begin{aligned} \phi(\alpha_{i,j_i}) &> \phi(0) + \omega_1 \sum_{j=0}^{i-1} (\alpha_{j+1} - \alpha_j) \max\{\phi'(\alpha_l) : 0 \leq l \leq j\} \\ &\quad + \omega_1 (\alpha_{i,j_i} - \alpha_i) \max\{\phi'(\alpha_l) : 0 \leq l \leq i\}. \end{aligned}$$

Subtracting $\phi(\alpha_i)$ on both sides, using (2.3) and next (2.2), we obtain

$$\frac{\phi(\alpha_{i,j_i}) - \phi(\alpha_i)}{\alpha_{i,j_i} - \alpha_i} - \phi'(\alpha_i) \geq (1 - \omega_1) \varepsilon.$$

By taking limit $i \rightarrow \infty$, we obtain $0 \geq (1 - \omega_1) \varepsilon$ a contradiction with $\omega_1 < 1$. \square

This line search technique allows to compute a step length that satisfies the Wolfe or strong Wolfe conditions. It suffices to use it with one of the following stopping criteria:

$$\phi'(\alpha_i) \geq \omega_2 \phi'(0) \quad (2.4)$$

or

$$|\phi'(\alpha_i)| \leq \omega_2 |\phi'(0)|, \quad (2.5)$$

where $\omega_2 \in (0, 1)$.

Proposition 2.4 *Suppose that Assumption 2.1 (i) holds and that ϕ is bounded from below. If (2.4) (resp. (2.5)) is used as line search stopping criterion, then the line search algorithm terminates at some α_i and $\phi(\alpha_i) \leq \phi(0) + \omega_1 \omega_2 \alpha_i \phi'(0)$.*

Proof. Since the stopping criteria (2.4) and (2.5) are satisfied in a neighborhood of a stationary of ϕ , Assumptions 2.1 are satisfied, and thus Proposition 2.3 implies that the line search terminates after a finite number of iterations.

While (2.4) (resp. (2.5)) is not satisfied, one has $\phi'(\alpha_i) < \omega_2 \phi'(0)$ (resp. $\phi'(\alpha_i) < \omega_2 \phi'(0)$ or $\phi'(\alpha_i) > -\omega_2 \phi'(0) > 0$). Using Property (2.1) and the remarks which go with, we obtain the result. \square

3 Application to the PRP conjugate gradient method

Let x_0 be the starting point of the minimization method. We assume that the following assumption is satisfied.

Assumption 3.1 *The function f is continuously differentiable in a neighborhood \mathcal{N} of the level set $\mathcal{L} := \{x : f(x) \leq f(x_0)\}$ and its gradient is Lipschitzian on \mathcal{N} , that is, there exists $L > 0$ such that $\|g(x) - g(y)\| \leq L\|x - y\|$ for all $(x, y) \in \mathcal{N} \times \mathcal{N}$.*

Let us begin by proving that the Zoutendijk condition is satisfied for any descent algorithm using the preceding line search algorithm, that includes in the line search stopping criterion the following curvature condition:

$$g_{k+1}^\top d_k \geq \omega_2 g_k^\top d_k. \quad (3.1)$$

Proposition 3.2 *Suppose that Assumption 3.1 holds and that f is bounded below on \mathcal{L} . Consider any iteration of the form (1.1), starting at $x_0 \in \mathbb{R}^n$, where d_k is a descent direction (i.e. $g_k^\top d_k < 0$) and α_k is computed with the line search algorithm described in Section 2. Suppose that the line search function is defined by $\alpha \rightarrow f(x_k + \alpha d_k)$ and that the line search stopping criterion includes condition (3.1). If for all iteration k the number of line search iterations is finite, then*

$$\sum_{k \geq 0} (g_k^\top d_k)^2 \|d_k\|^{-2} < \infty. \quad (3.2)$$

Proof. Since for all $k \geq 0$, the number of line search iterations is finite, there exists $w_k \geq 1$ such that

$$g_{k,i}^\top d_k < \omega_2 g_k^\top d_k \quad \text{for all } 1 \leq i < w_k \quad \text{and} \quad g_{k,w_k}^\top d_k \geq \omega_2 g_k^\top d_k. \quad (3.3)$$

In particular, for all $1 \leq i < w_k$ one has $g_{k,i}^\top d_k < 0$ and $0 = \alpha_{k,0} < \alpha_{k,1} < \dots < \alpha_{k,w_k}$. By using the properties (2.1) and (3.3), we deduce that

$$\begin{aligned}
f(x_{k+1}) &\leq f(x_k) + \omega_1 \sum_{i=0}^{w_k-1} (\alpha_{k,i+1} - \alpha_{k,i}) \max\{g_{k,l}^\top d_k : 0 \leq l \leq i\} \\
&\leq f(x_k) + \omega_1 \sum_{i=0}^{w_k-1} (\alpha_{k,i+1} - \alpha_{k,i}) \omega_2 g_k^\top d_k \\
&\leq f(x_k) + \omega_1 \omega_2 \alpha_{k,w_k} g_k^\top d_k.
\end{aligned} \tag{3.4}$$

Using the second inequality of (3.3) and the Lipschitz continuity of g , one has

$$(1 - \omega_2) |g_k^\top d_k| \leq (g_{k,w_k} - g_k)^\top d_k \leq L \alpha_{k,w_k} \|d_k\|^2.$$

Combining this last inequality with (3.4) we obtain

$$L^{-1} \omega_1 \omega_2 (1 - \omega_2) (g_k^\top d_k)^2 \|d_k\|^{-2} \leq f(x_k) - f(x_{k+1}).$$

Since f is bounded below, we obtain (3.2). \square

We consider now the following PRP conjugate gradient algorithm. Starting from x_0 , a sequence $\{x_k\}$ is generated according to (1.1)–(1.3), the step length α_k being computed with the line search algorithm described in Section 2, where the line search function is defined by $\alpha \rightarrow f(x_k + \alpha d_k)$ and where the line search stopping criterion is composed of both conditions (1.5) and (1.6). At each iteration k , the line search algorithm generates intermediate step lengths $\alpha_{k,i}$ satisfying the decrease conditions at Steps 2.4 or 5.4. We denote by $x_{k,i} = x_k + \alpha_{k,i} d_k$ the corresponding intermediate iterates and we use the notation $g_{k,i} = g(x_{k,i})$.

Proposition 3.3 *Suppose that the function f is continuously differentiable. If at some iteration k of the PRP algorithm, the line search does not terminate, then either f is unbounded below or $\liminf_{i \rightarrow \infty} \|g_{k,i}\| = 0$.*

Proof. Suppose that the line search does not terminate at iteration k . For all index $i \geq 1$ one has

$$|g_{k,i}^\top d_k| > -\omega_2 g_k^\top d_k \quad \text{or} \quad \frac{(g_{k,i} - g_k)^\top g_{k,i}}{\|g_k\|^2} g_{k,i}^\top d_k \geq \|g_{k,i}\|^2.$$

Suppose that f is bounded below. By Proposition 2.3, there exists a subset I of indices such that $g_{k,i}^\top d_k \rightarrow 0$ when $i \rightarrow \infty$ in I . Since $g_k^\top d_k < 0$, the first inequality is satisfied for large i . Using the Cauchy-Schwarz inequality, the second inequality implies that $|g_{k,i}^\top d_k| \geq \|g_k\|^2 \|g_{k,i}\| / (\|g_{k,i}\| + \|g_k\|)$. By taking the limit $i \rightarrow \infty$ in I we deduce that $\|g_{k,i}\| \rightarrow 0$. \square

This result suggests a point of comment. It shows that, to be robust and complete, the PRP method needs to check termination of the overall algorithm

during the line search procedure, otherwise we could observe a failure in satisfying the descent condition while the iterates would go to a stationary point. A similar observation has been done in [15] and we will come back to this point in the numerical tests.

For nonlinear functions, global convergence of PRP algorithm with our line search strategy cannot be expected because of the counterexample of Powell [25]. Nevertheless, we can prove the global convergence with a strong convexity assumption. The function f is said to be strongly convex on \mathcal{L} , if there exists $\kappa > 0$ such that the function $f(\cdot) - \frac{\kappa}{2} \|\cdot\|^2$ is convex on \mathcal{L} . For a differentiable function, the strong convexity is equivalent to the the strong monotonicity of its gradient, namely $\tau\|x - y\|^2 \leq (g(x) - g(y))^\top(x - y)$ for all $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$. Note that a strongly convex function has bounded level sets.

Lemma 3.4 *Suppose that φ is continuously differentiable and strongly convex. If $\{\xi_j\}$ is a sequence in \mathbb{R}^n such that $\{\varphi(\xi_j)\}$ is nonincreasing and a subsequence of $\{\nabla\varphi(\xi_j)\}$ tends to zero, then the whole sequence $\{\xi_j\}$ converges to the unique minimum of φ .*

Proof. The strong convexity of φ implies that its level sets are bounded. Since $\{\varphi(\xi_j)\}$ is nonincreasing, the sequence $\{\xi_j\}$ is bounded. Moreover, φ is bounded from below and thus $\lim_{j \rightarrow \infty} \varphi(\xi_j) = \varphi^*$ exists. The continuity of φ implies that all the convergent subsequences of $\{\xi_j\}$ have the same limit point ξ^* such that $\varphi(\xi^*) = \varphi^*$, therefore $\lim_{j \rightarrow \infty} \xi_j = \xi^*$. By assumption, a subsequence of the gradient values $\nabla\varphi(\xi_j)$ goes to zero, therefore ξ^* is the unique minimum of φ . \square

Theorem 3.5 *Suppose that Assumption 3.1 holds and that f is strongly convex. The whole sequence $\{x_{k,i}\}$ converges to the unique minimum of f .*

Proof. Suppose that at some iteration k , the line search does not terminate. By assertion (i) of Proposition 3.3, there exists a subset I of indices such that $\|g_{k,i}\| \rightarrow 0$ when $i \rightarrow \infty$ in I . Using the strong convexity of f one has $\|g_{k,i}\| \geq \tau\|x_{k,i} - x_*\|$, where x_* is the unique minimum of f . If the line search algorithm stays in Phase I, then the sequence $\{\alpha_{k,i}\}$ is increasing, bounded and thus $x_{k,i} = x_k + \alpha_{k,i}d_k$ converges to x_* . On the other hand, if the line search is entered in Phase II, the values $f(x_{k,i})$ become decreasing and by Lemma 3.4 we obtain $x_{k,i} \rightarrow x_*$ when $i \rightarrow \infty$.

Suppose now that for each iteration k the line search terminates. Proposition 3.2 implies that the Zoutendijk condition (3.2) is satisfied. By the curvature condition (1.5) one has $g_{k+1}^\top d_k \leq -\omega_2 g_k^\top d_k$. Subtracting $g_k^\top d_k$ on both sides and using the strong convexity of f , we obtain

$$\alpha_k \|d_k\|^2 \leq \frac{1 + \omega_2}{\tau} |g_k^\top d_k|. \quad (3.5)$$

Inequality (3.5) and the Zoutendijk condition (3.2) imply that $\alpha_k \|d_k\| \rightarrow 0$. We proceed now by contradiction by supposing that there exists $\epsilon > 0$ such that

$$\|g_k\| \geq \epsilon, \quad \text{for all } k \geq 0. \quad (3.6)$$

The strong convexity of f and the Lipschitz continuity of g imply that there exists a positive constant c such that $\|g(x)\| \leq c$ for all $x \in \mathcal{L}$. From the definition of β_{k+1} and the Lipschitz continuity of g , we deduce that

$$\begin{aligned} |\beta_{k+1}| &\leq \|g_k\|^{-2} \|g_{k+1} - g_k\| \|g_{k+1}\| \\ &\leq Lc\epsilon^{-2} \alpha_k \|d_k\|, \end{aligned}$$

therefore $\beta_k \rightarrow 0$. By (1.2) one has $\|d_k\| \leq c + |\beta_k| \|d_{k-1}\|$, and thus the sequence $\{d_k\}$ is bounded. The Zoutendijk condition (3.2) and the boundedness of $\|d_k\|$ imply that $g_k^\top d_k \rightarrow 0$. Finally, by using the definition of d_{k+1} and the curvature condition (1.5), we obtain

$$\begin{aligned} \|g_{k+1}\|^2 &\leq |g_{k+1}^\top d_{k+1}| + |\beta_{k+1}| |g_{k+1}^\top d_k| \\ &\leq |g_{k+1}^\top d_{k+1}| + \omega_2 |\beta_{k+1}| |g_k^\top d_k|. \end{aligned}$$

It follows that $g_k \rightarrow 0$ when $k \rightarrow \infty$, a contradiction with (3.6), which implies that $\liminf \|g_k\| = 0$. By virtue of Lemma 3.4, the sequence $\{x_k\}$ converges to x_* . The convergence of the whole sequence $\{x_{k,i}\}$ to x_* is a consequence of the following inequalities:

$$\tau \|x_{k,i} - x_*\|^2 \leq f(x_{k,i}) - f(x_*) \leq f(x_k) - f(x_*).$$

The left inequality follows from the strong convexity of f and $\nabla f(x^*) = 0$, the right inequality is a property of the line search. \square

The convergence result given by Polak-Ribière [23], which was obtained for exact line searches, is a consequence of Theorem 3.5.

Corollary 3.6 *Suppose that assumptions of Theorem 3.5 hold. Let α_k^* be the unique minimum of the function $\alpha \rightarrow f(x_k + \alpha d_k)$. If $\omega_1 > 0$ is sufficiently small, then conditions (1.4)-(1.6) are satisfied at $\alpha_k = \alpha_k^*$, in particular, the sequence $\{x_k\}$ defined by $x_{k+1} = x_k + \alpha_k^* d_k$, converges to the unique minimum of f .*

Proof. By definition of α_k^* one has $g(x_k + \alpha_k^* d_k)^\top d_k = 0$. Therefore, both conditions (1.5) and (1.6) are satisfied at α_k^* . It suffices to verify that the decrease condition (1.4) is satisfied at α_k^* . One has

$$-g_k^\top d_k = (g(x_k + \alpha_k^* d_k) - g(x_k))^\top d_k \leq L\alpha_k^* \|d_k\|^2.$$

By using the strong convexity of f and next by choosing $\omega_1 \leq \frac{\tau}{2L}$ one has

$$\begin{aligned} f(x_k + \alpha_k^* d_k) &\leq f(x_k) - \frac{\tau}{2} \|\alpha_k^* d_k\|^2 \\ &\leq f(x_k) + \frac{\tau}{2L} \alpha_k^* g_k^\top d_k \\ &\leq f(x_k) + \alpha_k^* \omega_1 g_k^\top d_k. \end{aligned}$$

□

The proof of Theorem 3.5 shows that the Zoutendijk condition (3.2) and inequality (3.5) allow to force the convergence of $\|x_{k+1} - x_k\|$ to zero, so that the directions d_k remain bounded. For nonconvex functions, the global convergence can be obtained by bounding the distance between both points where β_{k+1} is computed. We propose to modify the line search, so that a trial step, can be used to defined an intermediate point, say \bar{x}_k , to compute β_{k+1} using the gradient values at \bar{x}_k and x_{k+1} .

The modification of the line search algorithm is very simple and only concerns the choice of the extrapolation step (Step 2.2 when $b_i = \infty$). At the beginning of the modified line search, the following additional constants are defined: $0 < \gamma_1 < \gamma_2$ and $\rho = \frac{|g_k^\top d_k|}{\|d_k\|^2}$, where k stands for the iteration index of the conjugate gradient algorithm (the index does not appear in the description of the modified line search). An intermediate step length value is introduced and initially set to $\bar{\alpha}_0 = 0$.

MODIFIED LINE SEARCH ALGORITHM (modified steps only)

- 2.2 If $b_i = \infty$, choose $\tau_{i,0} \in [\gamma_1\rho, \gamma_2\rho]$ and set $\alpha_{i,1} = \alpha_i + \tau_{i,0}$,
 else choose $\tau_{i,0} \in [\tau_1, (1 - \tau_1)]$ and set $\alpha_{i,1} = \alpha_i + \tau_{i,0}(b_i - \alpha_i)$.
 Go to Step 2.4.
3. Set $\alpha_{i+1} = \alpha_{i,j+1}$. If $b_i = \infty$ and $\alpha_{i+1} - \bar{\alpha}_i > \gamma_2\rho$, set $\bar{\alpha}_{i+1} = \alpha_i$,
 else set $\bar{\alpha}_{i+1} = \bar{\alpha}_i$. If the stopping criterion holds, stop.
-

It is not difficult to verify that Propositions 2.3 and 3.2 are still valid under this modification. To define the modified PRP algorithm, we introduce the following notation: $x_{k,i} = x_k + \alpha_i d_k$, $\bar{x}_{k,i} = x_k + \bar{\alpha}_i d_k$, $g_{k,i} = g(x_{k,i})$ and $\bar{g}_{k,i} = g(\bar{x}_{k,i})$. The stopping criterion for the line search of the modified PRP algorithm is defined by

$$|g_{k,i}^\top d_k| \leq -\omega_2 g_k^\top d_k \quad \text{and} \quad g_{k,i}^\top d_{k,i} < 0, \quad (3.7)$$

where

$$d_{k,i} = -g_{k,i} + \beta_{k,i} d_k \quad \text{and} \quad \beta_{k,i} = \frac{(g_{k,i} - \bar{g}_{k,i})^\top g_{k,i}}{\|\bar{g}_{k,i}\|^2}. \quad (3.8)$$

It follows that the scalar β_{k+1} is computed between the end point of the line search $x_{k+1} = x_k + \alpha_k d_k$ and an intermediate point, say $\bar{x}_k = x_k + \bar{\alpha}_k d_k$, where $0 \leq \bar{\alpha}_k < \alpha_{k+1}$ and such that

$$\|x_{k+1} - \bar{x}_k\| \leq \gamma_2 \frac{|g_k^\top d_k|}{\|d_k\|}. \quad (3.9)$$

Theorem 3.7 *Suppose that Assumption 3.1 holds and that the function f is bounded below on the level set \mathcal{L} . Then, the modified PRP algorithm is globally convergent in the sense that $\liminf \|g_{k,i}\| = 0$.*

Proof. The proof is by contradiction. Suppose that there exists $\varepsilon > 0$ such that

$$\|g_{k,i}\| \geq \varepsilon \quad (3.10)$$

for all iteration indices k and i . Suppose first that for some iteration k , the line search does not terminate. Proposition 2.3 implies that $\liminf_{i \rightarrow \infty} |g_{k,i}^\top d_k| = 0$. The first inequality in (3.7) is satisfied for sufficiently large i . Let us show that the descent condition (second inequality in (3.7)) is also satisfied for large i . The assumptions imply that there exists $c > 0$ such that

$$\|g(x)\| \leq c \quad \text{for all } x \in \mathcal{L}. \quad (3.11)$$

Using (3.8) and (3.10) one has,

$$|\beta_{k,i}| \leq \frac{\|g_{k,i}\|(\|g_{k,i}\| + \|\bar{g}_{k,i}\|)}{\|\bar{g}_{k,i}\|^2} \leq \frac{2c^2}{\varepsilon^2} \quad \text{for all } i \geq 1.$$

It follows that $g_{k,i}^\top d_{k,i} = -\|g_{k,i}\|^2 + \beta_{k,i} g_{k,i}^\top d_k \leq -\varepsilon + 2(c/\varepsilon)^2 |g_{k,i}^\top d_k|$, and thus $g_{k,i}^\top d_{k,i} < 0$ for sufficiently large i . This shows that for all iteration k the number of line search iterations is finite.

By using (3.8), the Lipschitz continuity of g , (3.11) and (3.9) one has

$$\begin{aligned} |\beta_{k+1}| &\leq \|\bar{g}_k\|^{-2} \|g_{k+1} - \bar{g}_k\| \|g_{k+1}\| \\ &\leq Lc\varepsilon^{-2} \|x_{k+1} - \bar{x}_k\| \\ &\leq Lc\varepsilon^{-2} \gamma_2 \frac{|g_k^\top d_k|}{\|d_k\|}. \end{aligned}$$

By the Zoutendijk condition (3.2) we deduce that $\beta_k \rightarrow 0$. By using similar arguments to those of the proof of Theorem 3.5, we show that the sequence $\{d_k\}$ is bounded and that $g_k \rightarrow 0$, a contradiction with (3.10). \square

Note that the modified PRP algorithm is related to the Grippo-Lucidi work [14, 15] in at least two points. The first point is on the use of a bound on the distance between the points where the gradient values are evaluated to compute β_{k+1} , cf (3.9). The second point is to make a distinction between the actual iterates and the iterates that are used to compute β_{k+1} .

4 Numerical experiments

We tested the algorithm described in this paper, which we call CGA (the letter A stands for adapted), on some problems of the CUTER [13] collection. The code was written in Fortran 77 with double precision on a Dell bi-Xeon 3GHz machine, with 4Go memory and running under GNU/Linux. We made a comparison with the code CG+ [19] and took the same experimental framework as described in [12]. The overall stopping test was

$$\|g(x)\|_\infty < 10^{-5}(1 + |f(x)|). \quad (4.1)$$

For the line search, we used the values $\omega_1 = 10^{-4}$ and $\omega_2 = 0.1$. The first trial step was computed with the formula proposed by Shanno and Phua [27], namely $\alpha_{0,1} = 1/\|g_0\|$ for $k = 0$ and $\alpha_{k,1} = \alpha_{k-1}g_{k-1}^\top d_{k-1}/g_k^\top d_k$ for $k \geq 1$. To preserve the finite termination property of the conjugate gradient method when the function is quadratic, we proceeded as follows. Whenever the first step is acceptable for the line search, we compute the minimizer of the quadratic that interpolates $f(x_k)$, $g_k^\top d_k$ and $g_{k,1}^\top d_k$, and the one that interpolates $f(x_k)$, $g_k^\top d_k$ and $f(x_{k,1})$. If the two minimizers are nearly equal (up to 10^{-7}), a new line search iteration is performed, otherwise the line search is terminated and the conjugate gradient iterations are pursued. We found that this strategy performs better than a systematic quadratic (or cubic) interpolation at the first line search iteration. At the other line search iterations, the computations of the trial steps were done by using quadratic and cubic interpolations formulas as described by Moré and Thente in [21, Section 4]. At Step 2.2 of the line search, the extrapolation and interpolation constants were $\tau_E = 10^{-2}$, $\tau'_E = 9$ and $\tau_I = 10^{-2}$. The stopping criterion for the line search iterations was performed as follows. At first, condition (4.1) with $x = x_{k,i}$ is tested, if it does not hold, then the curvature condition (1.5) is checked and at last the descent condition (1.6). The runs were stopped whenever the number of function evaluations exceeds 9999. The tests with both codes were done without using a restart strategy.

The computational results are reported in Table 2. The column headings have the following meanings: **pname** is the name of the problem in CUTER, **n** is the number of variables, **iter** is the number of conjugate gradient iterations, **nfev** is the number of function and gradient evaluations, f_* and $\|g_*\|_\infty$ are the function value and the infinity norm of the gradient at the last iteration. From the results, we see that both codes give comparable results, with an overall better performance for CGA. We observed one failure for CG+ due to a too large number of function evaluations (**nfev** > 9999). Note that, at each iteration, CG+ evaluates the function at least twice and computes a quadratic or cubic step to guarantee the quadratic termination property. If one modifies the code such that the first step can be accepted, the two problems ERRINROS and NONDQUAR are solved with much less functions evaluations, but other failures can be observed, for example for the quadratic problems TESTQUAD and TRIDIA. This emphasizes the importance of a careful choice of the first trial step. In view of these first tests, we can conclude that our line search technique is acceptable.

During the numerical tests with large problems, we did not observe a failure of CG+ because of a descent condition that would not have been satisfied. The only failure of this kind that we noticed, comes from a problem with only two variables, named HIMMELBB in CUTER. The minimization function is of the form $f = p^2$ where $p(u, v) = uv(1-u)(1-v-u(1-u)^5)$. At the third conjugate gradient iteration, once the step length that satisfies the strong Wolfe conditions is computed, the descent condition (1.6) is tested. But at this stage, the gradient values become very small, because the second component of the variables goes to zero, the computed values of $\|g_{k+1}\|^2$ and $\beta_{k+1}g_{k+1}^\top d_k$ are of the same order,

so that $g_{k+1}^\top d_{k+1}$ remain nonnegative (see Table 1). This example emphasizes that it is necessary to include the overall stopping test during the line search procedure.

$\ g_{k+1}\ ^2$	β_{k+1}	$g_{k+1}^\top d_k$	$g_{k+1}^\top d_{k+1}$
$1.8111E-20$	$-5.1043E-06$	$-3.5783E-15$	$1.5405E-22$
$2.0874E-22$	$-5.4710E-07$	$-3.8416E-16$	$1.7766E-24$
$2.2541E-46$	$5.6945E-19$	$3.9920E-28$	$1.9185E-48$

Table 1: Failure of the descent condition for HIMMELBB.

We performed also some experiments with the modified PRP algorithm as described at the end of Section 3. We called the corresponding algorithm CGAm. We used the values $\gamma_1 = 10^{-2}$, $\gamma_2 = 10^3$ and the extrapolation step was of the form $\alpha_{i,1} = \alpha_i + \tau_{i,0}$ where $\tau_{i,0} \in [\min(\tau_E \alpha_i, \gamma_1 \rho), \min(\tau_E' \alpha_i, \gamma_2 \rho)]$. The results are reported in Table 3. The column mod indicates the number of times that the scalar β_{k+1} was computed by using the gradient values at x_{k+1} and at an intermediate point \bar{x}_k . The results show that the performance of the CG algorithm is not greatly affected with CGAm, except for problems DIXON3DQ, POWELLSG (deterioration) and ERRINROS, FLETGBV2, NONDQUAR (improvement). Note that even when mod is equal to 0, the performances are not always equal. This is due to different values of the constants that are used in the computation of the extrapolation step in CGAm.

5 Conclusion

We have proposed a line search technique that guarantees the realization of a relaxed form of the strong Wolfe conditions and also provides a descent direction for the Polak-Ribière-Polyak conjugate gradient algorithm.

We have shown that the resulting PRP algorithm is globally convergent when the function is strongly convex, without any descent assumption on the directions.

For nonlinear nonconvex functions, we have shown that the line search can be easily adapted to modify the update formula of the PRP directions in a way that the algorithm is globally convergent.

The numerical tests validate the efficiency of the line search technique and show that the performance of the modified PRP algorithm remain at a good level. The numerical results also confirm to include a stopping test of the overall algorithm during the line search procedure.

At last, we note that the line search technique proposed here can also be used in any descent algorithm that needs a sufficient decrease of the function value at each iteration and that needs other line search conditions that are satisfied near a stationary point of the line search function.

pname	n	CG+				CGA			
		iter	nfev	f_*	$\ g_*\ _\infty$	iter	nfev	f_*	$\ g_*\ _\infty$
ARGLINA	200	1	5	2.00E+02	7.77E-14	1	4	2.00E+02	1.89E-14
ARWHEAD	500	5	15	5.32E-12	1.68E-06	5	17	2.77E-13	2.20E-06
BDQRTIC	5000	45	125	2.00E+04	1.87E-01	85	232	2.00E+04	1.99E-01
BROWNAL	200	4	35	1.47E-09	1.17E-06	3	28	1.47E-09	1.40E-06
BRYBND	5000	47	109	1.35E-11	7.09E-06	32	77	2.16E-11	7.25E-06
CHAINWOO	1000	384	798	4.57E+00	5.35E-05	361	721	2.01E+01	2.08E-04
CHNROSNB	50	246	500	7.45E-13	4.40E-06	234	450	9.78E-13	9.81E-06
COSINE	10000	4	19	-1.00E+04	3.46E-02	3	15	-1.00E+04	9.77E-02
CRAAGLVY	5000	35	95	1.69E+03	1.63E-02	32	81	1.69E+03	1.63E-02
CURLY10	1000	288	590	-1.00E+05	9.71E-01	287	538	-1.00E+05	8.91E-01
CURLY20	1000	656	1326	-1.00E+05	9.18E-01	636	1268	-1.00E+05	9.95E-01
CURLY30	1000	1219	2453	-1.00E+05	9.79E-01	884	1773	-1.00E+05	9.10E-01
DIXMAANA	3000	8	27	1.00E+00	1.13E-06	6	20	1.00E+00	1.32E-06
DIXMAANB	3000	6	23	1.00E+00	2.90E-06	5	21	1.00E+00	5.42E-06
DIXMAANC	3000	7	25	1.00E+00	5.43E-07	6	24	1.00E+00	8.44E-07
DIXMAAND	3000	8	27	1.00E+00	6.87E-06	6	24	1.00E+00	1.56E-05
DIXMAANE	3000	170	346	1.00E+00	1.96E-05	164	333	1.00E+00	1.96E-05
DIXMAANF	3000	134	277	1.00E+00	1.86E-05	152	254	1.00E+00	1.93E-05
DIXMAANG	3000	128	266	1.00E+00	1.86E-05	154	266	1.00E+00	1.94E-05
DIXMAANH	3000	128	267	1.00E+00	1.85E-05	149	251	1.00E+00	1.95E-05
DIXMAANI	3000	633	1273	1.00E+00	1.90E-05	452	904	1.00E+00	1.99E-05
DIXMAANJ	3000	121	252	1.00E+00	1.72E-05	132	224	1.00E+00	1.45E-05
DIXMAANK	3000	98	206	1.00E+00	1.79E-05	130	222	1.00E+00	1.43E-05
DIXMAANL	3000	98	207	1.00E+00	1.99E-05	131	217	1.00E+00	1.88E-05
DIXON3DQ	1000	1000	2005	1.77E-17	9.18E-10	1007	2019	1.13E-10	7.65E-06
DQDRTIC	5000	5	15	1.24E-15	1.41E-08	12	37	1.68E-12	2.59E-06
DQRTIC	5000	22	79	2.38E-07	5.97E-06	35	111	8.61E-07	1.06E-06
EDENSCH	2000	12	42	1.20E+04	2.40E-02	11	37	1.20E+04	6.51E-02
EIGENALS	110	1241	2487	6.41E-11	9.43E-06	492	947	1.01E-10	8.64E-06
EIGENBLS	110	336	678	6.69E-02	9.85E-06	315	558	6.69E-02	6.92E-06
EIGENCLS	462	1573	3158	1.78E-10	9.51E-06	1765	3280	3.80E-10	8.37E-06
ENGVAL1	5000	11	34	5.55E+03	1.99E-02	8	28	5.55E+03	3.50E-02
ERINROS	50	4995	10002	3.99E+01	2.34E-02	522	1039	3.99E+01	3.47E-04
EXTROSNB	1000	65	162	9.57E-14	7.58E-06	68	149	6.43E+01	6.17E-04
FLETCBV2	5000	2321	4643	-5.00E-01	1.50E-05	1205	1544	-5.00E-01	1.46E-05
FLETCHCR	1000	4311	8647	4.13E-11	9.35E-06	4824	7694	4.41E-11	9.47E-06
FMINSRF2	5625	268	543	1.00E+00	1.80E-05	283	427	1.00E+00	1.99E-05
FMINSURF	5625	342	691	1.00E+00	1.99E-05	481	767	1.00E+00	1.72E-05
FREUROTH	5000	12	38	6.08E+05	3.90E-01	12	36	6.08E+05	5.50E-01
GENROSE	500	1114	2256	1.00E+00	1.30E-05	1175	2061	1.00E+00	1.27E-05
HILBERTB	50	6	14	1.00E-13	6.93E-07	5	17	2.26E-12	2.19E-06
LIARWHD	5000	16	43	1.34E-20	3.23E-08	18	50	1.30E-17	6.18E-07
MOREBV	5000	36	73	8.18E-10	8.95E-06	36	75	8.18E-10	8.95E-06
MSQRTALS	100	257	518	8.65E-10	9.62E-06	296	527	1.08E-09	8.76E-06
MSQRTBLS	100	251	506	2.64E-09	8.55E-06	319	568	5.41E-10	9.09E-06
NONCVXU2	5000	716	1442	1.16E+04	1.16E-01	732	1068	1.17E+04	1.08E-01
NONCVXUN	5000	863	1735	1.16E+04	1.13E-01	812	1189	1.17E+04	1.15E-01
NONDIA	5000	5	26	1.44E-17	6.58E-08	6	31	1.00E-16	6.68E-06
NONDQUAR	5000	4717	9455	1.07E-06	9.84E-06	1211	2383	6.37E-06	6.87E-06
PENALTY1	1000	38	159	9.69E-03	3.09E-06	42	147	9.69E-03	6.62E-07
POWELLSG	5000	146	342	9.10E-07	5.19E-06	79	179	1.94E-05	8.79E-06
POWER	10000	330	669	3.08E-08	9.52E-06	408	660	3.39E-08	9.92E-06
QUARTIC	5000	22	79	2.38E-07	5.97E-06	35	111	8.61E-07	1.06E-06
SCHMIVETT	5000	10	29	-1.50E+04	1.36E-01	9	22	-1.50E+04	1.03E-01
SENSORS	100	17	42	-2.11E+03	1.43E-02	16	40	-2.11E+03	1.24E-02
SINQUAD	5000	31	82	-6.76E+06	3.39E-01	26	71	-6.76E+06	8.93E-01
SPARSQR	10000	32	103	1.89E-08	2.58E-06	29	85	5.20E-07	9.81E-06
SPMSRTLS	4999	182	370	1.70E-09	8.63E-06	184	329	2.34E-09	8.39E-06
SROSENR	5000	10	28	1.22E-14	8.75E-08	9	28	5.81E-12	1.86E-06
TESTQUAD	5000	1398	2799	2.54E-11	9.73E-06	1767	3535	8.65E-11	8.64E-06
TOINTGOR	50	61	124	1.37E+03	1.12E-02	62	113	1.37E+03	1.05E-02
TOINTGSS	5000	1	7	1.00E+01	7.19E-07	1	7	1.00E+01	9.84E-41
TOINTPSP	50	71	203	2.26E+02	2.13E-03	75	188	2.26E+02	1.78E-03
TOINTQOR	50	17	36	1.18E+03	6.30E-03	19	39	1.18E+03	6.70E-03
TQUARTIC	5000	13	41	3.40E-17	3.09E-07	12	37	1.58E-17	1.11E-06
TRIDIA	5000	735	1473	5.20E-13	9.73E-06	742	1486	4.24E-13	9.60E-06
VARDIM	200	9	44	2.11E-27	1.83E-11	7	46	2.77E-23	2.10E-09
VAREIGVL	50	82	169	1.40E-09	9.40E-06	83	153	1.62E-09	9.11E-06
WOODS	4000	16	41	7.88E+03	1.88E-02	14	36	7.88E+03	5.54E-02
TOTAL		32189	65398			23319	41918		

Table 2: Test results for CG+/CGA.

name	n	CGAm				
		iter	nfev	f_*	$\ g_*\ _\infty$	mod
ARGLINA	200	1	4	2.00E+02	1.80E-14	0
ARWHEAD	500	5	13	2.77E-13	1.93E-06	0
BDQRTIC	5000	69	192	2.00E+04	1.96E-01	0
BROWNAL	200	3	10	1.47E-09	1.40E-06	0
BRYBND	5000	32	77	2.16E-11	7.25E-06	0
CHAINWOO	1000	361	721	2.01E+01	2.08E-04	0
CHNROSNB	50	234	450	9.78E-13	9.81E-06	0
COSINE	10000	3	15	-1.00E+04	9.77E-02	0
CRAGGLVY	5000	32	81	1.69E+03	1.63E-02	0
CURLY10	1000	287	538	-1.00E+05	8.91E-01	0
CURLY20	1000	636	1268	-1.00E+05	9.95E-01	0
CURLY30	1000	884	1773	-1.00E+05	9.10E-01	0
DIXMAANA	3000	6	20	1.00E+00	1.32E-06	0
DIXMAANB	3000	5	19	1.00E+00	5.42E-06	0
DIXMAANC	3000	6	23	1.00E+00	7.45E-06	1
DIXMAAND	3000	6	24	1.00E+00	1.90E-05	0
DIXMAANE	3000	164	333	1.00E+00	1.96E-05	0
DIXMAANF	3000	157	272	1.00E+00	1.93E-05	1
DIXMAANG	3000	154	264	1.00E+00	1.95E-05	1
DIXMAANH	3000	153	251	1.00E+00	1.83E-05	1
DIXMAANI	3000	452	904	1.00E+00	1.99E-05	0
DIXMAANJ	3000	132	221	1.00E+00	1.74E-05	1
DIXMAANK	3000	142	240	1.00E+00	1.55E-05	0
DIXMAANL	3000	151	254	1.00E+00	1.74E-05	0
DIXON3DQ	1000	2000	4060	2.54E-12	3.50E-07	59
DQDRTIC	5000	12	36	1.71E-12	2.62E-06	0
DQRTIC	5000	34	105	6.02E-06	9.43E-06	1
EDENSCH	2000	10	35	1.20E+04	8.37E-02	1
EIGENALS	110	492	947	1.01E-10	8.64E-06	0
EIGENBLS	110	315	558	6.69E-02	6.92E-06	0
EIGENCLS	462	1765	3280	3.80E-10	8.37E-06	0
ENGVALL	5000	8	27	5.55E+03	3.50E-02	1
ERRINROS	50	433	879	3.99E+01	3.21E-04	2
EXTROSNB	1000	68	149	6.43E+01	6.17E-04	0
FLETCHV2	5000	33	51	-5.00E-01	1.47E-05	4
FLETCHCR	1000	4877	7788	1.17E-10	9.14E-06	0
FMINSRF2	5625	283	427	1.00E+00	1.99E-05	0
FMINSURF	5625	481	767	1.00E+00	1.72E-05	0
FREUROTH	5000	12	36	6.08E+05	5.50E-01	0
GENROSE	500	1175	2061	1.00E+00	1.27E-05	0
HILBERTB	50	6	18	5.65E-14	7.96E-07	0
LIARWHD	5000	19	51	2.78E-19	1.17E-07	0
MOREBV	5000	36	75	8.18E-10	8.95E-06	1
MSQRTALS	100	296	527	1.08E-09	8.76E-06	0
MSQRTBLS	100	319	568	5.41E-10	9.09E-06	0
NONCVXU2	5000	755	1096	1.17E+04	1.04E-01	0
NONCVXUN	5000	841	1222	1.17E+04	1.09E-01	0
NONDIA	5000	3	15	3.14E-12	1.90E-07	0
NONDQUAR	5000	967	1915	1.21E-05	7.17E-06	15
PENALTY1	1000	39	131	9.69E-03	2.49E-06	0
POWELLSG	5000	188	403	2.82E-06	3.26E-06	3
POWER	10000	406	653	3.45E-08	9.89E-06	0
QUARTC	5000	34	105	6.02E-06	9.43E-06	1
SCHMVETT	5000	9	22	-1.50E+04	1.03E-01	0
SENSORS	100	16	40	-2.11E+03	1.24E-02	0
SINQUAD	5000	26	71	-6.76E+06	9.49E-01	0
SPARSQUR	10000	29	85	5.20E-07	9.81E-06	0
SPMSRTL	4999	184	329	2.34E-09	8.39E-06	0
SROSENBR	5000	9	24	5.61E-12	1.82E-06	1
TESTQUAD	5000	1767	3535	8.65E-11	8.64E-06	0
TOINTGOR	50	62	113	1.37E+03	1.05E-02	0
TOINTGSS	5000	1	7	1.00E+01	9.84E-41	0
TOINTPSP	50	75	188	2.26E+02	1.78E-03	0
TOINTQOR	50	19	39	1.18E+03	6.70E-03	0
TQUARTIC	5000	13	42	5.50E-11	3.85E-06	11
TRIDIA	5000	742	1486	4.24E-13	9.60E-06	0
VARDIM	200	7	46	2.77E-23	2.10E-09	0
VAREIGVL	50	83	153	1.62E-09	9.11E-06	0
WOODS	4000	14	36	7.88E+03	5.54E-02	0
TOTAL		23038	42168			

Table 3: Test results for CGA modified.

Acknowledgments

This paper is based on an earlier work presented at the ISMP at Lausanne in 1997. We would like to thank Jean-Charles Gilbert who brought us this subject and Mehi Albaali for some recent discussions, which motivated us to write this report. Special thanks to Guanghui Liu, Jorge Nocedal and Richard Waltz for the availability of their software CG+ and to Dominique Orban and Philippe Toint for their help to a correct installation of CUTER. We would like to thank the referees for their valuable comments and in particular one of them who has brought Grippo-Lucidi's paper [15] to our attention

References

- [1] M. AL-BAALI AND R. FLETCHER, *An efficient line search for nonlinear least squares*, Journal of Optimization Theory and Applications, 48 (1986), pp. 359–377.
- [2] P. ARMAND AND J. C. GILBERT, *A piecewise line-search technique for maintaining the positive definiteness of the matrices in the SQP method*, Comput. Optim. Appl., 16 (2000), pp. 121–158.
- [3] Y. DAI, J. HAN, G. LIU, D. SUN, H. YIN, AND Y.-X. YUAN, *Convergence properties of nonlinear conjugate gradient methods*, SIAM J. Optim., 10 (1999), pp. 345–358.
- [4] Y. H. DAI AND Y. YUAN, *Convergence properties of the Fletcher-Reeves method*, IMA J. Numer. Anal., 16 (1996), pp. 155–164.
- [5] ———, *A nonlinear conjugate gradient method with a strong global convergence property*, SIAM J. Optim., 10 (1999), pp. 177–182 (electronic).
- [6] Y.-H. DAI AND Y.-X. YUAN, *A note on the nonlinear conjugate gradient method*, J. Comput. Math., 20 (2002), pp. 575–582.
- [7] R. FLETCHER, *Practical methods of optimization*, John Wiley & Sons Ltd., Chichester, second ed., 1987.
- [8] R. FLETCHER AND C. M. REEVES, *Function minimization by conjugate gradients*, Comput. J., 7 (1964), pp. 149–154.
- [9] J. C. GILBERT, *Maintaining the positive definiteness of the matrices in reduced secant methods for equality constrained optimization*, Mathematical Programming, 50 (1991), pp. 1–28.
- [10] ———, *On the realization of the Wolfe conditions in reduced quasi-Newton methods for equality constrained optimization*, SIAM Journal on Optimization, 7 (1997), pp. 780–813.

- [11] ———, *Piecewise line-search techniques for constrained minimization by quasi-Newton algorithms*, in *Advances in Nonlinear Programming*, Y. Yuan, ed., Kluwer, 1998, pp. 73–103.
- [12] J. C. GILBERT AND J. NOCEDAL, *Global convergence properties of conjugate gradient methods for optimization*, *SIAM Journal on Optimization*, 2 (1992), pp. 21–42.
- [13] N. I. M. GOULD, D. ORBAN, AND P. L. TOINT, *CUTEr and SifDec: A constrained and unconstrained testing environment, revisited*, *ACM Transactions on Mathematical Software*, 29 (2003), pp. 373–394.
- [14] L. GRIPPO AND S. LUCIDI, *A globally convergent version of the Polak-Ribière conjugate gradient method*, *Mathematical Programming*, 78 (1997), pp. 375–391.
- [15] L. GRIPPO AND S. LUCIDI, *Convergence conditions, line search algorithms and trust region implementations for the Polak-Ribière conjugate gradient method*, *Optim. Methods Softw.*, 20 (2005), pp. 71–98.
- [16] M. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, *Journal of Research of the National Bureau of Standards*, 49 (1952), pp. 409–436.
- [17] C. LEMARÉCHAL, *A view on line searches*, in *Optimization and Optimal Control*, A. Auslender, W. Oettly, and J. Stoer, eds., vol. 30 of *Lectures Notes in Control and Information Sciences*, Berlin, 1981, Springer Verlag, pp. 59–78.
- [18] Z. LI, J. CHEN, AND N. DENG, *A new conjugate gradient method and its global convergence properties*, *Systems Sci. Math. Sci.*, 11 (1998), pp. 53–60.
- [19] G. LIU, J. NOCEDAL, AND R. WALTZ, *CG+, version 1.1*. Optimization Technology Center, Argonne National Laboratory and Northwestern University, <http://www.ece.northwestern.edu/~nocedal/CG+.html>, April 2000.
- [20] J. J. MORÉ AND D. C. SORENSEN, *Newton's method*, in *Studies in numerical analysis*, vol. 24 of *MAA Stud. Math.*, Math. Assoc. America, Washington, DC, 1984, pp. 29–82.
- [21] J. J. MORÉ AND D. J. THUENTE, *Line search algorithms with guaranteed sufficient decrease*, *ACM Transactions on Mathematical Software*, 20 (1994), pp. 286–307.
- [22] J. NOCEDAL AND S. J. WRIGHT, *Numerical optimization*, Springer-Verlag, New York, 1999.
- [23] E. POLAK AND G. RIBIÈRE, *Note sur la convergence de méthodes de directions conjuguées*, *Revue Française d'Informatique et de Recherche Opérationnelle*, 16 (1969), pp. 35–43.

- [24] B. T. POLYAK, *The conjugate gradient method in extremal problems*, USSR Comp. Maths. and Math. Phys., 9 (1969), pp. 94–112.
- [25] M. J. D. POWELL, *Nonconvex minimization calculations and the conjugate gradient method*, in Numerical analysis (Dundee, 1983), vol. 1066 of Lecture Notes in Math., Springer, Berlin, 1984, pp. 122–141.
- [26] D. F. SHANNO AND K. H. PHUA, *Algorithm 500: Minimization of unconstrained multivariate functions*, ACM Transactions on Mathematical Software, 2 (1976), pp. 87–94.
- [27] ———, *Remark on algorithm 500: Minimization of unconstrained multivariate functions*, ACM Transactions on Mathematical Software, 6 (1980), pp. 618–622.