

RESEARCH

Open Access



A family of conjugate gradient methods for large-scale nonlinear equations

Dexiang Feng^{1,2}, Min Sun³ and Xueyong Wang^{3*}

*Correspondence:
yonggk@163.com

³School of Management, Qufu Normal University, Shandong, 276826, China
Full list of author information is available at the end of the article

Abstract

In this paper, we present a family of conjugate gradient projection methods for solving large-scale nonlinear equations. At each iteration, it needs low storage and the subproblem can be easily solved. Compared with the existing solution methods for solving the problem, its global convergence is established without the restriction of the Lipschitz continuity on the underlying mapping. Preliminary numerical results are reported to show the efficiency of the proposed method.

MSC: 90C25; 94A08

Keywords: nonlinear equations; conjugate gradient method; projection method; global convergence

1 Introduction

Consider the following nonlinear equations problem of finding $x \in C$ such that

$$F(x) = 0, \tag{1.1}$$

where $F : R^n \rightarrow R^n$ is a continuous nonlinear mapping and C is a nonempty closed convex set of R^n . The problem finds wide applications in areas such as ballistic trajectory computation and vibration systems [1, 2], the power flow equations [3–5], economic equilibrium problem [6–8], etc.

Generally, there are two categories of solution methods for solving the problem. The first one is first-order methods including the trust region method, the Levenberg-Marquardt method and the projection method. The second one is second-order methods including the Newton method and quasi-Newton method. For the first method, Zhang et al. [9] proposed a spectral gradient method for problem (1.1) with $C = R^n$, and Wang et al. [3] proposed a projection method for problem (1.1). Later, Yu et al. [10] proposed a spectral gradient projection method for constrained nonlinear equations. Compared with the projection method in [3], the methods [9, 10] need the Lipschitz continuity of the underlying mapping $F(\cdot)$, but the former needs to solve a linear equation at each iteration, and its variants [11, 12] also inherit the shortcoming. Different from the above, in this paper, we consider the conjugate gradient method for solving the non-monotone problem (1.1). To this end, we briefly review the well-known conjugate gradient method for the unconstrained

optimization problem

$$\min_{x \in R^n} f(x).$$

The conjugate gradient method generates the sequence of iterates recurrently by

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

where x_k is the current iterate, $\alpha_k > 0$ is the step-size determined by some line search, and d_k is the search direction defined by

$$d_k = \begin{cases} -g_k & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1} & \text{if } k \geq 1, \end{cases}$$

in which $g_k = \nabla f(x_k)$ and $\beta_k > 0$ is a parameter. The famous conjugate gradient methods include the Fletcher-Reeves (FR) method, the Polak-Ribière-Polyak (PRP) method, the Liu-Storey (LS) method, the Dai-Yuan (DY) method. Recently, Sun et al. [13] and Li et al. [14] proposed two variants of PRP method which possesses the following property:

$$|\beta_k| \leq t \frac{\|g_k\|}{\|d_{k-1}\|}, \quad \forall k \geq 1,$$

where $t > 0$ is a constant.

In this paper, motivated by the projection methods in [3, 11] and the conjugate gradient methods in [13, 14], we propose a new family of conjugate gradient projection methods for solving nonlinear problem (1.1). The new designed method is derivative-free as it does not need to compute the Jacobian matrix or its approximation of the underlying function (1.1). Further, the new method does not need to solve any linear equations at each iteration, thus it is suitable to solve large scale problem (1.1).

The remainder of this paper is organized as follows. Section 2 describes the new method and presents its convergence. The numerical results are reported in Section 3. Some concluding remarks are drawn in the last section.

2 Algorithm and convergence analysis

Throughout this paper, we assume that the mapping $F(\cdot)$ is monotone, or more generally pseudo-monotone, on R^n in the sense of Karamardian [15]. That is, it satisfies that

$$\langle F(y), y - x^* \rangle \geq 0, \quad \text{for all } y \in R^n, x^* \in S, \tag{2.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the usual inner product in R^n . Further, we use $P_C(x)$ to denote the projection of point $x \in R^n$ onto the convex set C , which satisfies the following property:

$$\|P_C(x) - P_C(y)\|^2 \leq \|x - y\|^2 - \|P_C(x) - x + y - P_C(y)\|^2, \quad \forall x, y \in R^n. \tag{2.2}$$

Now, we describe the new conjugate gradient projection method for nonlinear constrained equations.

Algorithm 2.1

Step 0. Given an arbitrary initial point $x_0 \in R^n$, parameters $0 < \rho < 1, \sigma > 0, t > 0, \beta > 0, \epsilon > 0$, and set $k := 0$.

Step 1. If $\|F(x_k)\| < \epsilon$, stop; otherwise go to Step 2.

Step 2. Compute

$$d_k = \begin{cases} -F(x_k) & \text{if } k = 0, \\ -(1 + \beta_k \frac{F(x_k)^\top d_{k-1}}{\|F(x_k)\|^2})F(x_k) + \beta_k d_{k-1} & \text{if } k \geq 1, \end{cases} \tag{2.3}$$

where β_k is such that

$$|\beta_k| \leq t \frac{\|F(x_k)\|}{\|d_{k-1}\|}, \quad \forall k \geq 1. \tag{2.4}$$

Step 3. Find the trial point $y_k = x_k + \alpha_k d_k$, where $\alpha_k = \beta \rho^{m_k}$ and m_k is the smallest nonnegative integer m such that

$$-\langle F(x_k + \beta \rho^m d_k), d_k \rangle \geq \sigma \beta \rho^m \|d_k\|^2. \tag{2.5}$$

Step 4. Compute

$$x_{k+1} = P_{H_k} [x_k - \xi_k F(y_k)], \tag{2.6}$$

where

$$H_k = \{x \in R^n | h_k(x) \leq 0\},$$

with

$$h_k(x) = \langle F(y_k), x - y_k \rangle, \tag{2.7}$$

and

$$\xi_k = \frac{\langle F(y_k), x_k - y_k \rangle}{\|F(y_k)\|^2}.$$

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

Obviously, Algorithm 2.1 is different from the methods in [13, 14].

Now, we give some comment on the searching direction d_k defined by (2.3). We claim that it is derived from Schmidt orthogonalization. In fact, in order to make $d_k = -F(x_k) + \beta_k d_{k-1}$ satisfy the property

$$F(x_k)^\top d_k = -\|F(x_k)\|^2, \tag{2.8}$$

we only need to ensure that $\beta_k d_{k-1}$ is vertical to $F(x_k)$. As a matter of fact, by Schmidt orthogonalization, we have

$$d_k = -\left(1 + \beta_k \frac{F(x_k)^\top d_{k-1}}{\|F(x_k)\|^2}\right)F(x_k) + \beta_k d_{k-1}.$$

Equality (2.8) together with the Cauchy-Schwarz inequality implies that $\|d_k\| \geq \|F(x_k)\|$. In addition, by (2.3) and (2.4), we have

$$\begin{aligned} \|d_k\| &\leq \|F(x_k)\| + |\beta_k| \frac{\|F(x_k)\| \|d_{k-1}\|}{\|F(x_k)\|^2} \|F(x_k)\| + |\beta_k| \|d_{k-1}\| \\ &\leq \|F(x_k)\| + t \|F(x_k)\| + t \|F(x_k)\| \\ &= (1 + 2t) \|F(x_k)\|. \end{aligned}$$

Therefore, for all $k \geq 0$, it holds that

$$\|F(x_k)\| \leq \|d_k\| \leq (1 + 2t) \|F(x_k)\|. \tag{2.9}$$

Furthermore, it is easy to see that the line search (2.5) is well defined if $F(x_k) \neq 0$.

For parameter β_k defined by (2.4), it has many choices such as $\beta_k^{S1} = \|F(x_k)\|/\|d_{k-1}\|$, or [13, 14]

$$\begin{aligned} \beta_k^{NWYL} &= \frac{\langle F(x_k), F(x_k) - \frac{\|F(x_k)\|}{\|F(x_{k-1})\|} F(x_{k-1}) \rangle}{|F(x_k)^\top d_{k-1}| + t \|F(x_k)\| \|d_{k-1}\|}, \\ \beta_k^{NPRP} &= \frac{\langle F(x_k), F(x_k) - F(x_{k-1}) \rangle}{\max\{t \|d_{k-1}\|, \|F(x_{k-1})\|^2\}}. \end{aligned}$$

From the structure of H_k , the orthogonal projection onto H_k has a closed-form expression. That is,

$$P_{H_k}(x) = \begin{cases} x - \frac{\langle F(y_k), x - y_k \rangle}{\|F(y_k)\|^2} F(y_k) & \text{if } \langle F(y_k), x - y_k \rangle > 0, \\ x & \text{if } \langle F(y_k), x - y_k \rangle \leq 0. \end{cases}$$

Lemma 2.1 For function $h_k(x)$ defined by (2.7), let $x^* \in S^*$, it holds that

$$h_k(x_k) \geq \sigma \|x_k - y_k\|^2 \quad \text{and} \quad h_k(x^*) \leq 0. \tag{2.10}$$

In particular, if $x_k \neq y_k$, then $h_k(x_k) > 0$.

Proof From $x_k - y_k = -\alpha_k d_k$ and the line search (2.5), we have

$$\begin{aligned} h_k(x_k) &= \langle F(y_k), x_k - y_k \rangle \\ &= -\alpha_k \langle F(y_k), d_k \rangle \geq \alpha_k^2 \sigma \|d_k\|^2 = \sigma \|x_k - y_k\|^2. \end{aligned}$$

On the other hand, from condition (2.1), we can obtain

$$h_k(x^*) = \langle F(y_k), x^* - y_k \rangle \leq 0.$$

This completes the proof. □

Lemma 2.1 indicates that the hyperplane $\partial H_k = \{x \in R^n | h_k(x) = 0\}$ strictly separates the current iterate from the solutions of problem (1.1) if x_k is not a solution. In addition, from

Lemma 2.1, we also can derive that the solution set S^* of problem (1.1) is included in H_k for all k .

Certainly, if Algorithm 2.1 terminates at step k , then x_k is a solution of problem (1.1). So, in the following analysis, we assume that Algorithm 2.1 always generates an infinite sequence $\{x_k\}$. Based on the lemma, we can establish the convergence of the algorithm.

Theorem 2.1 *If F is continuous and condition (2.1) holds, then the sequence $\{x_k\}$ generated by Algorithm 2.1 globally converges to a solution of problem (1.1).*

Proof First, we show that the sequences $\{x_k\}$ and $\{y_k\}$ are both bounded. In fact, it follows from $x^* \in H_k$, (2.1), (2.2) and (2.6) that

$$\begin{aligned} \|x_{k+1} - x^*\|^2 &\leq \|x_k - \xi_k F(y_k) - x^*\|^2 \\ &= \|x_k - x^*\|^2 - 2\xi_k \langle F(y_k), x_k - x^* \rangle + \xi_k^2 \|F(y_k)\|^2 \\ &\leq \|x_k - x^*\|^2 - 2\xi_k \langle F(y_k), x_k - y_k \rangle + \xi_k^2 \|F(y_k)\|^2 \\ &= \|x_k - x^*\|^2 - \frac{\langle F(y_k), x_k - y_k \rangle^2}{\|F(y_k)\|^2} \\ &\leq \|x_k - x^*\|^2 - \frac{\sigma^2 \alpha_k^4 \|d_k\|^2}{\|F(y_k)\|^4}. \end{aligned}$$

Thus the sequence $\{\|x_k - x^*\|\}$ is decreasing and convergent, and hence the sequence $\{x_k\}$ is bounded, and from (2.9), the sequence $\{d_k\}$ is also bounded. Then, by $y_k = x_k + \alpha_k d_k$, the sequence $\{y_k\}$ is also bounded. Then, by the continuity of $F(\cdot)$, there exists constant $M > 0$ such that $\|F(y_k)\| \leq M$ for all k . So,

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - x^*\|^2 - \frac{\sigma^2 \alpha_k^4 \|d_k\|^4}{M^2}, \tag{2.11}$$

from which we can deduce that

$$\lim_{k \rightarrow \infty} \alpha_k \|d_k\| = 0. \tag{2.12}$$

If $\liminf_{k \rightarrow \infty} \|d_k\| = 0$, then from (2.9) it holds that $\liminf_{k \rightarrow \infty} \|F(x_k)\| = 0$. From the boundedness of $\{x_k\}$ and the continuity of $F(\cdot)$, $\{x_k\}$ has some accumulation point \bar{x} such that $F(\bar{x}) = 0$. Then from (2.11), $\{\|x_k - \bar{x}\|\}$ converges, and thus the sequence $\{x_k\}$ globally converges to \bar{x} .

If $\liminf_{k \rightarrow \infty} \|d_k\| > 0$, from (2.9) again, we have

$$\liminf_{k \rightarrow \infty} \|F(x_k)\| > 0. \tag{2.13}$$

By (2.12), it holds that

$$\lim_{k \rightarrow \infty} \alpha_k = 0. \tag{2.14}$$

Therefore, from the line search (2.5), we have

$$-\langle F(x_k + \beta \rho^{m_k-1} d_k), d_k \rangle < \sigma \beta \rho^{m_k-1} \|d_k\|^2. \tag{2.15}$$

Since $\{x_k\}$ and $\{d_k\}$ are both bounded, then letting $k \rightarrow \infty$ in (2.15) yields that

$$-(F(\bar{x}), \bar{d}) \leq 0, \tag{2.16}$$

where \bar{x}, \bar{d} are limits of corresponding sequences. In addition, from (2.8) and (2.13), we get

$$-(F(\bar{x}), \bar{d}) = \|F(\bar{x})\|^2 > 0. \tag{2.17}$$

Obviously, (2.16) contradicts (2.17). This completes the proof. □

3 Numerical results

In this section, numerical results are provided to substantiate the efficacy of the proposed method. The codes are written in Matlab R2010a and run on a personal computer with 2.0 GHz CPU processor. For comparison, we also give the numerical results of the spectral gradient method (denoted by SGM) in [9], the conjugate gradient method (denoted by CGM) in [16]. The parameters used in Algorithm 2.1 are set as $t = 1, \sigma = 0.01, \rho = 0.5, \beta = 1$, and

$$\beta_k = \frac{\|F(x_k)\|}{\|d_{k-1}\|}.$$

For SGM, we set $\beta = 0.4, \sigma = 0.01, r = 0.001$. For CGM, we choose $\rho = 0.1, \sigma = 10^{-4}$ and $\xi = 1$. Furthermore, the stopping criterion is set as $\|F(x_k)\| \leq 10^{-6}$ for all the tested methods.

Problem 1 The mapping $F(\cdot)$ is taken as $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$, where

$$f_i(x) = e^{x_i} - 1 \quad \text{for } i = 1, 2, \dots, n.$$

Obviously, this problem has a unique solution $x^* = (0, 0, \dots, 0)^T$.

Problem 2 The mapping $F(\cdot)$ is taken as $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$, where

$$\begin{aligned} f_1(x) &= x_1 - e^{\cos(\frac{x_1+x_2}{n+1})}, \\ f_i(x) &= x_i - e^{\cos(\frac{x_{i-1}+x_i+x_{i+1}}{n+1})}, \quad \text{for } i = 2, 3, \dots, n-1, \\ f_n(x) &= x_n - e^{\cos(\frac{x_{n-1}+x_n}{n+1})}. \end{aligned}$$

Problem 3 The mapping $F(x) : R^4 \rightarrow R^4$ is given by

$$F(x) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} x_1^3 \\ x_2^3 \\ 2x_3^3 \\ 2x_4^3 \end{pmatrix} + \begin{pmatrix} -10 \\ 1 \\ -3 \\ 0 \end{pmatrix}.$$

This problem has a degenerate solution $x^* = (2, 0, 1, 0)$.

Table 1 Numerical results with different dimensions of Problem 1

Dimension	10	50	100	500	1,000
Iter.	20	21	22	23	23
CPU	0.3594	0.8750	1.7031	41.4844	257.1406

Table 2 Numerical results with different initial points of Problem 3

Initial point	Iter.	CPU	$\ F(x_k)\ $
(0, 0, 0, 0)	20	0.9219	9.9320×10^{-7}
(3, 0, 0, 0)	22	0.8438	7.0942×10^{-7}
(1, 1, 1, 0)	41	1.6250	7.2649×10^{-7}
(0, 1, 1, 1)	28	1.1875	3.8145×10^{-7}
(0, 100, 100, 1)	34	1.4063	6.6320×10^{-7}

For Problem 1, the initial point is set as $x_0 = (1, 1, \dots, 1)$, and Table 1 gives the numerical results by Algorithm 2.1 with different dimensions, where Iter. denotes the iteration number and CPU denotes the CPU time in seconds when the algorithm terminates. Table 2 lists the numerical results of Problem 2 with different initial points. The numerical results given in Table 1 and Table 2 show that the proposed method is efficient for solving the given two test problems.

4 Conclusion

In this paper, we extended the conjugate gradient method to nonlinear equations. The major advantage of the method is that it does not need to compute the Jacobian matrix or any linear equations at each iteration, thus it is suitable to solve large-scale nonlinear constrained equations. Under mild conditions, the proposed method possesses global convergence.

In Step 4 of Algorithm 2.1, we have to compute a projection onto the intersection of the feasible set C and a half-space at each iteration, which is equivalent to quadratic programming, quite time-consuming work. Hence, how to remove this projection step is one of our future research topics.

Acknowledgements

The authors thank anonymous referees for valuable comments and suggestions, which helped to improve the manuscript. This work is supported by the Natural Science Foundation of China (11671228).

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

DXF and MS organized and wrote this paper. XYW examined all the steps of the proofs in this research and gave some advice. All authors read and approved the final manuscript.

Author details

¹School of Management Sciences, Fudan University, Shanghai, China. ²Qufu Normal University, Shandong, 276826, China. ³School of Management, Qufu Normal University, Shandong, 276826, China.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 10 July 2017 Accepted: 11 September 2017 Published online: 22 September 2017

References

- Wang, YJ, Caccetta, L, Zhou, GL: Convergence analysis of a block improvement method for polynomial optimization over unit spheres. *Numer. Linear Algebra Appl.* **22**, 1059-1076 (2015)

2. Zeidler, E: *Nonlinear Functional Analysis and Its Applications*. Springer, Berlin (1990)
3. Wang, CW, Wang, YJ: A superlinearly convergent projection method for constrained systems of nonlinear equations. *J. Glob. Optim.* **40**, 283-296 (2009)
4. Wang, YJ, Caccetta, L, Zhou, GL: Convergence analysis of a block improvement method for polynomial optimization over unit spheres. *Numer. Linear Algebra Appl.* **22**, 1059-1076 (2015)
5. Wood, AJ, Wollenberg, BF: *Power Generation, Operation, and Control*. Wiley, New York (1996)
6. Chen, HB, Wang, YJ, Zhao, HG: Finite convergence of a projected proximal point algorithm for the generalized variational inequalities. *Oper. Res. Lett.* **40**, 303-305 (2012)
7. Dirkse, SP, Ferris, MC: MCPLIB: A collection of nonlinear mixed complementarity problems. *Optim. Methods Softw.* **5**, 319-345 (1995)
8. Wang, YJ, Qi, L, Luo, S, Xu, Y: An alternative steepest direction method for the optimization in evaluating geometric discord. *Pac. J. Optim.* **10**, 137-149 (2014)
9. Zhang, L, Zhou, W: Spectral gradient projection method for solving nonlinear monotone equations. *J. Comput. Appl. Math.* **196**, 478-484 (2006)
10. Yu, ZS, Lin, J, Sun, J, Xiao, YH, Liu, LY, Li, ZH: Spectral gradient projection method for monotone nonlinear equations with convex constraints. *Appl. Numer. Math.* **59**, 2416-2423 (2009)
11. Ma, FM, Wang, CW: Modified projection method for solving a system of monotone equations with convex constraints. *J. Appl. Math. Comput.* **34**, 47-56 (2010)
12. Zheng, L: A new projection algorithm for solving a system of nonlinear equations with convex constraints. *Bull. Korean Math. Soc.* **50**, 823-832 (2013)
13. Sun, M, Wang, YJ, Liu, J: Generalized Peaceman-Rachford splitting method for multiple-block separable convex programming with applications to robust PCA. *Calcolo* **54**, 77-94 (2017)
14. Li, M, Qu, AP: Some sufficient descent conjugate gradient methods and their global convergence. *Comput. Appl. Math.* **33**, 333-347 (2014)
15. Karamardian, S: Complementarity problems over cones with monotone and pseudomonotone maps. *J. Optim. Theory Appl.* **18**, 445-454 (1976)
16. Xiao, YH, Zhu, H: A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing. *J. Math. Anal. Appl.* **405**, 310-319 (2013)

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
