Journal of Mathematical Modeling
Vol. -, No. -, 2026, pp. -. Research Article

JMM

Combining the steepest descent and a diagonal quasi-Newton method for unconstrained optimization

Narges Bidabadi[†][*]

[†] *Department of Mathematical Sciences, Yazd University, Yazd, Iran*

Email(s): n_bidabadi@yazd.ac.ir

Abstract. Optimization problems pose significant computational challenges in various scientific and engineering fields. This paper proposes a novel hybrid algorithm that combines the global convergence properties of the Steepest Descent (SD) method with the superior convergence rate of the Diagonal Quasi-Newton (DQN) method. The key idea is to initiate the optimization process using the SD method to ensure stable progress towards the minimum and then switch to the DQN method based on a defined switching criterion to accelerate the final convergence. Numerical experiments on a set of standard benchmark problems demonstrate that the proposed hybrid method significantly reduces the computational time compared to using the SD method, the BFGS method, and some DQN methods.

Keywords: Steepest descent, diagonal quasi-Newton, hybrid algorithm.
AMS Subject Classification 2010: 90C30, 65K05.

1 Introduction

Let $f:\mathbb{R}^n \to \mathbb{R}$ be a twice continuously differentiable function. This paper examines the unconstrained nonlinear optimization problem

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

One step of the iterative methods for solving problem (1) is defined as follows:

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

where $\alpha_k > 0$ is a step length and d_k is the search direction. Quasi-Newton methods obtain search direction d_k by solving

$$B_k d_k = -g_k,$$

---

[*] Corresponding author

Received: 19 February 2026/ Revised: 05 June 2026/ Accepted: 06 June 2026
DOI: 10.22124/jmm.2026.33059.3012

where $g_k = \nabla f(x_k)$ and B_k approximates the Hessian matrix of the function f at the point x_k. The foundation of these methods is the secant equation,

$$B_k s_{k-1} = y_{k-1}, \tag{3}$$

where, $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$.

Among quasi-Newton methods, the BFGS approach is widely regarded as one of the most efficient, in which the Hessian matrix approximations are obtained using the following iterative relation:

$$B_{k+1} = B_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{B_k s_k s_k^\top B_k}{s_k^\top B_k s_k}.$$

In many real-world optimization problems, the cost of storing and updating a full Hessian approximation makes standard quasi-Newton methods infeasible. The memory and computational resources required grow rapidly with problem dimension, limiting their applicability. Diagonal quasi-Newton methods offer a compelling solution to this issue. By constraining the Hessian approximation to be a positive definite diagonal matrix, they drastically cut down on memory usage and per-iteration complexity. This design places them at a useful intersection: they retain some of the curvature information that makes quasi-Newton methods effective, while maintaining the low computational profile typically associated with gradient-based methods. Significant recent research has therefore focused on developing robust diagonal updating strategies.

Enshaei et al. [7] introduced an innovative diagonal quasi-Newton method derived from variational principles under the generalized Frobenius norm. This methodology yields computationally efficient diagonal Hessian approximations that significantly reduce storage requirements and arithmetic complexity. Mokhtari and Ingber [8] proposed a diagonal-augmented BFGS method for convex optimization, in which the Hessian estimates are based not only on the gradients but also on the diagonal part of the true Hessian matrix. Pioneering work by Andrei [3] laid a firm groundwork, introducing innovative update techniques based on minimizing the Byrd and Nocedal measure function. Leong et al. [11] presented a class of low-memory quasi-Newton methods with a standard backtracking line search for large-scale unconstrained minimization. Their methods were derived by means of a least-change updating technique analogous to that used for the DFP method, except that the full quasi-Newton matrix was replaced by a diagonal matrix. Al-Siyabi and Al-Baali [1] considered several diagonal quasi-Newton methods for solving large-scale unconstrained optimization problems. A simple and effective approach for diagonal quasi-Newton algorithms was presented by proposing new updates for the diagonal entries of the Hessian. Moreover, they suggested employing an extra BFGS update of the diagonal updating matrix and using its diagonal again. Subsequently, they proposed some diagonal Hessian approximations of the least-squares nonlinear function [2]. Bukar et al. [5] presented a diagonal quasi-Newton updating strategy. The elements of the diagonal matrix approximating the Hessian were determined using the log-determinant norm that satisfies a weaker secant equation. To ensure the positive definiteness of the proposed diagonal updating matrices, their Cholesky factors are considered within the variational problem. Nosrati and Amini [10] presented a new diagonal quasi-Newton method for solving unconstrained optimization problems based on the weak secant equation.

This paper extends this foundation by introducing a new hybrid algorithm. The rest of this paper proceeds as follows. Section 2 details the development of our combined steepest descent and diagonal quasi-Newton approach. We then analyze its theoretical foundations in Section 3. The efficacy of the algorithm is evaluated through numerical experiments in Section 4. The paper concludes in Section 5 with a summary of our results and a discussion of their significance.

2 A combined steepest descent and diagonal quasi-Newton approach

The step length in this study is determined by the Wolfe conditions [9]:

$$\begin{aligned} &f(x_k+\alpha_k d_k) \le f(x_k) + c_1 \alpha_k g_k^T d_k, \\ &g(x_k+\alpha_k d_k)^T d_k \ge c_2 g(x_k)^T d_k, \end{aligned} \tag{4}$$

with $0 < c_1 < c_2 < 1$. Ensuring that the line search satisfies the Wolfe conditions guarantees that $s_k^T y_k > 0$ for all k [9].

The search direction d_k is computed by considering two distinct cases.

Case 1: $s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2 > 0$

In this case, the search direction is computed using a diagonal quasi-Newton method. The matrix B_k is confined to a diagonal representation of the form:

$$B_k = Diag(b_k^1, ..., b_k^n).$$

The construction of B_k is then guided by two key requirements: maintaining positive definiteness and satisfying a weak secant condition, which can be stated as:

$$s_{k-1}^T B_k s_{k-1} = s_{k-1}^T y_{k-1}. \tag{5}$$

To reduce the condition number of the Hessian matrix B_k, it is necessary that the eigenvalues are neither very small nor very large. Hence, if the eigenvalues are bounded both from below and above, the condition number of the matrix remains bounded. For this purpose, we consider the following optimization problem to compute the eigenvalues of the diagonal matrix B_k (namely, the diagonal entries):

$$\begin{aligned} \min \quad & trace(B_k) = b_k^1 + ... + b_k^n \\ s.t. \quad & s_{k-1}^T B_k s_{k-1} = s_{k-1}^T y_{k-1} \\ & b_k^i \ge 1, \ i = 1, \ldots, n. \end{aligned} \tag{6}$$

Problem (6) is equivalent to

$$\begin{aligned} \min \quad & \tilde{b_k}^1 + ... + \tilde{b_k}^n \\ s.t. \quad & \textstyle\sum_{i=1}^n (s_{k-1}^i)^2 \tilde{b_k}^i = s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2, \\ & \tilde{b_k}^i \ge 0, \ i = 1, \ldots, n, \end{aligned} \tag{7}$$

where $\tilde{b_k}^i = b_k^i - 1$. This optimization shares similarities with the knapsack problem and admits an optimal solution as follows:

$$\tilde{b_k}^i = \begin{cases} \frac{s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2}{(s_{k-1}^j)^2}, & j = argmax\{(s_{k-1}^i)^2\}, \\ 0, & o.w. \end{cases} \tag{8}$$

Therefore, the optimal solution to the problem (6) is

$$b_k^i = \begin{cases} \frac{s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2}{(s_{k-1}^j)^2} + 1, & j = argmax\{(s_{k-1}^i)^2\}, \\ 1, & o.w. \end{cases} \tag{9}$$

Case 2: $s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2 \leq 0$
Note that in this case, problem (7) is infeasible, and consequently problem (6), which is equivalent to it, becomes infeasible as well. Therefore, one cannot obtain a matrix with diagonals no smaller than 1 that satisfies the weak secant condition. Reducing the diagonal entries of matrix B_k will lead to ill-conditioning of this matrix. In this case, the search direction is computed using steepest descent method, that is $d_k = -g_k$. Note that this represents a specific instance of the quasi-Newton method, obtained by setting $B_k = I$.

The combined steepest descent and diagonal quasi-Newton algorithm is outlined in the following.

Algorithm 1: A Combined Steepest Descent and Diagonal Quasi-Newton Algorithm (SD-DQN)

```
1: Select a starting point $x_0$, a stopping parameter $\varepsilon > 0$, and constants $c_1$ and $c_2$ satisfying $0 < c_1 < c_2 < 1$.
2: Set $k = 0$. Compute $g_0 = \nabla f(x_0)$ and set $d_0 = -g_0$. Go to Step 5.
3: If $s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2 \geq 0$, calculate $b_k^i$'s using relation (9), calculate the search direction $d_k$ using $d_k^i = -\frac{g_k^i}{b_k^i}$ and go to step 5. Otherwise, proceed to the next step.
4: Calculate the search direction $d_k$ using $d_k = -g_k$.
5: Calculate $\alpha_k$ satisfying equation (4). Set $x_{k+1} = x_k + \alpha_k d_k$, and $g_{k+1} = \nabla f(x_{k+1})$.
6: If $\|g_{k+1}\| < \varepsilon \|g_k\|$, stop and accept $x_{k+1}$ as an approximate solution to the problem; otherwise calculate the values of $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$. Set $x_k = x_{k+1}$, $g_k = g_{k+1}$ and $k = k+1$, go to step 3.
```

In the upcoming section, we turn our attention to proving the global convergence properties of Algorithm 1.

3 Convergence analysis

The convergence properties of the proposed method are investigated here. To facilitate this analysis, the following assumptions are made.
Assumptions:
(A1) The functions f, is twice continuously differentiable.
(A2) ∇f is Lipschitz continuous , that is, there exist a constant $L \geq 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \ \forall x, y.$$

We begin by proving the following key theorem.

Theorem 1. *Let $f : \mathbb{R}^n \to \mathbb{R}$ and ∇f satisfy the Assumptions (A1) and (A2). Then, the condition number of the matrices B_k produced by Algorithm 1 have an upper bound.*

Proof. In Case 1, we have $b_k^i \geq 1$, for all i that yields to $\|B_k^{-1}\| \leq 1$. Also we have

$$\|B_k\| = \max_i b_k^i = \frac{s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2}{(s_{k-1}^j)^2} + 1, \quad j = argmax\{(s_{k-1}^i)^2\}.$$

Using Assumption (A2), we obtain

$$s_{k-1}^T y_{k-1} \leq \|s_{k-1}\|\|y_{k-1}\| = \|s_{k-1}\|\|\nabla f(x_k) - \nabla f(x_{k-1})\| \leq L\|s_{k-1}\|\|x_k - x_{k-1}\| = L\|s_{k-1}\|^2.$$

Therefore,

$$\frac{s_{k-1}^T y_{k-1} - \|s_{k-1}\|^2}{(s_{k-1}^j)^2} \leq \frac{s_{k-1}^T y_{k-1}}{(s_{k-1}^j)^2} \leq \frac{L\|s_{k-1}\|^2}{(s_{k-1}^j)^2} = \frac{L\sum_{i=1}^n (s_{k-1}^i)^2}{(s_{k-1}^j)^2} = L\sum_{i=1}^n \frac{(s_{k-1}^i)^2}{(s_{k-1}^j)^2} \leq L\sum_{i=1}^n 1 \leq nL,$$

which implies $\|B_k\| \leq nL + 1$. Thus

$$cond(B_k) = \|B_k^{-1}\|\|B_k\| \leq nL + 1. \tag{10}$$

In Case 2, we have $B_k = I$ and $cond(B_k) = \|B_k^{-1}\|\|B_k\| = 1$. □

Note: The obtained upper bound for the condition number of the matrix depends on the size of the problem, and this bound increases as n increases. However, this bound occurs in the worst case, and in practice, the condition number of matrices is much smaller than this value.

The following theorem plays an important role in proving the convergence of iterative methods in optimization.

Theorem 2. *([9]) Let $f : \mathbb{R}^n \to \mathbb{R}$ be a twice continuously differentiable function bounded from below. Consider an open set Ω containing the sublevel set $\{x : f(x) \leq f(x_0)\}$, where x_0 denotes the initial iterate. Assume ∇f is Lipschitz continuous on Ω. If d_k constitutes a descent direction and α_k is chosen to meet the Wolfe conditions, then we have the following inequality:*

$$\sum_{k=0}^{\infty} \cos^2\theta_k \|\nabla f(x_k)\|^2 < \infty \tag{11}$$

in which θ_k measures the angle separating the search direction d_k from the steepest descent direction $-\nabla f(x_k)$.

The following lemma demonstrates the convergence of the algorithm.

Lemma 1. *Given Assumptions (A1) and (A2), convergence of the SD-DQN algorithm to a stationary point is assured.*

Proof. The SD-DQN algorithm defines its search direction through the following formula:

$$d_k = -B_k^{-1}\nabla f(x_k).$$

Therefore,

$$\cos\theta_k = \frac{d_k^T B_k d_k}{\|d_k\|\|B_k d_k\|} \geq \frac{d_k^T B_k d_k}{\|d_k\|^2\|B_k\|} \geq \frac{1}{\|B_k^{-1}\|\|B_k\|} = \frac{1}{cond(B_k)}.$$

A direct consequence of Theorem 1 is that $\cos\theta_k$ is uniformly bounded away from zero whenever the condition numbers of B_k admit a uniform upper bound. An immediate consequence of Theorem 2 is that

$$\lim_{k\to\infty} \|\nabla f(x_k)\| = 0, \tag{12}$$

which establishes the global convergence of the algorithm. □

4 Numerical results

This section numerically compares the output of the proposed algorithm (SD-DQN) against the following four methods:
SD: Steepest descent method.
BFGS: BFGS quasi-Newton method.
ADQN: The diagonal quasi-Newton method proposed in [10].
MINFI: The diagonal quasi-Newton method proposed in [3].

We implemented the algorithms in MATLAB and were run on a system with a 3.3 GHz CPU and 16 GB of RAM. The step length α_k is computed using a quadratic interpolation algorithm that satisfies the Wolfe conditions (4) with $c_1 = 0.1$ and $c_2 = 0.8$, following Algorithms 3.5 and 3.6 in Nocedal and Wright [9]. Algorithm execution was halted upon the occurrence of any of the following conditions:

- Iteration count > 300.
- $\|\nabla f(x_k)\| < 10^{-4}$.
- $\|x_{k+1} - x_k\|_\infty < 10^{-6}$.
- $\|x_{k+1} - x_k\|_2 < 10^{-6}$.

We collected numerical results for 102 test problems from [4] for different n values($n = 1000, 2000, 3000$). Figures 1-3 present Dolan-Moré plots [6] that compare the algorithms based on computational time revealing the superior performance of our proposed algorithm over the other algorithms. In terms of the number of problems solved, although Algorithm ADQN performed better (successfully solving 82% of the problems), Algorithm SD-DQN still performed better than the other three algorithms, successfully solving 73% of the problems.

5 Conclusion

This paper proposes a hybrid SD-DQN algorithm for unconstrained optimization. The method switches from steepest descent to a diagonal quasi-Newton update when $s_{k-1}^T y_{k-1} > \|s_{k-1}\|^2$. The diagonal Hessian approximation minimizes trace subject to a weak secant constraint, yielding a closed-form update that modifes a single diagonal entry. Global convergence is proved under standard Lipschitz assumptions. Numerical experiments on 102 test problems with dimensions up to 3000 show that SD-DQN reduces CPU time compared to SD, BFGS, and two existing DQN methods.

Conflicts of interest

The author declares that there are no conflicts of interest.

References

[1] A. Al-Siyabi, M. Al-Baali, *Improved diagonal hessian approximations for large-scale unconstrained optimization*, SQU J. Sci.**26(2)** (2021) 126–140.

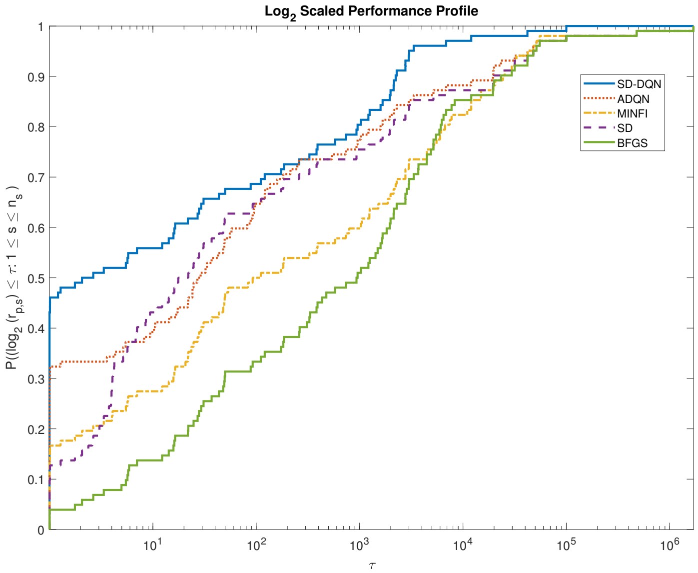

Figure 1: Dolan-Moré plot based on CPU time for problems with $n = 1000$

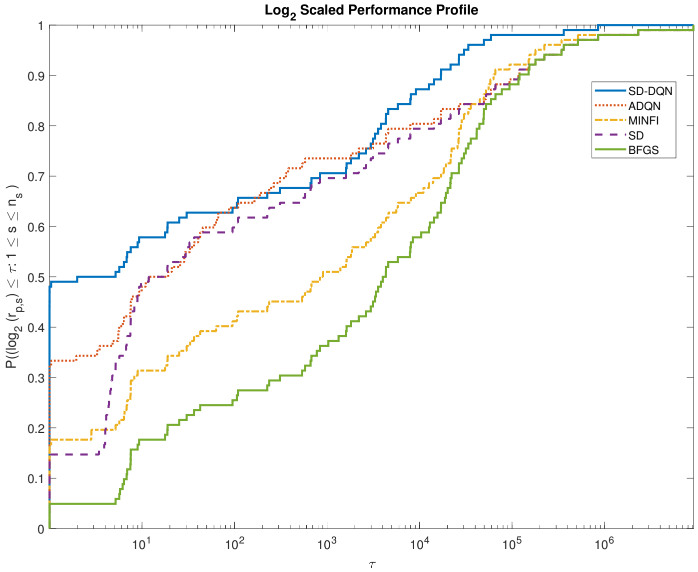

Figure 2: Dolan-Moré plot based on CPU time for problems with $n = 2000$

[2] A. Al-Siyabi, M. Al-Baali, *Diagonal quasi-Newton updating methods for large-scale nonlinear least squares*, International Journal of Research and Analytical Reviews **8(4)** (2024) 841–847.

[3] N. Andrei, *A diagonal quasi-Newton updating method based on minimizing the measure function*

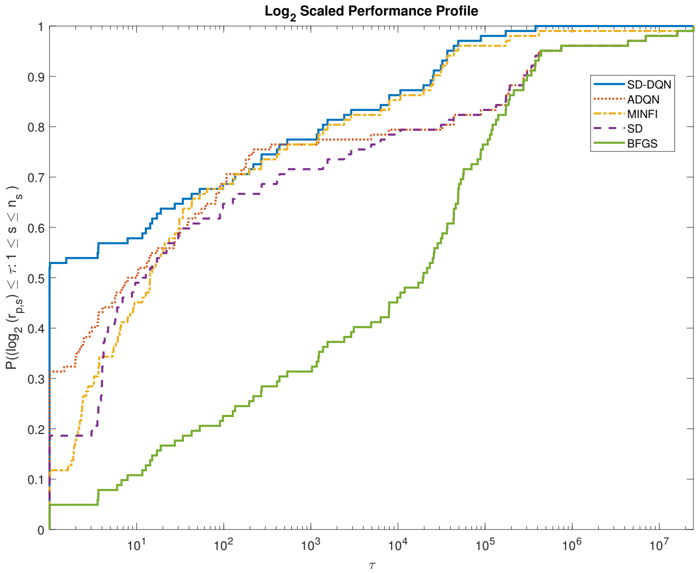

Figure 3: Dolan-Moré plot based on CPU time for problems with $n = 3000$

of Byrd and Nocedal for unconstrained optimization, Optim. **67** (2018) 1553–1568.

[4] N. Andrei, *An unconstrained optimization test functions collection*, Adv. Model. Optim. **10** (2008) 147–161.

[5] T. Bukar, W.J. Leong, M. Marjugi, C.Y. Chen, H.S. Sim, *Diagonal quasi-Newton updating strategy with Cholesky factor via variational principle*, Menemui Mat. **44(1)** (2022) 14–22.

[6] E.D. Dolan, J.J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program. **91** (2002) 201–213.

[7] S. Enshaei, W.J. Leong, M. Farid, *Diagonal quasi-Newton method via variational principle under generalized Frobenius norm*, Optim. Meth. Soft. **31(6)** (2016) 1258–1271.

[8] A. Mokhtari, A. Ingber, *A diagonal-augmented quasi-Newton method with application to factorization machines*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, (2017) 2671–2675.

[9] J. Nocedal, S.J. Wright, *Numerical Optimization*, Springer Ser. Oper. Res., 2006.

[10] M. Nosrati, K. Amini, *A new diagonal quasi-Newton algorithm for unconstrained optimization problems*, Appl. Math. **69(4)** (2024) 501–512.

[11] W.J. Leong, S. Enshaei, S.L. Kek, *Diagonal quasi-Newton methods via least change updating principle with weighted Frobenius norm*, Numer. Alg. **86** (2021) 1225–1241.