*ICIC Express Letters*
Volume **10**, Number **12**, December **2016**

ICIC International ©2016 ISSN 1881-803X
pp. **2921–2925**

# A MODIFIED NLS-DY CONJUGATE GRADIENT METHOD

Rui Shan[1], Changxia Qi[1], Jingyi Zhao[2,3] and Yumin Xu[1]

[1]School of Science
[2]Hebei Provincial Key Laboratory of Heavy Machinery Fluid Power Transmission and Control
[3]Key Laboratory of Advanced Forging and Stamping Technology and Science
Yanshan University
No. 438, Hebei Ave., Qinhuangdao 066004, P. R. China
m18712791869@163.com; zjy@ysu.edu.cn

Received May 2016; accepted August 2016

Abstract. *This paper proposes a modified NLS-DY conjugate gradient method with disturbance parameters. And under the strong Wolfe line search, the sufficient descent condition and the global convergence are established. At last, the results of numerical experiments demonstrate the effectiveness of the presented algorithm.*

**Keywords:** Conjugate gradient method, Global convergence, Strong Wolfe line search, Disturbance parameters

1. **Introduction.** Consider the following unconstrained optimization problem (UP):

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

where $x \in R^n$ is a real vector with $n \geq 1$ component, $f : R^n \to R$ is a smooth function and its gradient $g$ is available. Conjugate gradient method (CGM) is widely used to solve UP, especially when the dimension $n$ is large. The detailed CGM is as follows:

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

where $x_1$ is a given initial point and $\alpha_k$ is the step length along $d_k$. The search direction $d_k$ is generated by

$$d_k = \begin{cases} -g_1, & k = 1 \\ -g_k + \beta_k d_{k-1}, & k \geq 2 \end{cases} \tag{3}$$

where $g_k = \nabla f(x_k)$ is the gradient of $f(x)$ at $x_k$ and $\beta_k$ is an important parameter depending on $x_{k-1}$ and $x_k$. In respect to different formulas of $\beta_k$, there are distinct conjugate gradient methods (CGMs) including Fletcher-Reeves method (FR), Polak-Ribiere-Polyak method (PRP), Dai-Yuan method (DY), Liu-Storey method (LS), Hestenes-Stiefel method (HS), etc. Over the years, improved algorithms and many variants of hybrid methods based on them have been proposed. For instance, there are some hybrid CGMs [1-4]. In [5,6], some improved algorithms for a kind of CGM are presented. In [7], Wu and Du raised a modified CGM with disturbance factors. And in [8], Shi and Shan presented the NLS-DY algorithm with disturbance factors, which is based on LS and DY methods, where the parameter $\beta_k$ is yielded by

$$\beta_k = \begin{cases} \dfrac{g_k^T (g_k - g_{k-1})}{\left(g_k^T d_{k-1}\right)^2 - d_{k-1}^T g_{k-1}}, & \text{if } (1 - \cos\theta)\|g_k\|^2 > \left|g_k^T g_{k-1}\right| \\ \max\left\{\dfrac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}, 0\right\}, & \text{others} \end{cases} \tag{4}$$

Introducing the disturbance factor $\theta$, on the premise of guaranteeing $\beta_k \geq 0$, can reduce unnecessary calculating steps and improve convergence rate. In addition, the NLS-DY method makes full use of the strong convergence property of the DY method.

In [9], Fan proposed a new CGM with double parameters to improve the deficiency of traditional CGMs. And Shan and Liu [10] presented an algorithm with three parameters. A remarkable feature of the two methods is that the values of parameters are adjustable to guarantee $\beta_k > 0$. Thus the algorithms are more flexible in practical computation.

Based on [8] and inspired by [9] and [10], we give a new algorithm with new parameter $\beta_k$, and denote it by mLS-DY method, as follows:

$$\beta_k = \begin{cases} \dfrac{g_k^T (g_k - g_{k-1})}{u\,|g_k^T d_{k-1}| - d_{k-1}^T g_{k-1}}, & \text{if } (1-\cos\theta)\,\|g_k\|^2 > \left|g_k^T g_{k-1}\right| \\ \max\left\{ \dfrac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}, 0 \right\}, & \text{others} \end{cases} \tag{5}$$

where $\|\cdot\|$ is the 2-norm. $\theta$ satisfies $\theta \in (0, \pi/2)$. For $\beta_k > 0$, the parameter $u > 0$.

The search direction $d_k$ satisfies sufficient descent condition [11], that is

$$g_k^T d_k \leq -m\,\|g_k\|^2, \ m > 0 \tag{6}$$

And inspired by [8], the search direction $d_k$ of our algorithm is:

$$d_k = \begin{cases} -g_k, & k = 1 \\ -\theta_k g_k + \beta_k d_{k-1} = -\left(1 + \beta_k \dfrac{g_k^T d_{k-1}}{\|g_k\|^2}\right) g_k + \beta_k d_{k-1}, & k \geq 2 \end{cases} \tag{7}$$

The characteristic of our algorithm, compared with the NLS-DY method, is introducing a parameter $u$. You can assign $u$ different values to solve optimization problems of various functions. Furthermore, an appropriate value for the parameter $u$ can generate a promising computational performance. Thus, adding $u$ can not only enlarge the using scope but also improve the efficiency of the mLS-DY method.

The rest of the paper is organized as follows. In Section 2, we present our algorithm. In Section 3, we discuss the sufficient descent property and the global convergence of algorithm under the strong Wolfe line search. In Section 4, numerical results are shown to illustrate the efficiency of the proposed method. Conclusions are stated in the last section.

2. **Algorithm.** We describe our algorithm framework as follows:

Step 1. Give any initial point $x_1 \in R^n$, compute $f_1 = f(x_1)$ and $g_1 = g(x_1)$, set $d_1 = -g_1$, accuracy tolerance $\varepsilon > 0$, and set the counter $k := 1$.

Step 2. If $\|g_1\| \leq \varepsilon$, then stop. Otherwise, go to Step 3.

Step 3. Compute $\alpha_k$ by the strong Wolfe line search, namely

$$\begin{cases} f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \\ \left|g_k^T d_{k-1}\right| \leq -\sigma g_{k-1}^T d_{k-1} \end{cases} \tag{8}$$

where $\delta$ satisfies $0 < \delta < 1/2$, and $\sigma$ satisfies $\delta < \sigma < 1$.

Step 4. Generate the next iteration by $x_{k+1} = x_k + \alpha_k d_k$, and compute $g_{k+1} = g(x_{k+1})$. If $\|g_{k+1}\| \leq \varepsilon$, terminate; otherwise, go to Step 5.

Step 5. Compute $\beta_k$ by (5), and $d_k$ by (7).

Step 6. Let $k := k + 1$, and go to Step 3.

The numbers of iterations, function evaluations, gradient evaluations and time are distinct when the parameter $u$ takes different values. In order to improve the computational efficiency, you can choose an appropriate parameter value.

3. **Sufficient Descent Property and Global Convergence of the mLS-DY Method.** In this section, sufficient descent property and global convergence of the proposed algorithm are proved. And the following assumptions are necessary.

**Assumption 3.1.** *The objective function $f(x)$ is bounded from below on the level, namely, $\wedge = \left\{x \in R^n \middle| f(x) \le f(x_1)\right\}$, where $x_1$ is the initial point.*

**Assumption 3.2.** *Within a neighborhood $\psi$ of the level, $f(x)$ is continuously differentiable, and its gradient $g(x) = \nabla f(x)$ satisfies the Lipschitz condition.*

3.1. **Sufficient descent property.** The following lemma gives Zoutendijk condition [12].

**Lemma 3.1.** *Suppose that $f(x)$ satisfies Assumptions 3.1 and 3.2. If $d_k$ satisfies sufficient descent condition (6), and the step-length $\alpha_k$ satisfies the strong Wolfe line search condition (8), then the Zoutendijk condition is established, that is $\sum_{k=1}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < +\infty$.*

**Theorem 3.1.** *Suppose that Assumptions 3.1 and 3.2 hold. The step-length $\alpha_k$ satisfies the strong Wolfe line search condition (8). Compute $\beta_k$ by (5), and $d_k$ by (7); if $g_k \neq 0$ is found for each $k \ge 1$, then $g_k^T d_k < 0$ for all $k \ge 1$.*

**Proof:** We divide the proof into three following cases.

Case (i) If $\beta_k = \dfrac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}$

$$\begin{aligned} g_k^T d_k &= g_k^T \left[ -\left(1 + \beta_k \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) g_k + \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}} d_{k-1} \right] \\ &= g_k^T \left[ -\left(1 + \frac{g_k^T d_{k-1}}{d_{k-1}^T y_{k-1}}\right) g_k + \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}} d_{k-1} \right] \\ &= -\|g_k\|^2 < 0. \end{aligned}$$

Case (ii) If $\beta_k = 0$ or $g_k^T d_{k-1} = 0$, we obtain $g_k^T d_k = -\|g_k\|^2 < 0$.

Case (iii) If $\beta_k = \frac{g_k^T (g_k - g_{k-1})}{u\left|g_k^T d_{k-1}\right| - d_{k-1}^T g_{k-1}}$, for $k = 1$, we have $g_1^T d_1 = -\|g_1\|^2 < 0$. We assume that $g_{k-1}^T d_{k-1} < 0$ holds for $k-1$ and $k > 2$, and we prove $g_k^T d_k < 0$ for $k$.

$$\begin{aligned} & g_k^T d_k \\ &= -\left(1 + \beta_k \cdot \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) \cdot \|g_k\|^2 + \beta_k \cdot g_k^T d_{k-1} \\ &= -\left(1 + \frac{g_k^T (g_k - g_{k-1})}{u\left|g_k^T d_{k-1}\right| - d_{k-1}^T g_{k-1}} \cdot \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) \|g_k\|^2 + |\beta_k| \cdot \left|g_k^T d_{k-1}\right| \\ &\le -\left(1 + \frac{\|g_k\|^2 + (1-\cos\theta)\|g_k\|^2}{-d_{k-1}^T g_{k-1}} \cdot \frac{g_k^T d_{k-1}}{\|g_k\|^2}\right) \|g_k\|^2 + \left| \frac{g_k^T (g_k - g_{k-1})}{u\left|g_k^T d_{k-1}\right| - d_{k-1}^T g_{k-1}} \right| \cdot \left|g_k^T d_{k-1}\right| \\ &\le -\left[1 + \frac{2-\cos\theta}{-d_{k-1}^T g_{k-1}} \cdot \left(-\sigma d_{k-1}^T g_{k-1}\right)\right] \cdot \|g_k\|^2 + \frac{\|g_k\|^2 + (1-\cos\theta)\|g_k\|^2}{-d_{k-1}^T g_{k-1}} \cdot \left(-\sigma d_{k-1}^T g_{k-1}\right) \\ &= -\|g_k\|^2 < 0 \end{aligned}$$

Therefore, for all $k \ge 1$, $g_k^T d_k < 0$ always holds.

3.2. **Global convergence.**

**Theorem 3.2.** *Suppose that Assumptions 3.1 and 3.2 hold. Let $\{g_k\}$ be generated by algorithm mLS-DY. Then we obtain $\lim\limits_{k\to\infty} \inf \|g_k\| = 0$.*

**Proof:** Suppose that the stated conclusion is not true. Then, there exists a constant $\varepsilon > 0$ such that $\|g_k\|^2 \geq \varepsilon^2$, $\forall k > 0$. For $d_k = -\theta_k g_k + \beta_k d_{k-1}$, we can get

$$\|d_k\|^2 = (\beta_k)^2 \cdot \|d_{k-1}\|^2 - 2\theta_k d_k^T g_k - \theta_k^2 \cdot \|g_k\|^2$$

Case (i) If $\beta_k = \dfrac{g_k^T (g_k - g_{k-1})}{u\,|g_k^T d_{k-1}| - d_{k-1}^T g_{k-1}}$, divided by $\left(g_k^T d_k\right)^2$ and for $g_k^T d_k \leq -\|g_k\|^2$, we obtain

$$\begin{aligned}
\frac{\|d_k\|^2}{(g_k^T d_k)^2} &= (\beta_k)^2 \cdot \frac{\|d_{k-1}\|^2}{(g_k^T d_k)^2} - \frac{2\theta_k}{g_k^T d_k} - \frac{\theta_k^2 \|g_k\|^2}{(g_k^T d_k)^2} \\
&= \left[\frac{g_k^T (g_k - g_{k-1})}{u\,|g_k^T d_{k-1}| - d_{k-1}^T g_{k-1}}\right]^2 \frac{\|d_{k-1}\|^2}{(g_k^T d_k)^2} - \frac{2\theta_k}{g_k^T d_k} - \frac{\theta_k^2 \|g_k\|^2}{(g_k^T d_k)^2} \\
&\leq \frac{(2-\cos\theta)^2 \|g_k\|^4}{\left(d_{k-1}^T g_{k-1}\right)^2} \cdot \frac{\|d_{k-1}\|^2}{\|g_k\|^4} + \frac{2\theta_k}{\|g_k\|^2} - \frac{\theta_k^2 \|g_k\|^2}{\|g_k\|^4} \\
&= \frac{(2-\cos\theta)^2 \|d_{k-1}\|^2}{\left(d_{k-1}^T g_{k-1}\right)^2} - \frac{1}{\|g_k\|^2} \left(\theta_k^2 - 2\theta_k + 1 - 1\right) \\
&= \frac{(2-\cos\theta)^2 \|d_{k-1}\|^2}{\left(d_{k-1}^T g_{k-1}\right)^2} - \frac{(\theta_k - 1)^2}{\|g_k\|^2} + \frac{1}{\|g_k\|^2} \\
&\leq (2-\cos\theta)^2 \frac{\|d_{k-1}\|^2}{\left(d_{k-1}^T g_{k-1}\right)^2} + \frac{1}{\|g_k\|^2}
\end{aligned}$$

Set $M = \max\left\{1, \left[(2-\cos\theta)^2\right]^{k-1}, \left[(2-\cos\theta)^2\right]^{k-2}, \ldots, (2-\cos\theta)^2\right\}$, we get $\frac{\|d_k\|^2}{\left(g_k^T d_k\right)^2} \leq M \sum_{i=1}^{k} \frac{1}{\|g_i\|^2} \leq \frac{Mk}{\varepsilon^2}$, and thus, $\frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} \geq \frac{\varepsilon^2}{Mk}$, $\sum_{k=1}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} \geq \sum_{k=1}^{\infty} \frac{\varepsilon^2}{Mk} = \infty$ which contradicts Lemma 3.1. Therefore, the desired result holds.

Case (ii) If $\beta_k = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}$, contradictions can be obtained using the same method in Case (i), thus denying the original assumption, and Theorem 3.2 is proved.

## 4. **Numerical Experiments.**

In this section, we select some functions to test numerical performance of mLS-DY algorithm by comparing with the NLS-DY method in [8]. All codes run on PC with 1.80 GHz CPU processor, 4.0 GB RAM memory and Windows XP operating system.

Numerical results are presented in Table 1. Dim is the dimension of tested functions. NI/NF/NG/T denotes the total numbers of iterations, function evaluations, gradient evaluations and CPU time in seconds, respectively. The values of parameters are as follows: $\delta = 0.01$, $\sigma = 0.85$, $\varepsilon = 10^{-6}$, $\theta = \arccos(1/3)$, $u = 9$.

From the table, we can find that, in terms of numbers of iteration, objective function value calculation, objective function gradient value calculation and time, our new algorithm is more efficient.

Table 1. Analysis of the numerical results

| Problem | Dim | mLS-DY (NI/NF/NG/T) | NLS-DY (NI/NF/NG/T) |
|---|---|---|---|
| Rosenbrock | 2 | 30/51/36/1.6055 | 40/70/50/2.3262 |
| Freudenstein & Roth | 6 | 31/50/34/1.3440 | 53/85/59/2.5259 |
| Wood | 4 | 323/472/380/19.9956 | 405/577/472/24.1972 |

5. **Conclusions.** In this paper, we propose mLS-DY method based on NLS-DY method in [8]. Our algorithm guarantees sufficient descent condition and is proved to be globally convergent if it is implemented with the strong Wolfe conditions. Finally, numerical results show that our method is superior on the convergence and numerical performance, which can be used in numerical calculation.

The step-length $\alpha_k$ computed by the strong Wolfe line search requires a large amount of debugging. As further work, it should be worth simplifying the search process to improve the efficiency of optimization.

**Acknowledgment.** This work is partially supported by the National Natural Science Foundation of China (Grant No. 51175448), and Natural Science Foundation of Hebei Province of China (Grant No. E2012203071). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

**REFERENCES**

[1] C. M. Zhang and Y. T. Yang, The global convergence of two mixed conjugate gradient methods, *Journal on Numerical Methods and Computer Applications*, vol.33, no.2, pp.92-98, 2012.
[2] J. K. Liu and S. J. Li, New hybrid conjugate gradient method for unconstrained optimization, *Applied Mathematics and Computation*, vol.245, pp.36-43, 2014.
[3] B. K. Saman and G. Reza, A hybridization of the Polak-Ribiere-Polyak and Fletcher-Reeves conjugate gradient methods, *Numerical Algorithms*, vol.68, no.3, pp.481-495, 2015.
[4] M. Sun and J. Liu, New hybrid conjugate gradient projection method for the convex constrained equations, *Calcolo*, vol.6, no.9, pp.1-13, 2015.
[5] Z. F. Dai, Two modified HS type conjugate gradient methods for unconstrained optimization problems, *Nonlinear Analysis*, vol.74, pp.927-936, 2011.
[6] S. Hiroyuki, A Dai-Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions, *Computational Optimization and Applications*, vol.64, no.1, pp.101-118, 2015.
[7] S. J. Wu and X. W. Du, A modified DL conjugate gradient method with disturbance factors, *Journal of Hubei University for Nationalities*, vol.33, no.4, pp.375-378, 2015.
[8] S. T. Shi and R. Shan, A conjugate gradient method with disturbance factor and its convergence, *Journal of Henan University of Science and Technology: Natural Science*, vol.34, no.6, pp.83-87, 2013.
[9] J. Fan, *The Two Methods of Conjugate Gradient Parameters Estimation of ARMA Model and the Application of the ARIMAX Model*, Master Thesis, Yanshan University, 2010.
[10] R. Shan and Y. N. Liu, Improved conjugate gradient parameter estimation of ARIMA model, *Journal of Henan University of Science and Technology: Natural Science*, vol.36, no.4, pp.85-90, 2015.
[11] L. Zheng, *Research of Several Conjugate Gradient Methods*, Master Thesis, Chongqing University, 2009.
[12] T. F. Zhu and X. Y. Liu, Global convergence of a modified conjugate gradient method, *Journal of Dynamics and Control*, vol.8, no.4, pp.343-344, 2010.