Wei-Yao-Liu Conjugate Gradient Algorithm for Nonsmooth Convex Optimization Problems

Yaping Hu*, Liying Liu, Yujie Wang

College of Science, Tianjin University of Science and Technology, China

Abstract   This paper presents a Wei-Yao-Liu conjugate gradient algorithm for nonsmooth convex optimization problem. The proposed algorithm makes use of approximate function and gradient values of the Moreau-Yosida regularization function instead of the corresponding exact values. Under suitable conditions, the global convergence property could be established for the proposed conjugate gradient method. Finally, some numerical results are reported to show the efficiency of our algorithm.

Keywords   Nonsmooth convex minimization, Wei-Yao-Liu conjugate gradient algorithm, Moreau-Yosida regularization, Global convergence.

1. Introduction

We consider the unconstrained minimization problem

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

where $f : R^n \to R$ is a nonsmooth convex function. The following problem

$$\min_{x \in R^n} F(x) \tag{2}$$

is the so-called Moreau-Yosida regularization of f [10], which is defined by

$$F(x) = \min_{z \in R^n} \{ f(z) + \frac{1}{2\lambda} \|z - x\|^2 \}, \tag{3}$$

where λ is a positive parameter and $\| \cdot \|$ denotes the Euclidean norm. It is well known that problems (1) and (2) are equivalent in the sense that the solution sets of the two problems coincidentally are the same. The function F has some good properties: it is a differentiable convex function, it has a Lipschitz continuous gradient even when the function f is nondifferentiable, and although F is not twice differentiable in general, but the gradient function of F can be proved to be semismooth under some reasonable conditions [8, 21]. Based on these features, some algorithms have been proposed for solving (2), see [2, 8, 21, 24]. The proximal methods have been proved to be effective in dealing with evaluating the function value of $F(x)$ and its gradient $\nabla F(x)$ exactly at a given point x

---

*Correspondence to: Yaping Hu (Email: huyaping@tust.edu.cn). College of Science, Tianjin University of Science and Technology, 29 Thirteenth Street, TEDA, Tianjin, China (300457).

[1, 3, 6, 23]. Lukšan [15] and Monjezi [19] proposed the bundle method, which can handle convex and nonconvex f. And many trust region algorithms of minimizing a nonsmooth objective function have been presented, see [5, 13, 27, 30, 33]. Recently, Yuan et al. [28, 29, 32] and Li [16, 17] have extended the spectral gradient method and conjugate gradient-type method to solve nonsmooth optimization problem, respectively.

Conjugate gradient techniques have been developed for solving large-scale optimization problems recently (see [9, 12, 29, 31] etc.). Motivated by these techniques as well as the Moreau-Yosida regulation (smoothing) approach, we will propose the Wei-Yao-Liu conjugate gradient algorithm for solving a nonsmooth unconstrained convex minimization problem in this paper. The Wei-Yao-Liu (WYL) conjugate gradient method with formula (21) has not only excellent numerical results but also some good properties. First, the β_k^{WYL} always retains nonnegativity while $g_k \neq 0$. Second, the WYL method has a nice property which was introduced by Gilbert and Nocedal [9]. This property pertains to the PRP method and ensures that the small steplength would not be too much long so that it is helpful for the global convergence, see [25] for details. Even more, the β_k^{WYL}, like β_k^{PRP}, β_k^{HS} and β_k^{LS} whose numerators share the common $g_k^T(g_k - g_{k-1})$, which can avoid jamming automatically. Further research of the WYL method can be found in [11, 12, 18, 25, 26]. The purpose of this paper is to extend Wei-Yao-Liu conjugate gradient algorithm to solve the nonsmooth optimization problem (1). The presented algorithm has the following main attributes: (1) A WYL conjugate gradient algorithm is introduced for nonsmooth problem (1) and smooth problem (2); (2) The search direction satisfies the sufficient descent property; (3) This algorithm possesses the global convergence; (4) Numerical results show that this algorithm is efficient.

This paper is organized as follows. In Section 2, we briefly review some basic results in convex analysis and nonsmooth analysis. In Section 3, we present a WYL conjugate gradient algorithm for (1), prove its global convergence, and then present some numerical results in Section 4. Finally, we have a conclusion section.

2. Preliminaries

In this section, we review some important and useful results in convex analysis and nonsmooth analysis. The following proposition ([10], Chapter XV, Theorem 4.1.4) shows some basic properties of the Moreau-Yosida regularization function F.

Proposition 2.1
The function F is finite-valued, convex, and everywhere differentiable with gradient

$$g(x) = \nabla F(x) = \frac{x - p(x)}{\lambda}, \tag{4}$$

where $p(x)$ is the unique minimizer in (3), that is

$$p(x) = arg \min_{z \in R^n} \{f(z) + \frac{1}{2\lambda}\|z - x\|^2\}. \tag{5}$$

Moreover, the gradient mapping $g : R^n \to R^n$ is globally Lipschitz continuous with modulus $\frac{1}{\lambda}$ for all $x, y \in R^n$, i.e.,

$$\|g(x) - g(y)\| \leq \frac{1}{\lambda}\|x - y\|. \tag{6}$$

The generalized Jacobian of $\nabla F(x)$ and the property of BD-regularity can be found in [4, 13, 20].

Proposition 2.2
By the Rademacher theorem and the Lipschitzian property of $\nabla F(x)$, for each $x \in \Re^n$, the set of generalized Jacobian matrices

$$\partial_B g(x) = \{V \in \Re^{n \times n} : V = \lim_{x_k \to x} \nabla g(x_k), x_k \in D_g\},$$

is nonempty and compact, where $D_g = \{x \in \Re^n : g \text{ is differentiable at } x\}$.
(i) Since g is a gradient mapping of the convex function F, every $V \in \partial_B g(x)$ is a symmetric positive semidefinite

matrix for each $x \in \Re^n$.

(ii) If g is BD-regular at x, which means all matrices $V \in \partial_B g(x)$ are nonsingular, then there exist constants $\mu_1 > 0$, $\mu_2 > 0$ and a neighborhood Ω of x such that for all $y \in \Omega$

$$d^T V d \geq \mu_1 \|d\|^2, \|V^{-1}\| \leq \mu_2, \forall d \in \Re^n, V \in \partial_B g(y).$$

The next proposition ([10], Chapter XV, Theorem 4.1.7) formally states the equivalence between problems (1.1) and (1.2).

Proposition 2.3
The following statements are equivalent:
(i) x minimizes f on R^n;
(ii) $x = p(x)$;
(iii) $g(x) = 0$;
(iv) x minimizes F on R^n.

It is obvious that $F(x)$ and $g(x)$ can be obtained through the optimal solution of (3), but $p(x)$ is difficult and sometimes impossible to be solved precisely, so we often use the approximate values of $F(x)$ and $g(x)$ in the real computation. Suppose that for each $x \in R^n$ and any $\varepsilon > 0$, there exists an approximation vector $p^a(x, \varepsilon) \in R^n$ of the unique minimizer $p(x)$ in (3) such that

$$f(p^a(x,\varepsilon)) + \frac{1}{2\lambda}\|p^a(x,\varepsilon) - x\|^2 \leq F(x) + \varepsilon. \tag{7}$$

An implementable procedure for finding such an approximate minimizer may be found, for example, in [7]. The existence and computation theorem of $p^a(x, \varepsilon)$ are presented as following.

Proposition 2.4
[Lemma 2.1 in [6]] Let $\{x_k\}$ be constructed according to the formula

$$x_{k+1} = x_k - \alpha_k \upsilon_k, \ for \ k = 1, 2, \cdots , \tag{8}$$

where $\alpha_k > 0$ is a stepsize and υ_k is an approximate subgradient at x_k, i.e.,

$$\upsilon_k \in \partial_{\varepsilon_k} f(x_k) = \{\upsilon | f(x_k) - \langle \upsilon, x_k \rangle \leq f(x_k) + \varepsilon_k\}, \ for \ k = 1, 2, \cdots . \tag{9}$$

(i) If υ_k satisfies

$$\upsilon_k \in \partial f(x_{k+1}), \ for \ k = 1, 2, \cdots , \tag{10}$$

then (9) holds with

$$\varepsilon_k = f(x_k) - f(x_{k+1}) - \alpha_k \|\upsilon_k\|^2 \geq 0. \tag{11}$$

(ii) Conversely, if (9) holds with ε_k given by (11), then (10) holds: $x_{k+1} = p^a(x_k, \varepsilon_k)$.

We can use $p^a(x, \varepsilon)$ to define approximations of $F(x)$ and $g(x)$ by

$$F^a(x, \varepsilon) = f^a(x, \varepsilon) + \frac{1}{2\lambda}\|p^a(x, \varepsilon) - x\|^2, \tag{12}$$

and

$$g^a(x, \varepsilon) = \frac{1}{\lambda}(x - p^a(x, \varepsilon)), \tag{13}$$

respectively, where

$$p^a(x, \varepsilon) = arg \min_{z \in R^n} \{f(z) + \frac{1}{2\lambda}\|z + \varepsilon - x\|^2\}. \tag{14}$$

The next proposition (see [8]) states that we can compute the approximations $F^a(x, \varepsilon)$ and $g^a(x, \varepsilon)$ by choosing the parameter ε to be sufficiently small such that the approximations remain as close as possible to $F(x)$ and $g(x)$.

Proposition 2.5
Let $p^a(x, \varepsilon)$ be a vector that satisfies (14), and let $F^a(x, \varepsilon)$ and $g^a(x, \varepsilon)$ be defined by (12) and (13), respectively.
Then, we obtain
(i) $F(x) \leq F^a(x, \varepsilon) \leq F(x) + \varepsilon$;
(ii) $\|p^a(x, \varepsilon) - p(x)\| \leq \sqrt{2\lambda\varepsilon}$;
(iii) $\|g^a(x, \varepsilon) - g(x)\| \leq \sqrt{2\varepsilon/\lambda}$;

A remarkable property of $g^a(x, \varepsilon)$ is given as follows.

Proposition 2.6
[Lemma 4.3 in [22]] There exist positive constants l and L, and a positive integer k_0 such that

$$\frac{\langle g^a(x_{k+1}, \varepsilon_{k+1}) - g^a(x_k, \varepsilon_k), x_{k+1} - x_k \rangle}{\|x_{k+1} - x_k\|^2} \geq l, \tag{15}$$

and

$$\frac{\|g^a(x_{k+1}, \varepsilon_{k+1}) - g^a(x_k, \varepsilon_k)\|^2}{\langle x_{k+1} - x_k, g^a(x_{k+1}, \varepsilon_{k+1}) - g^a(x_k, \varepsilon_k) \rangle} \leq L, \tag{16}$$

for all $k \geq k_0$.

By (15) and (16), it is easy to deduce that

$$\|g^a(x_{k+1}, \varepsilon_{k+1}) - g^a(x_k, \varepsilon_k)\| \leq L\|x_{k+1} - x_k\|. \tag{17}$$

3. Algorithm

By using the Moreau-Yosida regularization (smoothing) approach and a nonmonotone line search technique, we propose a Wei-Yao-Liu conjugate gradient algorithm for solving a nonsmooth unconstrained convex minimization problem in this section. First we use the tool of the Moreau-Yosida regularization to smooth the function, then make using of the approximate values of the function F and its gradient g instead of their exact values in WYL conjugate gradient algorithm.

We first recall the Wei-Yao-Liu conjugate gradient method for unconstrained optimization problem:

$$\min\{f(x)|x \in R^n\}, \tag{18}$$

where $f : R^n \to R$ is continuously differentiable and its gradient g is available. The Wei-Yao-Liu conjugate gradient method [25] is defined by

$$x_{k+1} = x_k + \alpha_k d_k, \tag{19}$$

where x_k is the current iteration point, $\alpha_k > 0$ is the steplength, and d_k is the search direction determined by

$$d_{k+1} = \begin{cases} -g_{k+1}, & \text{if } k = 0, \\ -g_{k+1} + \beta_{k+1}^{WYL} d_k, & \text{if } k \geq 1, \end{cases} \tag{20}$$

where $g_{k+1} = \nabla f(x_{k+1})$ is the gradient vector of $f(x)$ at the point x_{k+1}, and

$$\beta_{k+1}^{WYL} = \frac{g_{k+1}^T (g_{k+1} - \frac{\|g_{k+1}\|}{\|g_k\|} g_k)}{g_k^T g_k}. \tag{21}$$

Now we state the steps of the algorithm as follows.

Algorithm 3.1. Wei-Yao-Liu Conjugate Gradient (WYLCG) Algorithm.
Step 0: Given $x_0 \in R^n$, $\sigma \in (0, 1)$, $\lambda > 0$, $\rho \in [0, 1]$, $\xi \in (0, 1)$, $E_0 = 1$ and a strictly decreasing positive sequence $\{\tau_k\}$ such that $\tau_0 \leq 1$ and $\lim_{k \to 0} \tau_k = 0$.

Step 1: Set $k = 0$. Let $\varepsilon_0 = \tau_0$. Compute $p^a(x_0, \varepsilon_0)$, $F^a(x_0, \varepsilon_0)$ and $g^a(x_0, \varepsilon_0)$ by (14), (12) and (13), respectively. Let $J_0 = F^a(x_0, \varepsilon_0)$ and $d_0 = -g^a(x_0, \varepsilon_0)$.

Step 2: If $\|g^a(x_k, \varepsilon_k)\| = 0$, stop. Otherwise, go to Step 3.

Step 3: Choose a scalar ε_{k+1} satisfying $0 < \varepsilon_{k+1} \le min\{\tau_k, \tau_k\|g^a(x_k, \varepsilon_k)\|^2\}$, and compute the step size α_k by the following nonmonotone Armijo-type line search:

$$F^a(x_k + \alpha_k d_k, \varepsilon_{k+1}) - J_k \le \sigma \alpha_k g^a(x_k, \varepsilon_k)^T d_k, \tag{22}$$

where $\alpha_k = s_k 2^{-i_k}$, $s_k = \frac{1-\xi}{2L} \frac{\|g^a(x_k, \varepsilon_k)\|^2}{\|d_k\|^2}$ and i_k is the smallest nonnegative integer such that (22) holds.

Step 4: Let $x_{k+1} = x_k + \alpha_k d_k$. Stop if $\|g^a(x_{k+1}, \varepsilon_{k+1})\| = 0$.

Step 5: Update J_{k+1} by the following formula

$$E_{k+1} = \rho E_k + 1, J_{k+1} = \frac{\rho E_k J_k + F^a(x_k + \alpha_k d_k, \varepsilon_{k+1})}{E_{k+1}}. \tag{23}$$

Step 6: Compute the search direction d_{k+1} by (20) with g_k and g_{k+1} replaced by $g^a(x_k, \varepsilon_k)$ and $g^a(x_{k+1}, \varepsilon_{k+1})$, respectively.

Step 7: Set $k := k + 1$ and go to Step 2.

Remarks.

(i) The definition of $0 < \varepsilon_{k+1} \le min\{\tau_k, \tau_k\|g^a(x_k, \varepsilon_k)\|^2\}$ in Algorithm 3.1, together with (13) and (14) deduce that

$$\varepsilon_{k+1} = o(\|g^a(x_k, \varepsilon_k)\|^2) = o(\|x_k - p^a(x_k, \varepsilon_k)\|^2) = o(\|x_k - x_{k+1}\|^2) = o(\alpha_k^2\|d_k\|^2), \tag{24}$$

then with the decreasing property of ε_{k+1}, the assumed condition $\varepsilon_k = o(\alpha_k^2\|d_k\|^2)$ in Lemma 3.4 holds.

(ii)The line search technique (22) is motivated by Zhang and Hager [34]. It is not difficult to see that J_{k+1} is a convex combination of J_k and $F^a(x_{k+1}, \varepsilon_{k+1})$. Noticing $J_0 = F^a(x_0, \varepsilon_0)$, it follows that J_k is a convex combination of the function values $F^a(x_0, \varepsilon_0)$, $F^a(x_1, \varepsilon_1)$, $\cdots$, $F^a(x_k, \varepsilon_k)$. The choice of ρ controls the degree of nonmonotonicity. If $\rho = 0$, then the line search is the usual monotone Armijo line search. If $\rho = 1$, then $J_k = C_k$, where

$$C_k = \frac{1}{k+1} \sum_{i=0}^{k} F^a(x_i, \varepsilon_i)$$

is the average function value.

(iii)Realizing the parameter $s_k = \frac{1-\xi}{2L} \frac{\|g^a(x_k, \varepsilon_k)\|^2}{\|d_k\|^2}$ is not a difficulty in practice, despite the fact that the constant L is not known. For example, we can set $L \simeq \frac{\|g^a(x_k, \varepsilon_k) - g^a(x_{k-1}, \varepsilon_{k-1})\|}{\|x_k - x_{k-1}\|}$.

We need the following assumptions which are given in papers [13, 29, 30, 32].

Assumption A. (i) The sequence $\{V_k\}$ is bounded, i.e., there exists a positive constant M such that

$$\|V_k\| \le M, \forall k. \tag{25}$$

where the matrix $V_k \in \partial_B g(x_k)$.

(ii) F is bounded from below.

The following lemma shows that Algorithm 3.1 satisfy the sufficient descent property.

Lemma 3.1

(Corollary 2.1. [18]) Let Assumption A holds and the sequence $\{x_k, d_k\}$ be generated by Algorithm 3.1. Then we have

$$d_k^T g^a(x_k, \varepsilon_k) \le -\xi\|g^a(x_k, \varepsilon_k)\|^2, \tag{26}$$

Lemma 3.2

Let Assumption A holds and the sequence $\{x_k, d_k\}$ be generated by Algorithm 3.1. Then

$$\|d_k\| \le (2 - \xi)\|g^a(x_k, \varepsilon_k)\|. \tag{27}$$

Proof

When $k = 0$, (27) is true since $d_0 = -g^a(x_0, \varepsilon_0)$ and $\xi \in (0, 1)$. Suppose that (27) holds for a certain $k \geq 0$. By the definition of d_{k+1}, Assumption A and $\alpha_k \leq \frac{1-\xi}{2L} \frac{\|g^a(x_k, \varepsilon_k)\|^2}{\|d_k\|^2}$, we have

$$
\begin{aligned}
&\|d_{k+1}\| - \|g^a(x_{k+1}, \varepsilon_{k+1})\| \\
&\leq \|d_{k+1} + g^a(x_{k+1}, \varepsilon_{k+1})\| = \|\beta_{k+1} d_k\| \\
&\leq \frac{\|g^a(x_{k+1}, \varepsilon_{k+1})\| \|g^a(x_{k+1}, \varepsilon_{k+1}) - g^a(x_k, \varepsilon_k) + g^a(x_k, \varepsilon_k) - \frac{\|g^a(x_{k+1}, \varepsilon_{k+1})\|}{\|g^a(x_k, \varepsilon_k)\|} g^a(x_k, \varepsilon_k)\|}{\|g^a(x_k, \varepsilon_k)\|^2} \|d_k\| \\
&\leq \frac{2\|g^a(x_{k+1}, \varepsilon_{k+1})\| \|d_k\| \|g^a(x_{k+1}, \varepsilon_{k+1}) - g^a(x_k, \varepsilon_k)\|}{\|g^a(x_k, \varepsilon_k)\|^2} \leq \frac{2\|g^a(x_{k+1}, \varepsilon_{k+1})\| \|d_k\| L\alpha_k \|d_k\|}{\|g^a(x_k, \varepsilon_k)\|^2} \\
&\leq \frac{2\|g^a(x_{k+1}, \varepsilon_{k+1})\| \|d_k\| L \frac{1-\xi}{2L} \frac{\|g^a(x_k, \varepsilon_k)\|^2}{\|d_k\|^2} \|d_k\|}{\|g^a(x_k, \varepsilon_k)\|^2} \\
&= (1 - \xi)\|g^a(x_{k+1}, \varepsilon_{k+1})\|,
\end{aligned}
$$

we further obtain $\|d_{k+1}\| \leq (2 - \xi)\|g^a(x_{k+1}, \varepsilon_{k+1})\|$. Thus (27) holds for all $k \geq 0$. $\qquad\square$

Using (26), (27) and Assumption A, similar to Lemma 1.1 in [34], it shows that Algorithm 3.1 is well-defined in the following lemma. For the proof is essentially the same as Lemma 1.1 in [34], we omit its proof here.

Lemma 3.3

Suppose that Assumption A holds. Then, for the iteration generated by Algorithm 3.1, we have $F^a(x_k, \varepsilon_k) \leq J_k \leq C_k$ for each k, where $C_k = \frac{1}{k+1} \sum_{i=0}^k F^a(x_i, \varepsilon_i)$. Moreover, there exists α_k satisfying Armijo conditions of the line search update.

Lemma 3.4

Suppose that Assumption A holds. Let $\{(x_k, \varepsilon_k)\}$ be the sequence generated by Algorithm 3.1. Suppose that $\varepsilon_k = o(\alpha_k^2 \|d_k\|^2)$ holds. Then, there exists a constant $m_0 > 0$ such that

$$\alpha_k \geq m_0. \tag{28}$$

Proof

Let α_k satisfies the nonmonotone Armijo-type line search (22). We proceed by the method of contradiction and suppose that $\liminf_{k \to \infty} \alpha_k = 0$. By passing to a subsequence if necessary, we may assume that $\alpha_k \to 0$. Then, by the line search, $\alpha_k' = \alpha_k/2$ satisfies

$$\sigma \alpha_k' g^a(x_k, \varepsilon_k)^T d_k < F^a(x_k + \alpha_k' d_k, \varepsilon_{k+1}) - J_k.$$

Combining this with $F^a(x_k, \varepsilon_k) \leq J_k \leq C_k$ in Lemma 3.3, Proposition 2.5 and Taylor's formula, we have

$$
\begin{aligned}
\sigma \alpha_k' g^a(x_k, \varepsilon_k)^T d_k &< F^a(x_k + \alpha_k' d_k, \varepsilon_{k+1}) - F^a(x_k, \varepsilon_k) \\
&\leq F(x_k + \alpha_k' d_k) - F(x_k) + \varepsilon_{k+1} \\
&= \alpha_k' d_k^T g(x_k) + \frac{1}{2}(\alpha_k')^2 d_k^T V(u_k) d_k + \varepsilon_{k+1} \\
&\leq \alpha_k' d_k^T g(x_k) + \frac{M}{2}(\alpha_k')^2 \|d_k\|^2 + \varepsilon_{k+1}, \tag{29}
\end{aligned}
$$

where $u_k \in (x_k, x_{k+1})$. It follows from (29) that

$$
\begin{aligned}
\frac{\alpha_k}{2} = \alpha'_k &> [\frac{(g^a(x_k, \varepsilon_k) - g(x_k))^T d_k - (1-\sigma)g^a(x_k, \varepsilon_k)^T d_k - \varepsilon_{k+1}/\alpha'_k}{\|d_k\|^2}]\frac{2}{M} \\
&\geq [\frac{\xi(1-\sigma)\|g^a(x_k, \varepsilon_k)\|^2 - \sqrt{2\varepsilon_k/\lambda}\|d_k\| - \varepsilon_k/\alpha'_k}{\|d_k\|^2}]\frac{2}{M} \\
&= [\frac{\xi(1-\sigma)\|g^a(x_k, \varepsilon_k)\|^2}{\|d_k\|^2} - o(\alpha_k)/\sqrt{\lambda} - o(\alpha_k)]\frac{2}{M} \\
&\geq [\frac{\xi(1-\sigma)}{(2-\xi)^2} - o(\alpha_k)/\sqrt{\lambda} - o(\alpha_k)]\frac{2}{M},
\end{aligned}
\tag{30}
$$

where the equality follows from $\varepsilon_k = o(\alpha_k^2\|d_k\|^2)$, the second inequality follows from (26), Proposition 2.5(iii) and $\varepsilon_{k+1} \leq \varepsilon_k$, and the last inequality follows (27). Dividing both sides by α_k and passing to limit, we see that

$$
\frac{1}{2} \geq \lim_{k\to\infty} (\frac{2\xi(1-\sigma)}{(2-\xi)^2 M})\frac{1}{\alpha_k} = +\infty,
$$

which is impossible, so the conclusion is obtained. $\qquad\square$

The following theorem establishes the global convergence of Algorithm 3.1.

Theorem 3.1
Let $\{x_k\}$ be generated by Algorithm 3.1 and suppose that the conditions in Lemma 3.4 hold. Then, $\lim_{k\to\infty}\|g(x_k)\| = 0$, and any accumulation point of $\{x_k\}$ is an optimal solution of (1).

Proof
In order to complete this proof, we first show that

$$
\lim_{k\to\infty}\|g^\alpha(x_k, \varepsilon_k)\| = 0.
\tag{31}
$$

Suppose that (31) is not true, then there exists constants $\epsilon_0 > 0$ and $k_0 > 0$ such that

$$
\|g^\alpha(x_k, \varepsilon_k)\| \geq \epsilon_0, \ \forall k > k_0.
\tag{32}
$$

Since F is bounded from below by Assumption A(ii) and $F(x_k) \leq F^a(x_k, \varepsilon_k)$ for all k by Proposition 2.5, we see that $F^a(x_k, \varepsilon_k)$ is bounded from below. Together with $F^a(x_k, \varepsilon_k) \leq J_k$ for all k by Lemma 3.3, it shows that J_k is also bounded from below, and

$$
\sum_{k>k_0} (J_k - J_{k+1}) < \infty.
\tag{33}
$$

On the other hand, by (22), (26) and (28), we have

$$
\begin{aligned}
J_k - F^a(x_{k+1}, \varepsilon_{k+1}) &\geq -\sigma\alpha_k g^a(x_k, \varepsilon_k)^T d_k = \sigma\alpha_k\|g^a(x_k, \varepsilon_k)\|^2 \\
&\geq \sigma m_0 \epsilon_0, \quad \forall k > k_0.
\end{aligned}
$$

Therefore, it follows from the above inequality and (23) that

$$
\begin{aligned}
\sum_{k>k_0} (J_k - J_{k+1}) &= \sum_{k>k_0} \frac{E_{k+1}J_k - \rho E_k J_k - F^a(x_{k+1}, \varepsilon_{k+1})}{E_{k+1}} \\
&= \sum_{k>k_0} \frac{J_k - F^a(x_{k+1}, \varepsilon_{k+1})}{1 + \rho + \cdots + \rho^{k+1}} \\
&\geq \sum_{k>k_0} \frac{\sigma m_0 \epsilon_0}{k+2} = +\infty,
\end{aligned}
$$

this is a contradiction with (33). Thus, (31) holds. By Proposition 2.5(iii), we obtain

$$\|g^\alpha(x_k, \varepsilon_k) - g(x_k)\| \leq \sqrt{\frac{2\varepsilon_k}{\lambda}}.$$

Since $\varepsilon_k \to 0$ by the construction of Algorithm 3.1, thus

$$\lim_{k \to \infty} \|g(x_k)\| = 0. \tag{34}$$

Let x^* be an accumulation point of $\{x_k\}$, without loss of generality, there exists a subsequence $\{x_k\}_K$ satisfying

$$\lim_{k \in K, \, k \to \infty} x_k = x^*. \tag{35}$$

From properties of $F(x)$, we have $g(x_k) = (x_k - p(x_k))/\lambda$. Then, by (34) and (35), $x^* = p(x^*)$ holds. Therefore x^* is an optimal solution of (1). The proof is complete.   □

4. Numerical Results

In this section, we perform numerical experiments to test the performance of the given algorithm, then compare it with the MPRP gradient method in [29] and the proximal bundle method (PBL) in [15]. All the nonsmooth problems of Table 1 can be found in [14]. Table 1 contains problem names and optimum function values. The codes were written in MATLAB R2010a and run on a personal computer with an Intel Core 2 Duo CPU at 2.8 GHz and 2 GB of memory. We set $\rho = 0.75$, $\sigma = 0.9$ and adopt the termination condition $\|g^a(x, \varepsilon)\| \leq 10^{-5}$. The quality of the objective function value at termination $\bar{f}$ is measured by the relative error to f_{ops}, i.e.,

$$RelErr = \frac{|\bar{f} - f_{ops}|}{|f_{ops}|}.$$

The subproblem (14) involves the finding of a vector $p^a(x_k, \varepsilon_k)$ for given x_k and ε_k. We use the Nelder-Mead simplex program solver fminsearch.m from the Matlab optimization toolbox to perform the subproblem (14), this subalgorithm stops if the maximum coordinate difference between the current best point and the other points in the simplex is less than or equal to 10^{-3}, and the corresponding difference in function values is less than or equal to 10^{-3}. The subalgorithm will also stop if the maximum number of iterations or function evaluations is larger than two hundred.

Table 1. Test problems.

Nr.	Problems	$f_{ops}(x)$	Nr.	Problems	$f_{ops}(x)$
1	Rosenbrock	0	6	QL	7.20
2	Crescent	0	7	LQ	-1.4142136
3	CB2	1.9522245	8	Mifflin 1	-1.0
4	CB3	2.0	9	Mifflin 2	-1.0
5	DEM	-3	10	Wolfe	-8.0

Firstly, we give some insight into the behavior of WYL conjugate gradient algorithm with different approximation parameter ε. In this test, we fixed $\lambda = 5$. The results are listed in Table 2, which contains the number of the iterations (NI), the number of the function evaluations (NF) and the relative error (RelErr). Viewing from the table, we conclude that the proposed algorithm works reasonably well for all the test cases. This table also illustrates that the effectiveness of the algorithm is improved while approximation parameter ε is small.

Table 2. Results on Problems 1-10 with different ε.

Problems	$\tau_k = \frac{1}{(k+2)^2}$		$\tau_k = \frac{1}{3(k+2)^3}$		$\tau_k = \frac{1}{4(k+2)^4}$		$\tau_k = \frac{1}{5(k+2)^5}$	
Nr.	NI/NF	RelErr	NI/NF	RelErr	NI/NF	RelErr	NI/NF	RelErr
1	10/13	3.8999e-10	10/13	2.6751e-10	10/13	5.0712e-10	11/14	1.1132e-10
2	7/10	3.3677e-06	17/20	3.1140e-06	19/22	3.5430e-06	14/17	3.4125e-06
3	8/11	5.1223e-09	9/12	7.1713e-09	4/7	9.4763e-08	6/9	1.1781e-08
4	3/5	2.2614e-05	3/5	1.0623e-05	3/5	1.0623e-05	3/5	1.0623e-05
5	4/7	5.3045e-05	4/7	1.9484e-05	4/7	1.9484e-05	4/7	1.9484e-05
6	6/9	1.6666e-09	6/9	1.8056e-09	7/10	8.3333e-10	6/9	8.3333e-10
7	7/10	3.2526e-08	6/9	1.3081e-07	4/7	6.7882e-08	4/7	6.7882e-08
8	4/10	8.1831e-05	6/8	2.9520e-06	5/7	8.9700e-07	6/8	6.7160e-06
9	7/10	4.5090e-06	6/9	6.8229e-06	5/8	4.5530e-06	6/9	6.8330e-06
10	5/8	6.5374e-08	4/7	1.2148e-06	4/7	1.5201e-06	4/7	4.2737e-07

Secondly, to specifically illustrate the performance of WYL conjugate gradient algorithm, we present two test results on Problems 1-10 in terms of number of iterations and the optimum function values error as the regularization parameter λ varies from 1 to 10 in Figure 1. In this test, we fixed $\tau_k = \frac{1}{(k+2)^2}$. From Figure 1, we can see that Problems 2, 3 and 8 are more sensitive to the regularization parameter.
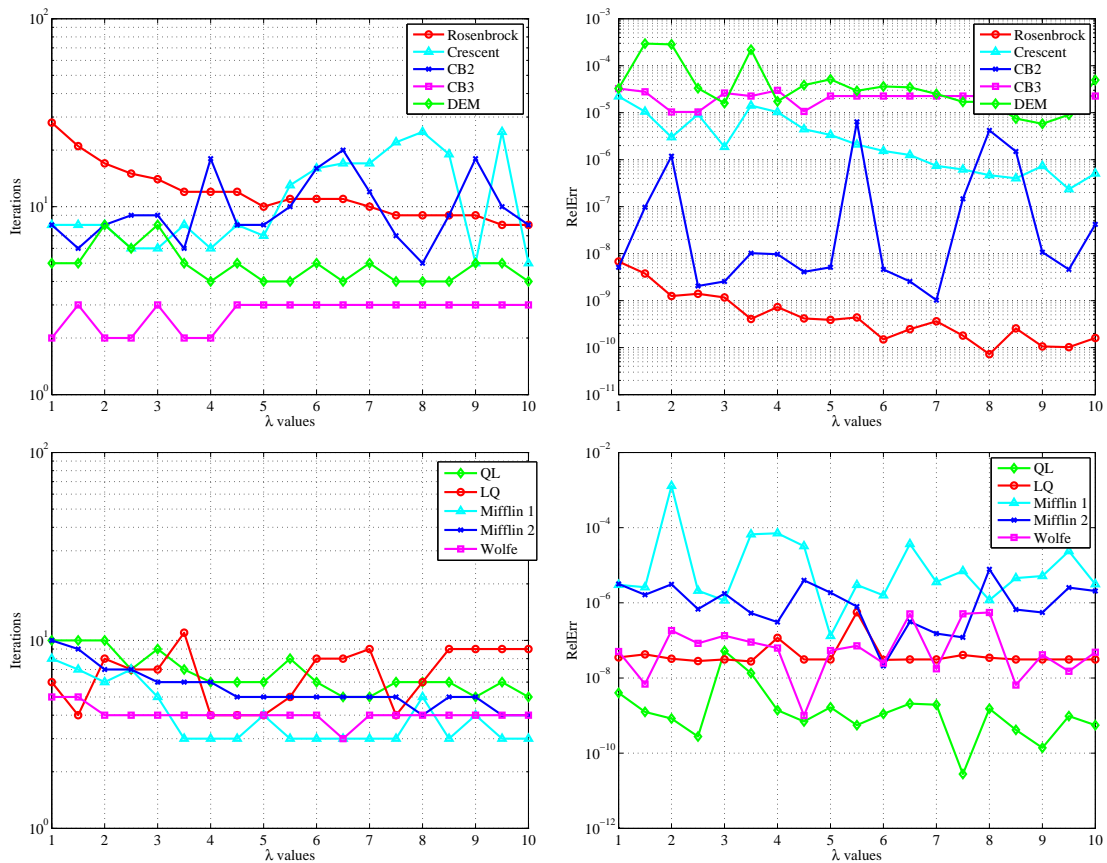

Figure 1. Convergence performance of WYL algorithm: number of iterations (left)
and relative error (right) versus the regularization parameter λ values.

Finally, we compared the performance of the proposed algorithm to the algorithm MPRP and PBL. In the test, we fixed $\tau_k = \frac{1}{5(k+2)^5}$. As we can see from Figure 1 that Problems 1-10 have different sensitiveness on the regularization parameter, we set $\lambda = 7$ for Problem 3 and 5, $\lambda = 2$ for Problems 7 and 9, $\lambda = 1$ for Problems 4 and 10, $\lambda = 10$ for other Problems. We present three comparison results in terms of number of iteration, number of function evaluations and the final objective function value ($f(x)$) in Table 3. From the numerical results in Table 3, we can conclude that these three methods are effective for nonsmooth optimization problems, and WYL

conjugate gradient algorithm performs best of all the methods in term of the iteration number, the number of function evaluations and the final objective function value.

Table 3. Numerical results for WYL/MPRP/PBL on Problems 1-10.

Problems	WYL		MPRP		PBL	
Nr.	NI/NF	f(x)	NI/NF	f(x)	NI/NF	f(x)
1	8/11	4.5570e-11	46/48	7.0918e-7	42/45	0.381e-6
2	4/7	4.9390e-6	11/13	6.7351e-5	18/20	0.679e-6
3	5/8	1.9522245	12/14	1.952225	32/34	1.9522245
4	3/5	2.000020	2/6	2.000098	14/16	2.0000000
5	4/7	-2.999952	4/6	-2.999866	17/19	-3.0000000
6	5/8	7.2000000	10/12	7.200011	13/15	7.2000015
7	6/8	-1.4142136	2/3	-1.414214	11/12	-1.4142136
8	4/7	-0.9999972	4/6	-0.9919815	66/68	-0.9999994
9	7/9	-0.9999983	20/23	-0.9999925	13/15	-1.0000000
10	8/10	-7.9999999	-	-	43/46	-8.0000000

5. Conclusions

By making use of the Moreau-Yosida regularization, a nonmonotone line search technique of [34] and a new formula in [25] developed by the authors earlier, we presented a Wei-Yao-Liu conjugate gradient algorithm for solving nonsmooth convex optimization problems. Our algorithm satisfies the sufficiently descent property, and the corresponding search direction belongs to a trust region automatically. The global convergence of the given algorithm was established under suitable conditions and the effectiveness of the algorithm can be observed from the result of numerical experiments.

Acknowledgement

REFERENCES

1. J.R. Birge, L. Qi and Z. Wei, *Convergence analysis of some methods for minimizing a nonsmooth convex function*, Journal of Optimization Theory and Applications, vol. 97, pp. 357-383, 1998.
2. J.R. Birge, L. Qi and Z. Wei, *A general approach to convergence properties of some methods for nonsmooth convex optimization*, Applied Mathematics and Optimization, vol. 38, pp. 141-158, 1998.
3. J.F. Bonnans, J.C. Gilbert, C. Lemaréchal and C.A. Sagastizábal, *A family of veriable metric proximal methods*, Mathematical Programming, vol. 68, pp. 15-47, 1995.
4. F.H. Clark, *Optimization and Nonsmooth Analysis*, New York, Wiley, 1983.
5. A.R. Conn, N.I.M. Gould and P.L. Toint, *Trust-Region Methods*, SIAM, Philadelphia, 2000.
6. R. Correa and C. Lemaréchal, *Convergence of some algorithms for convex minimization*, Mathematical Programming, vol. 62, pp. 261-273, 1993.
7. M. Fukushima, *A descent algorithm for nonsmooth convex optimization*, Mathematical Programming, vol, 30, pp. 163-175, 1984.
8. M. Fukushima and L. Qi, *A global and superlinearly convergent algorithm for nonsmooth convex minimization*, SIAM Journal on Optimization, vol. 6, pp. 1106-1120, 1996.
9. J.C. Gilbert and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM Journal of Optimization, vol. 2, pp. 21-42, 1992.
10. J.B. Hiriart-Urruty and C. Lemmaréchal, *Convex Analysis and Minimization Algorithms*, Berlin, Springer, 1993.
11. H. Huang, Z. Wei and S. Yao, *The proof of the sufficient descent condition of the Wei-Yao-Liu conjugate gradient method under the strong Wolfe-Powell line search*, Applied Mathematics and Computation, vol. 189, pp. 1241-1245, 2007.
12. Y. Hu and Z. Wei, *Wei-Yao-Liu conjugate gradient projection algorithm for nonlinear monotone equations with convex constraints*, International Journal of Computer Mathematics, vol. 92, pp. 2261-2272, 2015.
13. S. Lu, Z. Wei and L. Li, *A trust region algorithm with adaptive cubic regularization methods for nonsmooth convex minimization*, Computational Optimization and Applications, vol. 51, pp. 551-573, 2012.
14. L. Lukšan and J. Vlček, *Test Problems for Nonsmooth Unconstrained and Linearly Constrained Optimization*, Technical Report No.798, Institute of Computer Science, Academy of Sciences of the Czech Republic, 2000.

15. L. Lukšan and J. Vlček, *A bundle-Newton method for nonsmooth unconstrained minimization*, Mathematical Programming, vol. 83, pp. 373-391, 1998.
16. Q. Li, *Conjugate gradient type methods for the nondifferentiable convex minimization*, Optimization Letters, vol. 7, pp. 533-545, 2013.
17. Q. Li, *A Modified Fletcher-Reeves-Type Method for Nonsmooth Convex Minimization*, Statistics, Optimization & Information Computing, vol. 2, no. 3, pp. 200-210, 2014.
18. S. Lu, Z. Wei and L. Mo, *Some global convergence properties of the Wei-Yao-Liu conjugate gradient method with inexact line search*, Applied Mathematics and Optimization, vol. 217, pp. 7132-7137, 2011.
19. N. H. Monjezi, S. Nobakhtian, *A new infeasible proximal bundle algorithm for nonsmooth nonconvex constrained optimization*, Computational Optimization and Applications, vol. 74, pp. 443-480, 2019.
20. L. Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Mathematics of Operations Research, vol. 18, pp. 227-244, 1993.
21. L. Qi and J. Sun, *A nonsmooth version of Newtoń́s method*, Mathematical Programming, vol. 58, pp. 353-367, 1993.
22. A.I. Rauf and M. Fukushima, *Globally convergent BFGS method for nonsmooth convex optimization*, Journal of Optimization Theory and Applications, vol. 104, pp. 539-558, 2000.
23. Z. Wei and L. Qi, *Convergence analysis of a proximal Newton method*, Numerical Functional Analysis and Optimization, vol. 17, pp. 463-472, 1996.
24. Z. Wei, L. Qi and J.R. Birge, *A new methods for nonsmooth convex optimization*, Journal of Inequalities and Applications, vol. 2, pp. 157-179, 1998.
25. Z. Wei, S. Yao and L. Liu, *The convergence properties of some new conjugate gradient methods*, Applied Mathematics and Computation, vol. 183, pp. 1341-1350, 2006.
26. S. Yao, Z Wei and H. Huang, *A note about WYLs conjugate gradient method and its applications*, Applied Mathematics and Computation, vol. 191, pp. 381-388, 2007.
27. Z. Sheng and G. Yuan, *An effective adaptive trust region algorithm for nonsmooth minimization*, Computational Optimization and Applications, vol. 71, pp. 251-271, 2018.
28. G. Yuan, Z. Meng and Y. Li, *A modified Hestenes and Stiefel conjugate gradient algorithm for large-scale nonsmooth minimizations and nonlinear equations*, Journal of Optimization Theory and Applications, vol. 168, pp. 129-152, 2016.
29. G. Yuan, Z. Wei and G. Li, *A modified Polak-Ribière-Polyak conjugate gradient algorithm for nonsmooth convex programs*, Journal of Computational and Applied Mathematics, vol. 255, pp. 86-96, 2014.
30. G. Yuan, Z. Wei and Z. Wang, *Gradient trust region algorithm with limited memory BFGS update for nonsmooth convex minimization*, Computational Optimization and Applications, vol. 54, pp. 45-64, 2013.
31. G. Yuan and M. Zhang, *A modified Hestens-Stiefel conjugate gradient algorithm for large-scale optimization*, Numerical Functional Analysis and Optimization, vol. 34, pp. 914-937, 2013.
32. G. Yuan and Z. Wei, *The Barzilai and Borwein Gradient Method with Nonmonotone Line Search for Nonsmooth Convex Optimization Problems*, Mathematical Modelling and Analysis, vol. 17, pp. 203-216, 2012.
33. L. Zhang, *A new trust region algorithm for nonsmooth convex minimization*, Applied Mathematics and Computation, vol. 193, pp. 135-142, 2007.
34. H. Zhang and W. Hager, *A nonmonotone line search technique and its application to unconstrained optimization*, SIAM Journal on Optimization, vol. 14, pp. 1043-1056, 2004.