Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 940120, 7 pages
http://dx.doi.org/10.1155/2014/940120

Research Article

A New Method with Sufficient Descent Property for Unconstrained Optimization

Weiyi Qian and Haijuan Cui

College of Mathematics and Physics, Bohai University, Jinzhou 121000, China

Correspondence should be addressed to Weiyi Qian; weiyiqian2012@sina.cn

Received 14 September 2013; Revised 21 December 2013; Accepted 4 January 2014; Published 13 February 2014

Academic Editor: Adrian Petrusel

Recently, sufficient descent property plays an important role in the global convergence analysis of some iterative methods. In this paper, we propose a new iterative method for solving unconstrained optimization problems. This method provides a sufficient descent direction for objective function. Moreover, the global convergence of the proposed method is established under some appropriate conditions. We also report some numerical results and compare the performance of the proposed method with some existing methods. Numerical results indicate that the presented method is efficient.

1. Introduction

Consider the unconstrained optimization problem

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

where $f : R^n \rightarrow R$ is a continuously differentiable function. For solving (1), the following iterative formula is often used:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \ldots, \tag{2}$$

where x_k is the current iterative point, $\alpha_k > 0$ is a step size which is determined by some line search, and d_k is a search direction. Different search directions correspond to different iterative methods [1–4]. Throughout this paper, $g_k = \nabla f(x_k)$ is an n-dimensional column vector, $y_{k-1} = g_k - g_{k-1}$, $\|\cdot\|$ and T are defined as the Euclidian norm and transpose of vectors, respectively. Generally, if there exists a positive constant $c > 0$, such that

$$g_k^T d_k \le -c\|g_k\|^2, \tag{3}$$

then the search direction d_k possesses sufficient descent property. This property may be crucial for the iterative methods to be global convergence [5], and some numerical experiments have shown that sufficient descent methods are efficient [6]. However, not all iterative methods can satisfy sufficient descent condition (3) under some inexact linear search conditions, such as the conjugate gradient method proposed by Wei et al. [7] or the gradient method presented in [8]. In order to make the search direction d_k satisfy the condition (3) at each step, much effort has been done [9–12].

In [9], Cheng proposed a modified PRP conjugate gradient method in which the search direction d_k is determined by

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k \left(I - \dfrac{g_k g_k^T}{\|g_k\|^2} \right) d_{k-1}, & k \ge 1, \end{cases} \tag{4}$$

where $\beta_k = \beta_k^{\mathrm{PRP}} = g_k^T y_{k-1} / \|g_{k-1}\|^2$, $g_k g_k^T$ is a $n\times n$ matrix and I is an identity matrix.

In [10], Zhang et al. derived a simple sufficient descent method; the search direction d_k is given by

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \left(I - \dfrac{g_k g_k^T}{\|g_k\|^2} \right) g_{k-1}, & k \ge 1. \end{cases} \tag{5}$$

Recently, Zhang et al. [11] presented a three-term modified PRP conjugate gradient method; the search direction d_k is generated by

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k d_{k-1} - \theta_k y_{k-1}, & k \geq 1, \end{cases} \quad (6)$$

where

$$\beta_k = \beta_k^{\mathrm{PRP}} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \qquad \theta_k = \frac{g_k^T d_{k-1}}{\|g_{k-1}\|^2}. \quad (7)$$

We note that (4), (5), and (6) can be written as a linear combination of the steepest descent direction and the projection of the original direction; that is,

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \lambda_k \left(I - \dfrac{\mu_k g_k^T}{\mu_k^T g_k} \right) \overline{d}_k, & k \geq 1, \end{cases} \quad (8)$$

where $\overline{d}_k$ is an original direction, λ_k is a scalar, and $\mu_k \in R^n$ is any vector such that $\mu_k^T g_k \neq 0$ holds. Indeed, if $\lambda_k = \beta_k^{\mathrm{PRP}}$, $\mu_k = g_k$, and $\overline{d}_k = d_{k-1}$, then (8) reduces to the method (4). Let $\lambda_k = 1$, $\mu_k = g_k$, and $\overline{d}_k = g_{k-1}$; then (8) reduces to the method (5). When $\lambda_k = \beta_k^{\mathrm{PRP}}$, $\mu_k = y_{k-1}$, and $\overline{d}_k = d_{k-1}$, it is easy to deduce that (8) reduces to the method (6). From (8), we can easily obtain

$$g_k^T \left(\lambda_k \left(I - \frac{\mu_k g_k^T}{\mu_k^T g_k} \right) \overline{d}_k \right) = 0. \quad (9)$$

Thus, one has $g_k^T d_k = -\|g_k\|^2$ for all k. It implies that the sufficient descent condition (3) holds with $c = 1$. But the method (5) does not possess a restart feature which can avoid the jamming phenomenon. In addition, the methods (4) and (6) may not always be globally convergent under some inexact linear search [13], such as the standard Armijo-type line search which is given as follows:

$$\begin{gathered} \alpha_k = \max \left\{ \rho^j, j = 0, 1, 2, \ldots \right\}, \\ f(x_k + \alpha_k d_k) \leq f_k + \delta \alpha_k g_k^T d_k, \end{gathered} \quad (10)$$

where $\rho \in (0,1)$ and $\delta \in (0,1/2)$.

Motivated by (8) and (9), our purpose is to design a direction in the subspace $\{d \in R^n \mid g_k^T d = -t_k\}$, where $t_k \geq 0$ is a parameter. This direction can be written as

$$\widehat{d}_k = \lambda_k \left(I - \frac{\mu_k g_k^T}{\mu_k^T g_k} \right) \overline{d}_k - t_k \frac{v_k}{v_k^T g_k}, \quad (11)$$

where $v_k \in R^n$ is any vector such that $v_k^T g_k \neq 0$ holds. Let

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \widehat{d}_k, & k \geq 1. \end{cases} \quad (12)$$

It is clear that (8) can be regarded as a special case of (12) with $t_k = 0$. Therefore, (12) will have a wider application than (8). If we take $\lambda_k = \beta_k^{\mathrm{PRP}}$, $\mu_k = y_{k-1}$, $v_k = y_{k-1}$, $\overline{d}_k = g_{k-1}$, and $t_k = (g_k^T y_{k-1})^2 / \|g_{k-1}\|^2$ in (12), then a new search direction is given as follows:

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k g_{k-1} - \theta_k y_{k-1}, & k \geq 1, \end{cases} \quad (13)$$

where

$$\beta_k = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \qquad \theta_k = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}. \quad (14)$$

In this paper, we present a new iterative method for unconstrained optimization problems; the search direction is defined by (13) and (14). We prove that d_k satisfies $g_k^T d_k \leq -\|g_k\|^2$ without any line search. It means that the sufficient descent condition (3) holds with $c = 1$. Furthermore, we prove that the proposed method is globally convergent under the standard Armijo-type line search or the modified Armijo-type line search. From (13) and (14), we can see that the proposed method has a restart feature that directly addresses the jamming problem. In fact, when the step $x_k - x_{k-1}$ is small, then the factor y_{k-1} tends to zero vector. Therefore, the direction d_k generated by (13) is very close to the steepest descent direction $-g_k$.

The rest of this paper is organized as follows. In Section 2, we propose a new algorithm and discuss its sufficient descent property. In Section 3, the global convergence of the proposed method is proved under the modified Armijo-type line search or the standard Armijo line search. Some numerical results are given to test the performance of the proposed method in Section 4. Finally, we have some conclusions about the proposed method.

2. New Algorithm

In this section, the specific iterative steps of the proposed algorithm are listed as follows.

Algorithm 1. Consider the following.

Step 1. Choose parameters $\delta \in (0,1)$, $\rho \in (0,1)$, and $\beta > 0$; given an initial point $x_0 \in R^n$. Set $d_0 = -g_0$ and $k := 0$.

Step 2. If $\|g_k\| = 0$, then stop; otherwise go to the next step.

Step 3. Determine a step size α_k satisfying modified Armijo-type line search conditions:

$$\begin{gathered} \alpha_k = \max \left\{ \beta \rho^j, j = 0, 1, 2, \ldots \right\}, \\ f(x_k + \alpha_k d_k) \leq f(x_k) - \delta \alpha_k^2 \|d_k\|^2. \end{gathered} \quad (15)$$

Step 4. Let $x_{k+1} = x_k + \alpha_k d_k$.

Step 5. Calculate the search direction d_{k+1} by (13) and (14).

Step 6. Set $k := k + 1$, and go to Step 2.

Theorem 2. *Let sequences* $\{d_k\}$ *and* $\{x_k\}$ *be generated by* (13) *and* (2)*; then*

$$g_k^T d_k \leq -\|g_k\|^2, \tag{16}$$

for all $k \geq 0$.

Proof. Obviously, the conclusion is true for $k = 0$.

If $k \geq 1$, multiplying (13) by g_k^T, we have

$$\begin{aligned} g_k^T d_k &= -\|g_k\|^2 + g_k^T\left(\beta_k g_{k-1} - \theta_k y_{k-1}\right) \\ &= -\|g_k\|^2 + \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2} g_k^T g_{k-1} \\ &\quad - \frac{\|g_k\|^2}{\|g_{k-1}\|^2} g_k^T y_{k-1} \\ &= -\|g_k\|^2 + \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}\left(g_k^T g_{k-1} - \|g_k\|^2\right) \\ &= -\|g_k\|^2 + \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2} g_k^T\left(g_{k-1} - g_k\right) \\ &= -\|g_k\|^2 - \frac{\left(g_k^T y_{k-1}\right)^2}{\|g_{k-1}\|^2} \\ &\leq -\|g_k\|^2. \end{aligned} \tag{17}$$

Therefore, the inequality (16) holds for all $k \geq 0$. The proof is completed. □

Theorem 2 shows that the search direction d_k given by (13) possesses the sufficient descent property for any line search.

3. Convergence Analysis

The following assumptions are often needed to prove the global convergence of nonlinear conjugate gradient methods [14, 15]. In this section, we also use these assumptions in the convergence analysis of the proposed method.

Assumption 3. Consider the following.

(i) The level set $S = \{x \in R^n : f(x) \leq f(x_0)\}$ is bounded.

(ii) In a neighborhood N of S, the function f is continuously differentiable and its gradient is Lipchitz continuous; namely, there exists a constant $L > 0$, such that

$$\|g(x) - g(y)\| \leq L\|x - y\|, \quad \forall x, y \in N. \tag{18}$$

Lemma 4. *Suppose that Assumption 3 holds. Let* $\{x_k\}$ *and* $\{d_k\}$ *be generated by Algorithm 1. If the step size* α_k *is obtained by* (15) *or* (10)*, then there exists a constant* $m > 0$*, such that*

$$\alpha_k \geq m\frac{\|g_k\|^2}{\|d_k\|^2}, \tag{19}$$

and one can also have

$$\sum_{k=0}^{\infty}\frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \tag{20}$$

Proof. The results of this lemma will be proved in the following two cases.

Case 1. Let the step size α_k be computed by (15). From Theorem 2, we have $\|g_k\|\|d_k\| \geq -g_k^T d_k \geq \|g_k\|^2$; thus $\|d_k\| \geq \|g_k\|$. If $\alpha_k = \beta$, then we obtain $\alpha_k \geq \beta\|g_k\|^2/\|d_k\|^2$. If $\alpha_k < \beta$, then we know $\rho^{-1}\alpha_k$ does not satisfy the inequality (15). So we have

$$f\left(x_k + \rho^{-1}\alpha_k d_k\right) - f_k > -\delta\alpha_k^2\rho^{-2}\|d_k\|^2. \tag{21}$$

By Assumption 3(ii) and the mean value theorem, we have

$$\begin{aligned} &f\left(x_k + \rho^{-1}\alpha_k d_k\right) - f_k \\ &\quad = \rho^{-1}\alpha_k g\left(x_k + t_k\rho^{-1}\alpha_k d_k\right)^T d_k = \rho^{-1}\alpha_k g_k^T d_k \\ &\quad + \rho^{-1}\alpha_k\left(g\left(x_k + t_k\rho^{-1}\alpha_k d_k\right) - g_k\right)^T d_k \\ &\quad \leq \rho^{-1}\alpha_k g_k^T d_k + L\rho^{-2}\alpha_k^2\|d_k\|^2, \end{aligned} \tag{22}$$

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

From (21) and (22), we have

$$-\delta\alpha_k^2\rho^{-2}\|d_k\|^2 < \rho^{-1}\alpha_k g_k^T d_k + L\rho^{-2}\alpha_k^2\|d_k\|^2. \tag{23}$$

Using Theorem 2 again, we get

$$\alpha_k > \frac{\rho\|g_k\|^2}{(L + \delta)\|d_k\|^2}. \tag{24}$$

Let $m = \min\{\beta, \rho/(L+\delta)\}$; then the inequality (19) is obtained.

From Assumption 3(i), there exists a constant $M > 0$, such that $|f(x)| < M$, $\forall x \in S$. By (15), (19), and Theorem 2, we have

$$\begin{aligned} \sum_{k=0}^{n-1}\left(\delta m^2\frac{\|g_k\|^4}{\|d_k\|^4}\|d_k\|^2\right) &\leq \sum_{k=0}^{n-1}\left(\delta\alpha_k^2\|d_k\|^2\right) \\ &\leq \sum_{k=0}^{n-1}\left(f_k - f_{k+1}\right) < 2M. \end{aligned} \tag{25}$$

Therefore, from the above inequality, we have

$$\sum_{k=0}^{\infty}\frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \tag{26}$$

Case 2. Let the step size α_k be computed by (10). Similar to the proof of the above case, we can obtain

$$\begin{aligned} \alpha_k &\geq \frac{\|g_k\|^2}{\|d_k\|^2}, \quad \text{if } \alpha_k = 1, \\ \alpha_k &> \frac{\rho(1-\delta)\|g_k\|^2}{L\|d_k\|^2}, \quad \text{if } \alpha_k < 1. \end{aligned} \tag{27}$$

Let $m = \min\{1, \rho(1-\delta)/L\}$; then the inequality (19) is obtained. From (10), (19), and Theorem 2, we obtain

$$\sum_{k=0}^{n-1}\left(\delta m\frac{\|g_k\|^2}{\|d_k\|^2}\|g_k\|^2\right) \le \sum_{k=0}^{n-1}\left(-\delta\alpha_k g_k^T d_k\right) \le \sum_{k=0}^{n-1}\left(f_k - f_{k+1}\right) < 2M. \tag{28}$$

By the above inequality, we can get (20). The proof is completed. □

Theorem 5. *Suppose that Assumption 3 holds. If Algorithm 1 generates infinite sequences $\{d_k\}$ and $\{x_k\}$, then one has*

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

Proof. We obtain this conclusion (29) by contradiction. Suppose that (29) does not hold, then there exists a positive constant $\lambda_1 > 0$, such that $\|g_k\| \ge \lambda_1$, for all $k \ge 0$. From Assumption 3(i), we know that there also exists a positive constant $\lambda_2 > 0$, such that $\|g_k\| \le \lambda_2$, for all $k \ge 0$. Since $d_k = -g_k + \beta_k g_{k-1} + \theta_k y_{k-1}$, then we have

$$\begin{aligned}\|d_k\| &\le \|g_k\| + |\beta_k|\,\|g_{k-1}\| + |\theta_k|\,\|y_{k-1}\| \\ &\le \|g_k\| + \frac{\|g_k\|\left(\|g_k\| + \|g_{k-1}\|\right)}{\|g_{k-1}\|^2}\|g_{k-1}\| \\ &\quad + \frac{\|g_k\|^2}{\|g_{k-1}\|^2}\left(\|g_k\| + \|g_{k-1}\|\right) \\ &\le \lambda_2 + \frac{2\lambda_2^2}{\lambda_1} + \frac{2\lambda_2^3}{\lambda_1^2} \\ &\triangleq M_1.\end{aligned} \tag{30}$$

The above inequality implies

$$\sum_{k=0}^{\infty}\frac{\|g_k\|^4}{\|d_k\|^2} \ge \sum_{k=0}^{\infty}\frac{\lambda_1^4}{M_1^2}, \tag{31}$$

which contradicts with (20). This completes the proof. □

Remark 6. If the search direction d_k is defined by (13) with $\beta_k = -(g_k^T y_{k-1})/(g_{k-1}^T d_{k-1})$, $\theta_k = -\|g_k\|^2/(g_{k-1}^T d_{k-1})$, then the sufficient descent property and global convergence can also be proved similar to the proof of Theorems 2 and 5.

4. Numerical Results

In this section, some numerical results are provided to test the performance of the proposed method, and the proposed method is compared with the existing methods [9–11]. For the sake of simplicity, the proposed method and other comparative methods are named by NSDM, LPRP [11], SSD [10], and MPRP [9], respectively. The test problems and initial points are from [16]. The test problems are listed in Table 1. In our experiment, all the codes were written in MATLAB 7.0 and run on PC with 2.00 GB RAM memory, 2.10 GHz CPU, and windows 7 operation system.

Table 1: The test problems.

Number	Function name
P1	Generalized Tridiagonal 1
P2	Extended Himmelblau
P3	Liarwhd
P4	Diagonal 7
P5	Diagonal 8
P6	Nonscomp
P7	Cosine
P8	Hager
P9	Diagonal 2
P10	Raydan 1
P11	Extended Penalty
P12	Diagonal 3
P13	Generalized Quartic
P14	Power
P15	Extended Denschnf
P16	Perturbed Tridiagonal Quadratic
P17	Extended Denschnb
P18	Raydan 2
P19	Almost Perturbed Quadratic
P20	Extended BD1
P21	Extebded Tet
P22	Extended Denschnb
P23	Arwhead
P24	Extended Tridiagonal 2
P25	Quartc
P26	Extended Maratos
P27	Engval 1
P28	Extended Quadratic Exponential EP1

In all algorithms, the step size α_k is computed satisfying the modified Armijo-type line search (15) with $\delta = 0.1$, $\rho = 0.1$, and $\beta = 1$, and the stopping condition is $\|g_k\| \le 10^{-5}$. We also stop these algorithms if CPU time is over 500(s).

In Table 2, P, N, NI, NF, NG, and CPU stand for th number of test problems, the dimension of the vectors, the number of iterations, the number of function evaluations, the number of gradient evaluations, and the run time of CPU in seconds, respectively. The symbol "—" means that the corresponding method fails in solving the test problems when the CPU time is more than 500 seconds, and the star $*$ denotes that the numerical result is the best one among all the comparative methods.

In Table 2, we compare the performance of the new method by testing 28 different problems. According to the distribution of the star $*$, one can see that the NSDM method performs better than the LPRP, MPRP, and SSD methods with 14 test problems, worse than the MPRP method with 1 test problem and worse than the LPRP method with 6 test

Table 2: The numerical results of the NSDM/LPRP/SSD/MPRP methods.

P	N	NSDM NI/NF/NG/CPU	LPRP NI/NF/NG/CPU	SSD NI/NF/NG/CPU	MPRP NI/NF/NG/CPU
P1	400	57/164/58/1.934*	70/199/71/2.098	65/187/66/2.337	74/210/75/2.984
P2	1000	53/161/54/4.715*	60/181/61/4.764	119/361/120/13.974	58/175/59/7.881
P3	900	24/68/25/3.276*	68/140/69/8.175	65/199/66/9.594	80/216/81/13.400
P4	1000	36/73/37/5.990*	41/83/42/6.053	41/83/42/7.410	41/83/42/8.424
P5	900	29/59/30/3.946*	36/73/37/4.352	36/73/37/5.336	36/73/37/6.052
P6	300	70/213/71/1.424*	108/310/109/1.921	293/879/294/6.316	—/—/—/—
P7	4000	41/115/42/32.339*	73/201/74/49.889	79/216/80/95.581	82/203/83/115.440
P8	100	57/118/58/0.156	65/125/66/0.172	100/218/101/0.280	59/109/60/0.188
P9	100	960/1108/961/2.606	780/781/781/1.888*	1096/1266/1097/2.886	780/781/781/2.293
P10	100	362/802/363/0.967	230/414/231/0.546	742/1578/743/1.872	151/266/152/0.437*
P11	1000	53/186/54/9.388*	65/245/66/10.329	146/496/147/28.011	64/242/65/14.234
P12	1000	44/89/45/7.896*	49/99/50/7.933	49/99/50/9.718	49/99/50/10.955
P13	3000	52/105/53/35.802	54/109/55/33.056	55/116/56/48.891	54/109/55/55.973
P14	200	613/2798/614/5.210*	839/4045/840/6.412	650/2990/651/5.491	1601/5914/1602/16.114
P15	800	31/118/32/3.354*	86/331/87/8.206	78/304/79/9.142	82/302/83/10.982
P16	100	298/1015/299/0.796	157/504/158/0.374	499/1943/500/1.310	143/480/144/0.421
P17	1000	67/135/68/5.523	71/143/72/5.210	70/141/71/7.317	70/141/71/8.486
P18	3000	13/20/14/10.076	5/6/6/3.659*	5/6/6/5.373	5/6/6/6.194
P19	100	274/937/275/0.734	125/396/126/0.312*	544/2281/545/1.435	141/448/142/0.421
P20	3000	47/110/48/16.895	23/49/24/7.395*	58/140/59/39.243	27/59/28/19.451
P21	500	59/129/60/1.420	44/89/45/0.951*	81/185/82/2.527	46/93/47/1.576
P22	2000	69/139/70/23.469	73/147/74/22.386	73/147/74/32.423	73/147/74/36.179
P23	500	183/914/184/8.221*	—/—/—/—	—/—/—/—	—/—/—/—
P24	500	87/176/88/4.072	74/136/75/3.089	338/678/339/16.957	60/109/61/3.463
P25	100	3093/3096/3094/9.485	3145/3147/3146/8.159	3145/3147/3146/9.064	3145/3147/3146/9.984
P26	100	293/1189/294/0.936	111/410/112/0.327*	—/—/—/—	131/447/132/0.421
P27	1000	78/184/79/12.699*	92/238/93/13.478	101/267/102/18.533	—/—/—/—
P28	200	17/70/18/0.092*	29/121/30/0.137	29/121/30/0.183	29/121/30/0.198

problems. However, there also exist 7 test problems that are not marked by the symbol $*$. Among these 7 test problems, the NSDM method performs better than other methods with 5 test problems in the number of iterations, 4 test problems in the number of function evaluations, 5 test problems in the number of gradient evaluations, and 1 test problem in CPU time.

In order to compare the performance of these methods clearly, we adopt the performance profiles introduced by Dolan and Moré [17]. The performance results are shown in Figures 1–4, respectively. In [17], Dolan and Moré introduced the notion as a means to evaluate and compare the performance of the set solvers S on a test set P. Assuming n_s solvers and n_p problems exist, for each problem p and solver s, they defined

$$t_{p,s} = \text{computing time (the number of iterations or others)} \\ \text{required to solve problem } p \text{ by solver } s. \tag{32}$$

The performance ratio is given by

$$\gamma_{p,s} = \frac{t_{p,s}}{\min\left\{t_{p,s} : s \in S\right\}}. \tag{33}$$

Assume that a parameter $\gamma_M \geq \gamma_{p,s}$ for all p, s is chosen, and $\gamma_{p,s} = \gamma_M$ if and only if solver s does not solve problem p. The performance profile is defined by

$$P_s(t) = \frac{1}{n_p}\text{size}\left\{p \in P : \gamma_{p,s} \leq t\right\}. \tag{34}$$

Hence, $P_s(t)$ is the probability for solver $s \in S$ that a performance ratio $\gamma_{p,s}$ is within a factor $t \in R$ of the best possible ratio. The performance profile $P_s : R \to [0, 1]$ for a solver was nondecreasing, piecewise, and continuous from the right. The value of $P_s(1)$ is the probability that the solver will win over the rest of the solvers. In general, a solver with high values of $P_s(t)$ or at the top right of the figure is preferable or represents the best solver.

From Figures 1–4, we can obviously see that the NSDM method performs better than the MPRP method and SSD

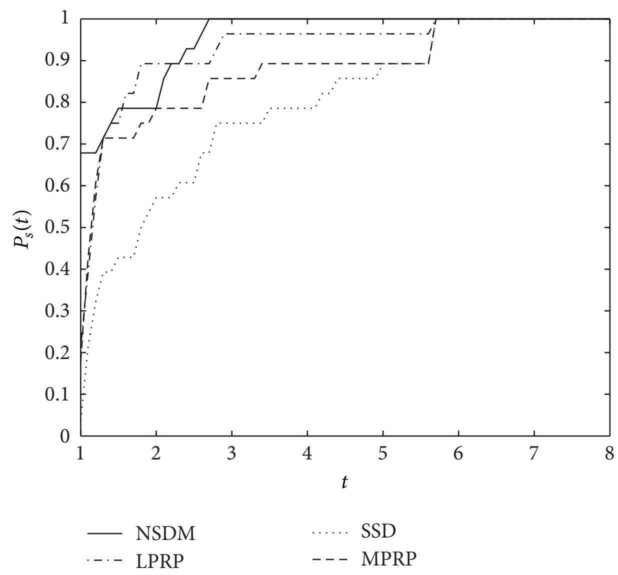

Figure 1: Performance profiles about the number of iterations.

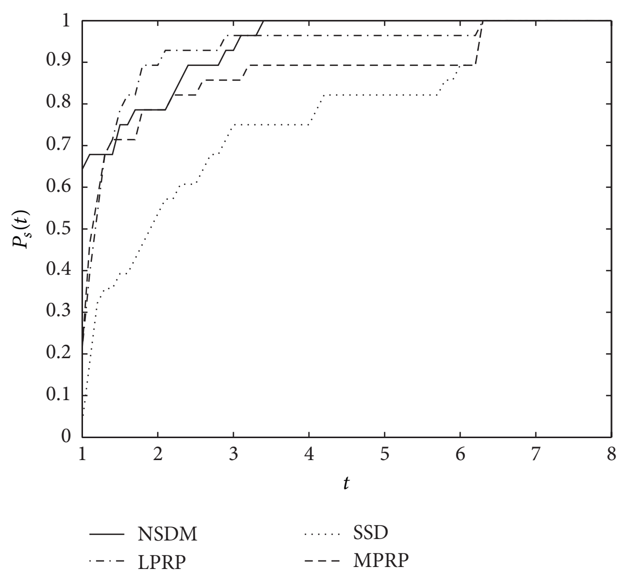

Figure 2: Performance profiles about the number of function evaluations.

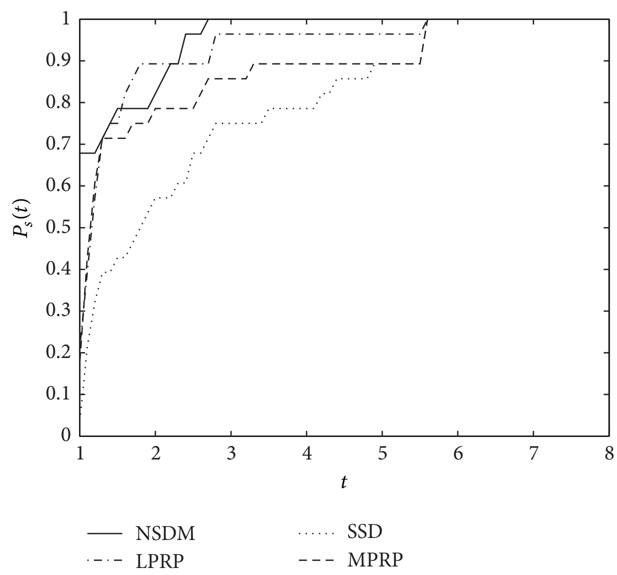

Figure 3: Performance profiles about the number of gradient evaluations.

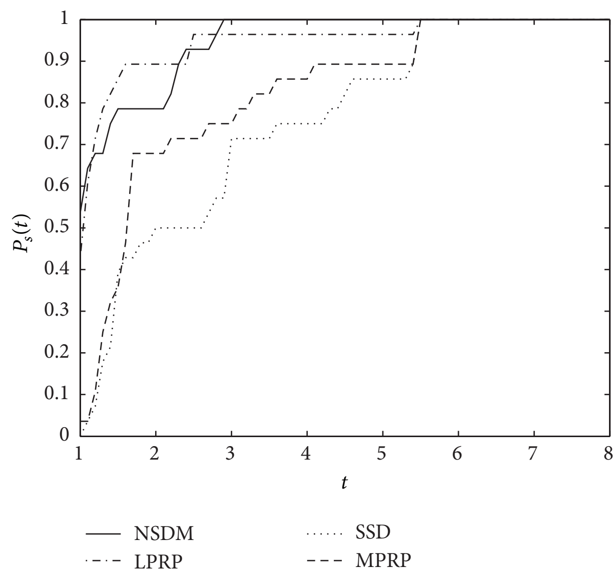

Figure 4: Performance profiles about CPU time.

method. Although the LPRP method outperforms the NSDM method for $1.2 < t < 2.4$ in Figure 1, $1.2 < t < 3.2$ in Figure 2, $1.2 < t < 2.2$ in Figure 3, and $1.1 < t < 2.8$ in Figure 4, the NSDM method is superior to the LPRP method in the remaining interval. Moreover, from Figures 1–4, we can see that the NSDM method can solve 100% of the test problems, while the LPRP method can solve about 96% of the problems. Hence, the NSDM method is superior to the LPRP method. By comparing the value of $P_s(1)$ in Figures 1–4, one can have a conclusion that the NSDM method is competitive to others; for example, the NSDM method is superior to other methods at least 45% in the number of iterations. In a word, one can have a conclusion that the presented method is much better than the LPRP, MPRP, and SSD methods from the analysis of the numerical results.

5. Conclusions

In this paper, we have proposed a new formula (11) that can generate different search directions by taking different parameters. Based on this formula, we have proposed a new sufficient descent method for solving unconstrained optimization problems. At each iteration, the generated direction is only related to the gradient information of two successive points. We have shown that this method is globally convergent. The numerical results indicate that the given method is superior to other methods for the test problems. In the future, we will study much better iterative methods according to (11) and perform new convergence analysis on them.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the editor and anonymous referees for their valuable comments and suggestions, which improve this paper greatly. This work is partly supported by the National Natural Science Foundation of China (11371071), Natural Science Foundation of Liaoning Province (20102003), Scientific Research Foundation of Liaoning Province Educational Department (L2013426), and Graduate Innovation Foundation of Bohai University (201208).

References

[1] Z. Dai and F. Wen, "Another improved Wei-Yao-Liu nonlinear conjugate gradient method with sufficient descent property," *Applied Mathematics and Computation*, vol. 218, no. 14, pp. 7421–7430, 2012.

[2] K. Ueda and N. Yamashita, "Convergence properties of the regularized Newton method for the unconstrained nonconvex optimization," *Applied Mathematics and Optimization*, vol. 62, no. 1, pp. 27–46, 2010.

[3] M. Raydan, "The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem," *SIAM Journal on Optimization*, vol. 7, no. 1, pp. 26–33, 1997.

[4] Y. Xiao, Z. Wei, and Z. Wang, "A limited memory BFGS-type method for large-scale unconstrained optimization," *Computers & Mathematics with Applications*, vol. 56, no. 4, pp. 1001–1009, 2008.

[5] J. C. Gilbert and J. Nocedal, "Global convergence properties of conjugate gradient methods for optimization," *SIAM Journal on Optimization*, vol. 2, no. 1, pp. 21–42, 1992.

[6] D.-H. Li and B.-S. Tian, "n-step quadratic convergence of the MPRP method with a restart strategy," *Journal of Computational and Applied Mathematics*, vol. 235, no. 17, pp. 4978–4990, 2011.

[7] Z. Wei, S. Yao, and L. Liu, "The convergence properties of some new conjugate gradient methods," *Applied Mathematics and Computation*, vol. 183, no. 2, pp. 1341–1350, 2006.

[8] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA Journal of Numerical Analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[9] W. Cheng, "A two-term PRP-based descent method," *Numerical Functional Analysis and Optimization*, vol. 28, no. 11-12, pp. 1217–1230, 2007.

[10] M.-L. Zhang, Y.-H. Xiao, and D. Zhou, "A simple sufficient descent method for unconstrained optimization," *Mathematical Problems in Engineering*, vol. 2010, Article ID 684705, 9 pages, 2010.

[11] L. Zhang, W. Zhou, and D.-H. Li, "A descent modified Polak-Ribière-Polyak conjugate gradient method and its global convergence," *IMA Journal of Numerical Analysis*, vol. 26, no. 4, pp. 629–640, 2006.

[12] X.-M. An, D.-H. Li, and Y. Xiao, "Sufficient descent directions in unconstrained optimization," *Computational Optimization and Applications*, vol. 48, no. 3, pp. 515–532, 2011, Supplementary material available online.

[13] Z. Dai, "Two modified Polak-Ribière-Polyak-type nonlinear conjugate methods with sufficient descent property," *Numerical Functional Analysis and Optimization*, vol. 31, no. 7–9, pp. 892–906, 2010.

[14] Z. Wan, C. Hu, and Z. Yang, "A spectral PRP conjugate gradient methods for nonconvex optimization problem based on modified line search," *Discrete and Continuous Dynamical Systems B*, vol. 16, no. 4, pp. 1157–1169, 2011.

[15] Z. Wei, G. Li, and L. Qi, "New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems," *Applied Mathematics and Computation*, vol. 179, no. 2, pp. 407–430, 2006.

[16] N. Andrei, "An unconstrained optimization test functions collection," *Advanced Modeling and Optimization*, vol. 10, no. 1, pp. 147–161, 2008.

[17] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.