A New Diagonal Quasi-Newton Algorithm for Unconstrained Optimization Problems

Mahsa Nosrati
Razi University
Keyvan Amini (kamini@razi.ac.ir)
Razi University

Research Article

Keywords: Unconstrained optimization, Diagonal quasi-Newton method, Weak secant equation, Global convergence

Posted Date: July 28th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1871447/v1

A New Diagonal Quasi-Newton Algorithm for Unconstrained Optimization Problems

Mahsa Nosrati · **Keyvan Amini***

Received: date / Accepted: date

Abstract This paper presents a new diagonal quasi-Newton method for solving unconstrained optimization problems based on the weak secant equation. The new method uses new criteria to generate the Hessian approximation to control the diagonal elements. We establish the global convergence of the proposed method with the Armijo line search. Numerical results on a collection of standard test problems show that the proposed method is superior to several certain diagonal methods.

Keywords Unconstrained optimization · Diagonal quasi-Newton method · Weak secant equation · Global convergence
Mathematics subject classification 90C30 . 65K05

1 Introduction

In this study, the following unconstrained optimization problem is considered

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth function. A basic iterative process to solve (1) is

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

in which $d_k \in \mathbb{R}^n$ is a search direction, t_k is a positive step length, and x_0 is a given initial point. The success of the process (2) depends on effective choices of both the direction d_k and the step length t_k. In this regard, the step length t_k is selected so that it can cause a sufficient reduction in the function value of f and the direction d_k fulfills the descent condition, i.e., $\nabla f(x_k)^T d_k < 0$.
It is known that choosing an appropriate line search is necessary to have an effective algorithm. There are various line search algorithms that can be referred to line searches, such as Armijo [1], Wolfe [2] or Goldstein [3]. Among them, the Armijo line search is a popular method that needs the step length to fulfill the following condition:

$$f(x_k + t_k d_k) \le f(x_k) + \sigma t_k g_k^T d_k, \tag{3}$$

in which $\sigma \in (0, 1)$.

M. Nosrati · K. Amini
Department of Mathematics, Faculty of Science, Razi University, Kermanshah, Iran
E-mail: m.nosrati@razi.ac.ir
E-mail: kamini@razi.ac.ir
* Corresponding author

Choosing a descent direction d_k is another critical factor in the line search algorithms. Various processes for generating d_k have been suggested, such as gradient, Newton and quasi-Newton, conjugate gradient and trust region methods, see [4]. Among these processes, the quasi-Newton methods are effective iterative ones to solve the problem (1) due to the strong convergence properties (see [5,6]). The customary iteration of this family is as follows:

$$x_{k+1} = x_k - t_k B_k^{-1} g_k \tag{4}$$

where $g_k = \nabla f(x_k)$ and matrix B_k is a symmetric approximation of the Hessian $\nabla^2 f(x_k)$. For the next iteration, the updated Hessian approximation B_{k+1} should fulfill the following secant equation:

$$B_{k+1} s_k = y_k. \tag{5}$$

$$s_k = x_{k+1} - x_k, \quad y_k = g_{k+1} - g_k.$$

Although, it is well known that these methods have good numerical and theoretical results. But, there are some major defects in these methods. They require full matrix storage to approximate the Hessian matrix or its inverse. Thus, they are not appropriate for solving large-scale optimization problems. Some modified versions of quasi-Newton algorithms, such as truncated Newton methods [7] and Limited-memory quasi-Newton methods [8] are designed to solve a large-scale problem while implementing of these strategies is very complicated.

To create some easy minimization algorithms for unconstrained optimization, Gill and Murray [9] proposed that the Hessian matrix of the minimizing function can be approximated by a diagonal matrix with positive diagonal elements. They only retained the diagonal elements of the BFGS updating matrix. It is obvious that a diagonal matrix does not satisfy the secant equation (5). Based on this fact, Dennis and Wolkowicz [10] suggested a weak secant equation as follows:

$$s_k^T B_{k+1} s_k = s_k^T y_k. \tag{6}$$

There are many various ideas for computing B_{k+1} satisfy (6). One of the techniques is based on some variational property similar to the technique used in PSB updates [11–13]. In this approach, the diagonal matrix B_{k+1} satisfying the weak secant equation (6) is determined by solving the following subproblem:

$$\min_{B \in \mathfrak{D}} \|B - B_k\| \tag{7}$$

$$\text{subject to} \quad s_k^T B s_k = s_k^T y_k,$$

where $\|.\|$ is an arbitrary norm defined on $\mathbb{R}^{n\times n}$ and

$$\mathfrak{D} = \{B \in \mathbb{R}^{n\times n} : B \text{ is diagonal}\}. \tag{8}$$

It is clear that selecting different matrix norms in (7) can generate some different diagonal quasi-Newton methods. Using the Frobenius norm, Zhu et al. [14] got the following update formula:

$$B_{k+1} = B_k + \frac{\left(s_k^T y_k - s_k^T B_k s_k\right)}{s_k^T E_k^2 s_k} E_k^2, \tag{9}$$

where

$$E_k = \operatorname{diag}\left(s_k^{(1)}, s_k^{(2)}, \dots, s_k^{(n)}\right). \tag{10}$$

They claimed that the diagonal quasi-Newton (DQN) algorithm based on (9) has some appropriate numerical performances. They also show this method has linear convergence. In 2021, to improve the above update formula, Leong et al. [15] by using the following weighted Frobenius norm

$$\|A\|_W = \|W^{1/2} A W^{1/2}\|_F, \tag{11}$$

proposed the following update formula:

$$B_{k+1} = B_k + \frac{\left(s_k^T y_k - s_k^T B_k s_k\right)}{s_k^T \left(W E_k^2 W\right) s_k} W E_k^2 W, \tag{12}$$

where E_k is defined by (10). Clearly, if W is chosen to be an $n \times n$ identity matrix I_n, the formula (12) reduces to (9). In addition, they found that $W = B_k$ is one of the best choices. In 2019, Andrei [16] presented another idea to generate B_{k+1} in which the diagonal matrix $\varDelta_k$ can be computed by solving the following constrained optimization subproblem:

$$\min_{\varDelta_k \in \mathfrak{D}} \frac{1}{2} \|\varDelta_k\|_F^2 + \operatorname{tr}\left(B_k + \varDelta_k\right) \tag{13}$$

$$\text{subject to} \quad s_k^T \left(B_k + \varDelta_k\right) s_k = s_k^T y_k,$$

in which the operator $\operatorname{tr}(A)$ defines the trace of matrix A and $\|.\|_F$ denotes the Frobenius norm. To control the eigenvalues of the updated matrix B_{k+1}, the second term in the objective function of (13), $\operatorname{tr}\left(B_k + \varDelta_k\right)$, is added. By solving the subproblem (13), Andrei introduced the following direction:

$$d_{k+1} = -B_{k+1}^{-1} g_{k+1}.$$

where B_{k+1} is given by

$$B_{k+1} = B_k + \frac{\left(s_k^T y_k + s_k^T s_k - s_k^T B_k s_k\right)}{s_k^T E_k^2 s_k} E_k^2 - I_n, \tag{14}$$

and E_k is defined by (10). It is clear that the positive definiteness of this update is not guaranteed. Thus, the search direction d_{k+1} may not be a descent direction. To cope with this shortcoming, Andrei modified the direction d_{k+1} by enforcing some simple technique. He also claimed this algorithm is well defined and converges q-linearly to a solution under Wolfe line search. Numerical experiments showed the numerical efficiency of this algorithm compared to some other diagonal methods. It is noticeable that the DQN direction can be considered as a spectral gradient direction; see, e.g., [17–20].

Applying the same pattern of the DQN method, another approach introduced by Andrei [21] in which the diagonal elements are determined by minimizing the measure of Byrd and Nocedal function. In 2021, Andrei [22] presented an accelerated DQN algorithm. He used scaling of the forward finite differences directional derivative of the gradient components as the elements of the diagonal matrix approximating the Hessian.

In this paper, we present a new DQN method by modifying the subproblem (13). Under Armijo line search, the global convergence of the new algorithm is established. Numerical results show that the proposed method is more promising.

The rest of the paper is as follows. In the next section, we explain the main idea of the new method. In section 3, the convergence analysis of the proposed algorithm is investigated. Numerical results are presented in section 4. Some conclusions are delivered in the last section.

2 New Algorithm

In this section, we present a new DQN method by modifying the subproblem (13). In this sense, in view of the fact that $\operatorname{tr}\left(B_k + \varDelta_k\right) = \operatorname{tr}\left(B_k\right) + \operatorname{tr}\left(\varDelta_k\right)$, it follows that $\operatorname{tr}\left(B_k\right)$ in the subproblem (13) does not play any role throughout the minimization procedure; while the updated matrix B_{k+1} is a combination of B_k and $\varDelta_k$. To control the eigenvalues of matrix B_{k+1}, we modify the subproblem (13) as follows:

$$\min_{\varDelta_k \in \mathfrak{D}} \frac{1}{2} \|\varDelta_k\|_F^2 + \operatorname{tr}\left(\varDelta_k B_k\right) \tag{15}$$

$$\text{subject to} \quad s_k^T \left(B_k + \varDelta_k\right) s_k = s_k^T y_k,$$

The constrained problem (15) can be solved by finding a stationary point of the following Lagrangian function

$$L(\varDelta_k, \lambda) = \frac{1}{2}\mathrm{tr}\left(\varDelta_k^T \varDelta_k\right) + \mathrm{tr}\left(\varDelta_k B_k\right) + \lambda\left(s_k^T B_k s_k + s_k^T \varDelta_k s_k - s_k^T y_k\right), \tag{16}$$

where λ is the Lagrange multiplier. In this sense, we obtain a solution of the following nonlinear system:

$$\frac{\partial L}{\partial \varDelta_k} = \varDelta_k + B_k + \lambda\, E_k^2 = 0, \tag{17}$$

$$\frac{\partial L}{\partial \lambda} = s_k^T B_k s_k + s_k^T \varDelta_k s_k - s_k^T y_k = 0, \tag{18}$$

where E_k is defined by (10), and subsequently $\varDelta_k$ is obtained as

$$\varDelta_k = -B_k - \lambda\, E_k^2. \tag{19}$$

Multiplying the relation (19) by s_k^T and invoking equation (18), we can result in

$$-s_k^T B_k s_k - \lambda\, s_k^T E_k^2 s_k = s_k^T y_k - s_k^T B_k s_k,$$

and so,

$$\lambda = -\frac{s_k^T y_k}{s_k^T E_k^2 s_k}. \tag{20}$$

Substituting (20) into (19) concludes that

$$\varDelta_k = \frac{s_k^T y_k}{s_k^T E_k^2 s_k} E_k^2 - B_k, \tag{21}$$

and hence,

$$B_{k+1} = \theta_k E_k^2, \tag{22}$$

where $\theta_k = \frac{s_k^T y_k}{s_k^T E_k^2 s_k} = \frac{s_k^T y_k}{\sum_{j=1}^n \left(s_k^{(j)}\right)^4}$. Therefore, the search direction can be obtained as follows:

$$d_{k+1} = -B_{k+1}^{-1} g_{k+1}, \tag{23}$$

i.e. on components:

$$d_{k+1}^{(i)} = -\frac{g_{k+1}^{(i)}}{b_{k+1}^{(i)}}, \qquad i = 1, \ldots, n, \tag{24}$$

where

$$b_{k+1}^{(i)} = \theta_k \left(s_k^{(i)}\right)^2, \qquad i = 1, \ldots, n. \tag{25}$$

To be sure of the descent property of the search direction d_{k+1}, it is sufficient to have $b_{k+1}^{(i)} > 0$ for all $i = 1, \ldots, n$. In this sense, after computing $b_{k+1}^{(i)}$ from (25), we modify it as follows:

$$\hat{b}_{k+1}^{(i)} = \begin{cases} b_{k+1}^{(i)}, & \text{if} \quad \varepsilon_b \le b_{k+1}^{(i)} \le \gamma, \\ 1, & \text{otherwise}, \end{cases} \tag{26}$$

where $\varepsilon_b \in (0, 1)$ and γ is a suitable upper bound for the diagonal elements. Theoretically, the roles of parameters γ and ε_b are to truncate the elements $b_{k+1}^{(i)}$ from two sides when they are too large or too small. As a result, the sequence $\{\widehat{B}_{k+1}\}$ always satisfies

$$\varepsilon_b \le \hat{b}_{k+1}^{(i)} \le \gamma, \qquad i = 1, \ldots, n, \tag{27}$$

and is bounded. Based on this discussion, we define the search direction as follows:

$$d_{k+1} = -\widehat{B}_{k+1}^{-1} g_{k+1}, \tag{28}$$

i.e., on components:

$$d_{k+1}^{(i)} = -\frac{g_{k+1}^{(i)}}{\hat{b}_{k+1}^{(i)}}, \qquad i = 1, \ldots, n. \tag{29}$$

From the above discussion, the following lemma implies.

Lemma 1. *Let $\hat{b}_{k+1}^{\min}$ and $\hat{b}_{k+1}^{\max}$ be the smallest and largest diagonal component of $\widehat{B}_{k+1}$, respectively. Then direction d_{k+1} yielded by* (28) *satisfies*

$$|g_{k+1}^T d_{k+1}| \geq \frac{\|g_{k+1}\|^2}{\hat{b}_{k+1}^{\max}}, \tag{30}$$

$$\frac{|g_{k+1}^T d_{k+1}|}{\|d_{k+1}\|^2} \geq \frac{\left(\hat{b}_{k+1}^{\min}\right)^2}{\hat{b}_{k+1}^{\max}}. \tag{31}$$

Proof Clearly, we know that the following inequality is held.

$$\frac{1}{\hat{b}_{k+1}^{\max}} I_{n\times n} \leq \widehat{B}_{k+1}^{-1} \leq \frac{1}{\hat{b}_{k+1}^{\min}} I_{n\times n},$$

Multiplying this inequality by g^T from the left and by g from the right result

$$\frac{\|g_{k+1}\|^2}{\hat{b}_{k+1}^{\max}} \leq g_{k+1}^T \widehat{B}_{k+1} g_{k+1} \leq \frac{\|g_{k+1}\|^2}{\hat{b}_{k+1}^{\min}}. \tag{32}$$

Combining (32) with (28) follows the inequalities (30) and (31) immediately. □

Based on the above discussion, the new algorithm can be introduced as follows.

Algorithm 2.1 (ADQN Algorithm)

Input: An initial point $x_0 \in \mathbb{R}^n$, given positive constants $\varepsilon_b, \varepsilon_g$, and $\sigma, \beta \in (0, 1)$.
Step 0. Set k = 0 and $d_0 = -g_0$.
Step 1. If the stop criterion is established, stop. Otherwise, go to step 2.
Step 2. Set $x_{k+1} = x_k + t_k d_k$ with $t_k = \beta^m$, where m is the smallest non-negative integer that t_k satisfies (3).
Step 3. Compute the search direction d_{k+1} by (29).
Step 4. Set $k = k + 1$ and go to step 1.

---

3 Convergence analysis

In this part, the global convergence of Algorithm 2.1 is proved. In this regard, we require the following assumptions.

Assumption 3.1

(A1) The level set $S = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}$ is bounded.
(A2) In a neighborhood U of S, the function f is continuously differentiable and its gradient is Lipschitz continuous; namely, there exists a constant $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\|, \qquad \forall\, x,\, y \in U.$$

The next lemma displays that at each iteration, the reduction of the function f is at least as much a constant coefficient of $\|g_k\|^2$.

Lemma 2. *Suppose that $\{x_k\}$ is generated by Algorithm 2.1 and Assumption 3.1 is fulfilled. Then,*

$$f(x_k) - f(x_{k+1}) \geq \mu\|g_k\|^2, \tag{33}$$

where

$$\mu = \min\left\{\frac{\sigma}{\gamma},\ \frac{\sigma\,\varepsilon_b^2\,(1-\sigma)}{L\,\gamma^2}\beta\right\}.$$

Proof According to step 2 of Algorithm 2.1, we know $t_k \leq 1$ and

$$f(x_k) - f(x_{k+1}) \geq \sigma t_k |g_k^T d_k|, \qquad \forall\, k \in \mathbb{N}. \tag{34}$$

Note that if $t_k = 1$, then from the relations (27) and (30), we result in

$$f_k - f_{k+1} \geq \mu_1\|g_k\|^2, \tag{35}$$

where $\mu_1 = \frac{\sigma}{\gamma}$. Now, we consider $t_k < 1$. In this case, similar to the proof of Proposition 1 in [23], we can result that

$$t_k > \frac{\beta(1-\sigma)}{L}\frac{|g_k^T d_k|}{\|d_k\|^2}. \tag{36}$$

It follows from the relations (31) and (36) that

$$t_k > \frac{\beta(1-\sigma)}{L}\frac{\left(\hat{b}_k^{\min}\right)^2}{\hat{b}_k^{\max}}. \tag{37}$$

Embedding the relations (27), (30), and (37) into (34), we get

$$f(x_k) - f(x_{k+1}) \geq \mu_2\|g_k\|^2, \tag{38}$$

in which $\mu_2 = \frac{\sigma\,\varepsilon_b^2\,(1-\sigma)}{L\,\gamma^2}\beta$.
Finally the inequalities (35) and (38) imply that

$$f(x_k) - f(x_{k+1}) \geq \mu\|g_k\|^2,$$

where $\mu = \min\{\mu_1, \mu_2\}$ and the proof is completed. □

In the next theorem, the global convergence of Algorithm 2.1 with Armijo line search is proved.

Theorem 1 *Assume that $\{x_k\}$ is generated by Algorithm 2.1. Under Assumption 3.1, we have*

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

Proof It follows from the relation (33) that $f(x_k) = f(x_{k+1})$ if and only if $g_k = 0$, which is the same as saying that x_k is a stationary point of the problem (1). This implies that $f(x_{k+1}) < f(x_k)$ for any $k \geq 0$ and since f is bounded below over a compact set, we have

$$\lim_{k\to\infty} f(x_k) - f(x_{k+1}) = 0.$$

This fact along with the relation (33) implies that that

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

and the proof is complete. □

4 Numerical experiments

This section reports some numerical behavior of Algorithm 2.1 to evaluate its numerical performance. We give some numerical results on a set of test problems consisting of 101 instances given by Andrei [24], in which most of the problems are extracted from the CUTE collection by Bongartz et al. [25]. We compare the ADQN method against the following DQN algorithms:

- SD: The steepest descent method. Consider Algorithm 2.1 with $d_{k+1} = -g_{k+1}$.
- DQN-B: A modified version of the DQN method proposed by Leong et al. [15]. Consider Algorithm 2.1 with $d_{k+1} = -B_{k+1}^{-1} g_{k+1}$ where B_{k+1} is given by (12).
- DNRTR: The DQN method proposed by Andrei [21]. Consider Algorithm 2.1 with $d_{k+1} = -B_{k+1}^{-1} g_{k+1}$ where B_{k+1} is given by (14).

All the codes were written in MATLAB R2018b and run on a Laptop with 1.8 GHz CPU, 4 GB RAM and Windows 10 operation system.

For four tested algorithms, the step length t_k was chosen by the backtracking procedure with the parameters $\sigma = 0.1$ and $\beta = 0.5$. The following same parameters are used for all algorithms:

$$\varepsilon_g = 10^{-4}, \quad \varepsilon_b = 0.01, \quad \gamma = 10^5;$$

We terminate algorithms when $\|g_k\| \leq \varepsilon_g(1 + |f_k|)$ or Itr > 10000, where Itr denotes the number of iterations.

To investigate the numerical performance of the above mentioned algorithms, we adopt the performance profiles of Dolan and Moré [26] based on the number of function evaluations (N_f) and gradient evaluations (N_g). These performance profiles enable us to study the various methods in terms of efficiency and robustness. It is called a robust solver if it finds an optimal solution and is called efficient if it needs fewer N_f and N_g. Efficiency and robustness rates are readable on the left and right vertical axes of the performance profiles, respectively.

Figures 1 and 2 indicate the performance of the above mentioned algorithms based on N_f and N_g for $n = 1008$, respectively.

Figure 1 displays that the ADQN solves nearly 46% of the test problems at the least N_f, while DNRTR, DQN-B and SD solve nearly 31% , 33%, and 26% of the test problems, respectively.

Figure 2 displays that the ADQN solves nearly 43% of the test problems at the least N_g, while DNRTR, DQN-B and SD solve nearly 36%, 37%, and 22% of the test problems, respectively.

According to Figures 1 and 2, we can result that the ADQN is robust and efficient for solving unconstrained optimization problems.

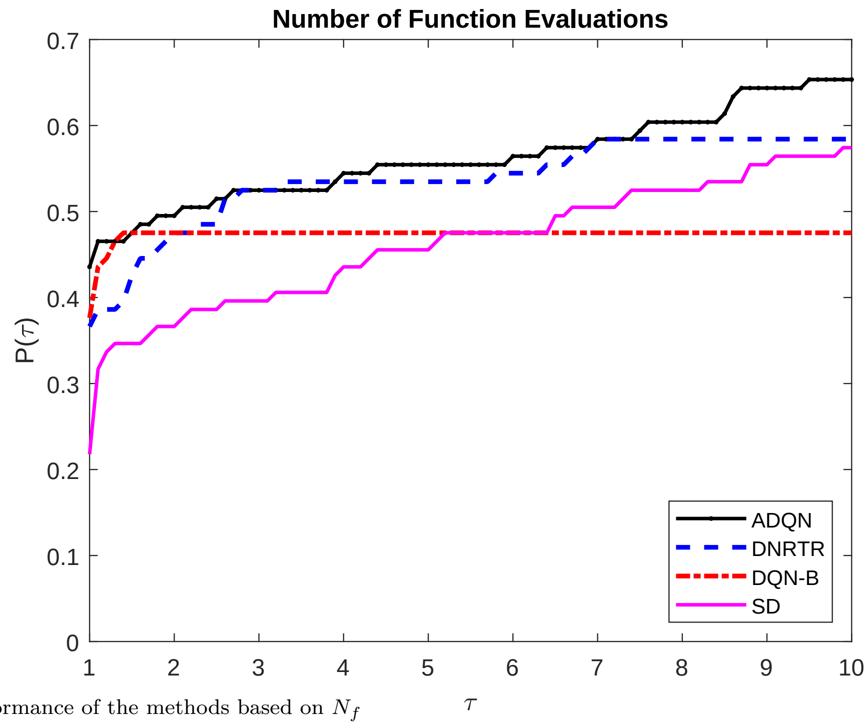

Fig. 1 Performance of the methods based on N_f

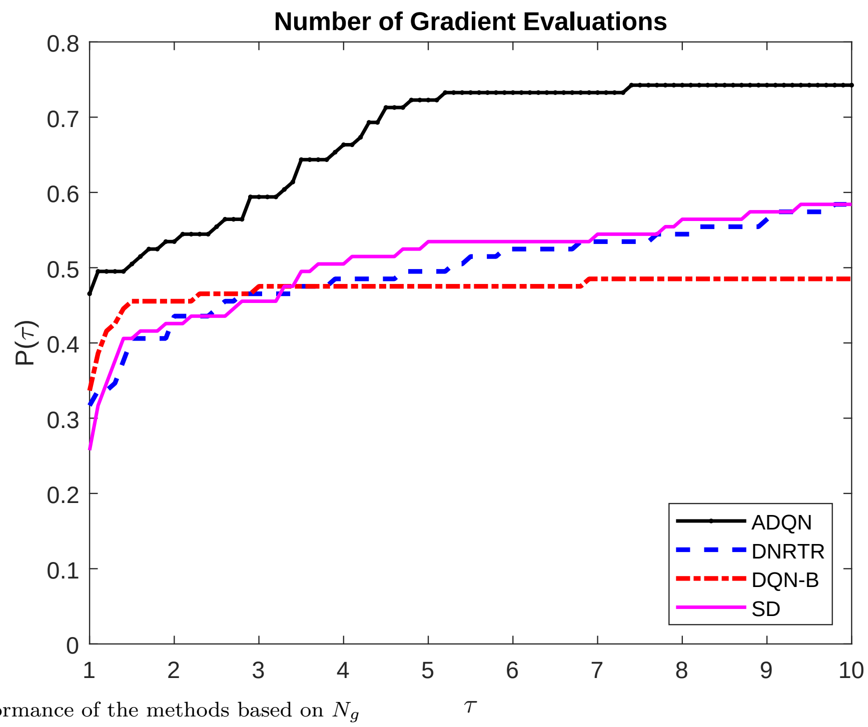

Fig. 2 Performance of the methods based on N_g

5 Conclusions

We presented a new DQN method for solving the unconstrained optimization problems and established the global convergence of the proposed method with the Armijo line search. Based on numerical results, the new method was more efficient and more robust than several available DQN methods.

Declarations

Ethical Approval and Consent to participate

Not Applicable

Consent for publication

Not Applicable

Human and Animal Ethics

Not Applicable

Availability of supporting data

The data and code that support the findings of this study are available from the corresponding author upon request.

Competing interests

The authors declare competing interests.

Funding

Not Applicable

Authors contributions

Not Applicable

Acknowledgments

Not Applicable

References

1. Armijo, L.: Minimization of functions having Lipschitz continuous first partial derivatives. Pacific J. Math. 16, 1–3 (1966). https://doi.org/pjm/1102995080
2. Wolfe, P.: Convergence Conditions for Ascent Methods. SIAM Rev. 11, 226–235 (1969)
3. Goldstein, A.A.: On Steepest Descent. J. Soc. Ind. Appl. Math. Ser. A Control. 3, 147–151 (1965). https://doi.org/10.1137/0303013
4. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer, New York (2006). https://doi.org/10.1007/978-0-387-40065-5
5. Dennis, J.E., Moré, J.J.: A Characterization of Superlinear Convergence and Its Application to Quasi-Newton Methods. Math. Comput. 28, 549–560 (1974). https://doi.org/10.2307/2005926
6. Dennis John E, J., Moré, J.J.: Quasi-Newton methods, motivation and theory. SIAM Rev. 19, 46–89 (1977). https://doi.org/10.1137/1019005
7. Nash, S.G.: Preconditioning of truncated-Newton methods. SIAM J. Sci. Stat. Comput. 6, 599–616 (1985). https://doi.org/10.1137/0906042
8. Nocedal, J.: Updating quasi-Newton matrices with limited storage. Math. Comput. 35, 773–782 (1980). https://doi.org/10.1090/S0025-5718-1980-0572855-7
9. Gill, P.E., Murray, W.: Conjugate-Gradient Methods for Large-Scale Nonlinear Optimization. STANFORD UNIV CALIF SYSTEMS OPTIMIZATION LAB (1979)
10. Dennis Jr., J.E., Wolkowicz, H.: Sizing and Least-Change Secant Methods. SIAM J. Numer. Anal. 30, 1291–1314 (1993). https://doi.org/10.1137/0730067
11. Broyden, C.G., Dennis Jr, J.E., Moré, J.J.: On the local and superlinear convergence of quasi-Newton methods. IMA J. Appl. Math. 12, 223–245 (1973). https://doi.org/10.1093/imamat/12.3.223
12. Powell, M.J.D.: A new algorithm for unconstrained optimization. In: Nonlinear programming. pp. 31–65. Elsevier (1970)
13. Powell, M.J.D.: Some global convergence properties of a variable metric algorithm for minimization without exact line searches. In: R.W. Cottle, C.E. Lemke (Eds.), Nonlinear Programming, SIAM-AMS Proceedings, Vol. IX, SIAM, Philadelphia, PA. 53–72 (1976)
14. Zhu, M., Nazareth, J.L., Wolkowicz, H.: The quasi-Cauchy relation and diagonal updating. SIAM J. Optim. 9, 1192–1204 (1999). https://doi.org/10.1137/S1052623498331793
15. Leong, W.J., Enshaei, S., Kek, S.L.: Diagonal quasi-Newton methods via least change updating principle with weighted Frobenius norm. Numer. Algorithms. 86, 1225–1241 (2021)
16. Andrei, N.: A diagonal quasi-Newton updating method for unconstrained optimization. Numer. Algorithms. 81, 575–590 (2019)
17. Barzilai, J., Borwein, J.M.: Two-point step size gradient methods. IMA J. Numer. Anal. 8, 141–148 (1988). https://doi.org/10.1093/imanum/8.1.141
18. Farid, M., Leong, W.J., Zheng, L.H.: A new diagonal gradient-type method for large scale unconstrained optimization. Univ. Politeh. BUCHAREST Sci. Bull. A-APPLIED Math. Phys. 75, 57–64 (2013)
19. Leong, W.J., Farid, M., Hassan, M.A.: Improved Hessian approximation with modified quasi-Cauchy relation for a gradient-type method. Adv. Model. Optim. 12, 37–44 (2010)
20. Raydan, M.: The Barzilai and Borwein Gradient Method for the Large Scale Unconstrained Minimization Problem. SIAM J. Optim. 7, 26–33 (1997)
21. Andrei, N.: A diagonal quasi-Newton updating method based on minimizing the measure function of Byrd and Nocedal for unconstrained optimization. OPTIMIZATION. 67, 1553–1568 (2018). https://doi.org/10.1080/02331934.2018.1482298
22. Andrei, N.: A new accelerated diagonal quasi-Newton updating method with scaled forward finite differences directional derivative for unconstrained optimization. OPTIMIZATION. 70, 345–360 (2021). https://doi.org/10.1080/02331934.2020.1712391
23. Andrei, N.: An acceleration of gradient descent algorithm with backtracking for unconstrained optimization. Numer. ALGORITHMS. 42, 63–73 (2006). https://doi.org/10.1007/s11075-006-9023-9
24. Andrei, N.: An unconstrained optimization test functions collection. Adv. Model. Optim. 10, 147–161 (2008)
25. Bongartz, I., Conn, A.R., Gould, N., Toint, P.L.: CUTE: Constrained and unconstrained testing environment. ACM Trans. Math. Softw. 21, 123–160 (1995)
26. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program. 91, 201–213 (2002)