AIMS Mathematics, 11(11): 38083–38108.
DOI: 10.3934/math.20261471
Received: 16 July 2026
Revised: 02 September 2026
Accepted: 09 September 2026
Published: 16 November 2026

http://www.aimspress.com/journal/Math

Research article

A structured family of Dai-Liao conjugate gradient methods with an adaptive parameter for nonlinear least squares

Bashir Danladi Garba[1,2], Mansur Hassan[2], Nooraini Zainuddin[3,*], Hadil Alhazmi[4], Abdulhadi Aminu[1], Nazreen Waeleh[5] and Rabiu Bashir Yunus[1]

[1] Department of Mathematics, Faculty of Computing and Mathematical Sciences, Aliko Dangote University of Science and Technology, Wudil, Kano, Nigeria

[2] Department of Mathematics, Faculty of Science, Northwest University Kano, Kano State, Nigeria

[3] Department of Applied Science, Faculty of Science, Management and Computing, Universiti Teknologi PETRONAS, Seri Iskandar, Perak, Malaysia

[4] Department of Mathematical Sciences, College of Science, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia

[5] Department of Electronic Engineering, Faculty of Electronics and Computer Technology and Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, Durian Tunggal, Melaka, Malaysia

* **Correspondence:** Email: aini_zainuddin@utp.edu.my.

Abstract: Nonlinear least-squares (NLS) problems arise in many scientific and engineering applications, motivating the development of efficient and reliable iterative methods, particularly for large-scale problems. Conjugate gradient (CG) methods are attractive because of their low memory requirements, but their performance depends strongly on the choice of conjugacy parameters and the use of curvature information. In this paper, we have proposed a structured family of the Dai-Liao conjugate gradient (SFDL) method for solving NLS problems. The proposed method combines a structured secant relation based on higher-order information from the residual functions with an adaptively selected Dai-Liao parameter derived from a spectral approximation of the inverse Hessian. A safeguard mechanism ensures the required curvature condition and the sufficient descent property of the search direction. Under standard assumptions, the global convergence of the proposed method was established. Numerical experiments on classical and large-scale NLS test problems showed that the SFDL method is competitive with existing structured CG methods in terms of iteration counts and computational time. Furthermore, experiments on three classical Moré-Garbow-Hillstrom data-fitting problems demonstrated its robustness and accuracy on both well-conditioned and severely ill-conditioned problems.

Keywords: nonlinear least squares; Dai-Liao condition; structured secant condition; global convergence; data-fitting

Mathematics Subject Classification: 65K05, 90C52, 90C53, 90C56

1. Introduction

Nonlinear least-squares (NLS) problems can be described as follows:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(x), \tag{1.1}$$

where the function $f(x)$ is defined by

$$f(x) = \frac{1}{2}\|r(x)\|^2 = \frac{1}{2}\sum_{i=1}^{m}(r_i(x))^2, \qquad x \in \mathbb{R}^n, \tag{1.2}$$

where $r(x) = (r_1(x), r_2(x), \ldots, r_m(x))^\top$ denotes the residual vector. NLS has recently gained increasing recognition for its outstanding performance in various applications in computational science and engineering, including robotic motion control [1, 2], image restoration [3], parametric identification [4], data fitting [5], and meteorological applications [6]. Recent advances in intelligent optimization, control, and data-driven computational methods have demonstrated the growing importance of adaptive and learning-based strategies in complex engineering applications. For instance, dynamic event-triggered control with dynamic authority allocation has been developed for human-machine cooperative systems, providing an effective mechanism for adaptive decision-making and coordination [7, 8]. Similarly, adaptive dynamic programming has been employed for approximate optimal consensus control of unknown nonlinear multi-agent systems, highlighting the potential of adaptive optimization techniques in uncertain environments [9]. Furthermore, recent studies have explored graph-neural-network-driven multidimensional optimization for secure unmanned aerial vehicle (UAV)-satellite communications and collaborative fusion networks for real-time detection of extremely small objects in intelligent sports analytics [10]. These developments demonstrate the broad applicability of adaptive optimization and intelligent computational techniques across control, communication, diagnosis, and vision-related applications [11, 12]. Nevertheless, the development of efficient and robust optimization algorithms for NLS problems remains an important research direction, motivating the methodology proposed in this study. NLS problems constitute a special class of unconstrained optimization problems. Therefore, general unconstrained minimization techniques can be employed to generate a sequence of iterates x_k through

$$x_{k+1} = x_k + \alpha_k d_k. \tag{1.3}$$

The stepsizes $\alpha_k > 0,\ k \geq 0$, are obtained using a line search scheme in the direction of the search vector d_k. The search direction d_k is generated using the conjugate gradient (CG) method defined by

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{if } k \geq 1, \end{cases} \tag{1.4}$$

where β_k denotes the scalar coefficient, commonly known as the CG parameter, while $g(x) = \nabla f(x)$ represents the gradient of the function f. However, the choice of β_k plays a crucial role in the computational efficiency of CG methods, which has motivated extensive research to develop effective formulas for the CG parameter [13]. In addition, the direction d_k described in Eq (1.4) is usually

required to satisfy the sufficient descent condition, which stipulates that there is a constant $c > 0$ such that

$$g_k^\top d_k \leq -c\|g_k\|^2.$$

Due to the unique structure inherent in the NLS objective function, its gradient vector and Hessian matrix can be expressed as follows:

$$\nabla f(x) := \sum_{i=1}^{m} r_i(x)\nabla r_i(x) = J(x)^\top r(x), \tag{1.5}$$

$$\begin{aligned}\nabla^2 f(x) &:= \sum_{i=1}^{m} \nabla r_i(x)\nabla r_i(x)^\top + \sum_{i=1}^{m} r_i(x)\nabla^2 r_i(x) \\ &= J(x)^\top J(x) + Q(x).\end{aligned} \tag{1.6}$$

Here, $J(x)$ denotes the Jacobian matrix of $r(x)$ and

$$Q(x) = \sum_{i=1}^{m} r_i(x)\nabla^2 r_i(x)$$

represents the second term in (1.6).

The Gauss-Newton [14] and the Levenberg-Marquardt method [15] both discard $Q(x)$, replacing $\nabla^2 f$ by $J^\top J$. This works well on small-residual (zero-residual limit) problems, where $Q(x^*) = 0$ at a solution x^*, but converges poorly on large-residual (nonzero-residual) problems where $Q(x^*)$ is significant.

To overcome this limitation, Dennis et al. [16] proposed a structured secant equation that approximates the complete Hessian while retaining the structure of the product. Their secant vector encodes first-order derivative information. Subsequent studies extended this approach to higher-order Taylor corrections. Zhang et al. [17] proposed a second-order structured secant condition incorporating a cubic correction term. Nosrati and Amini [18] recently introduced a modified extended Hessian (MEH) secant equation with an adaptive blending parameter ρ_k that ensure the positivity condition $s_{k-1}^\top y_{k-1} > 0$, and proved the global convergence of the resulting spectral CG method.

A closer examination reveals that existing structured CG methods for NLS fall short along two independent and largely unaddressed dimensions. First, in terms of the structured secant relation itself: Kobayashi et al. [19] and Dehghani and Mahdavi-Amiri [20] relied on classical, unstructured, or first-order secant equations that do not separate the Gauss-Newton term $J^\top J$ from the second-order residual-curvature term $Q(x)$. This is adequate near a zero-residual solution but produces an $O(\|s_{k-1}\|^2)$ truncation error on large-residual problems, where $Q(x^*) \neq 0$ dominates the true Hessian and the omitted curvature information can substantially slow or even prevent convergence. Zhang et al. [17] and Nosrati and Amini [18] improved on this by incorporating higher-order (cubic) corrections, but the corrections are blended in through a fixed or ad hoc safeguard rather than one tied to the optimality of the resulting search direction. Second, in terms of the conjugacy parameter: Even where a structured secant condition is used, the associated CG parameter β_k in [19–24] was calculated using a fixed Hestenes-Stiefel (HS)-type formula whose scale is not adaptively matched to the local curvature encoded by the secant pair. Yunus et al. [21–24] established global convergence for their respective

structured variants, but each requires either an explicit safeguarding step, a restart strategy, or manually tuned constants in [22] to enforce sufficient descent; the parameter itself carries no guarantee of being an optimal scaling of the inverse Hessian. Dehghani and Mahdavi-Amiri [20] introduced a spectral scaling, but it multiplies a classical (not structured) secant pair, so the scaling corrects the step size without correcting the underlying curvature representation.

Consequently, prior work addressed either the accuracy of the curvature approximation or the adaptivity of the conjugacy parameter, but never both jointly. This separation matters computationally, not just theoretically: Because the secant pair and the parameter are tuned independently, the resulting search direction typically requires an additional correction or safeguard to guarantee sufficient descent (as in [18, 21–24]), adding both algebraic complexity and computational overhead per iteration. The SFDL method closes this gap by deriving the Dai-Liao parameter t_k directly from a spectral (BB2-type) estimate of the inverse Hessian applied to the higher-order structured secant pair itself. This coupling produces an exact algebraic cancellation in the sufficient descent inequality that holds automatically, without any external safeguard, restart, or manually tuned constant, a property none of [17–24] possesses. The proposed SFDL method merges a higher-order secant condition with an adaptively chosen DL parameter. That parameter comes from a spectral estimate of the inverse Hessian and guarantees sufficient descent. Global convergence is proved under the standard assumptions, and numerical experiments on a set of classical test problems illustrate the competitiveness of the method against existing structured CG approaches. The remainder of the paper is organized as follows. Section 2 derives the modified structured secant relation that captures higher-order residual information, presents the proposed SFDL method and the derivation of the adaptive parameter, and proves the sufficient descent property. Section 3 establishes the global convergence of the SFDL method. Section 4 reports the numerical results on the benchmark test functions and three nonlinear data-fitting problems, and Section 5 concludes the paper and outlines directions for future work.

2. Derivation of a modified structured secant relation

This section develops the SFDL method in three parts. Subsection 2.1 derives a modified structured secant relation from a third-order Taylor expansion of the residual functions, correcting the truncation error present in earlier first- and second-order structured secant equations. Subsection 2.2 builds on this relation to derive the SFDL parameter and the adaptive parameter t_k using a Barzilai-Borwein-type spectral estimate. Finally, Subsection 2.3 presents the algorithm of the proposed method.

2.1. Higher-order Taylor expansions

Let $s_{k-1} = x_k - x_{k-1}$. We expand each residual component r_i around x_{k-1} in the direction s_{k-1}:

$$\begin{aligned} r_i(x_k) = r_i(x_{k-1}) + \nabla r_i(x_{k-1})^\top s_{k-1} + \frac{1}{2}\, s_{k-1}^\top \nabla^2 r_i(x_{k-1})\, s_{k-1} \\ + \frac{1}{6}\, s_{k-1}^\top \nabla^3 r_i(x_{k-1})[s_{k-1}, s_{k-1}] + O(\|s_{k-1}\|^4), \end{aligned} \tag{2.1}$$

$$r_i(x_k) = r_i(x_{k-1}) + \nabla r_i(x_{k-1})^\top s_{k-1} + \frac{1}{2}\, s_{k-1}^\top \nabla^2 r_i(x_{k-1})\, s_{k-1} + \frac{1}{6}\, D_i + O(\|s_{k-1}\|^4), \tag{2.2}$$

where

$$D_i = s_{k-1}^\top \nabla^3 r_i(x_{k-1})[s_{k-1},\ s_{k-1}], \tag{2.3}$$

which is the third-order directional derivative of r_i at x_{k-1} in the direction of s_{k-1}, and the gradient expansion is

$$\nabla r_i(x_k) = \nabla r_i(x_{k-1}) + \nabla^2 r_i(x_{k-1})\, s_{k-1} + \frac{1}{2} z_i + O(\|s_{k-1}\|^4), \tag{2.4}$$

where

$$z_i = \nabla^3 r_i(x_{k-1})[s_{k-1},\ s_{k-1}].$$

Taking the inner product with $s_{k-1}^\top$ gives the following:

$$s_{k-1}^\top \nabla r_i(x_k) = s_{k-1}^\top \nabla r_i(x_{k-1}) + s_{k-1}^\top \nabla^2 r_i(x_{k-1})\, s_{k-1} + \frac{1}{2} D_i + O(\|s_{k-1}\|^4), \tag{2.5}$$

since $s_{k-1}^\top z_i = D_i$.

We define the following notations:

$$A_i = r_i(x_k) - r_i(x_{k-1}) - \nabla r_i(x_{k-1})^\top s_{k-1}, \tag{2.6}$$

$$B_i = s_{k-1}^\top [\nabla r_i(x_k) - \nabla r_i(x_{k-1})], \tag{2.7}$$

$$C_i = s_{k-1}^\top \nabla^2 r_i(x_{k-1})\, s_{k-1}. \tag{2.8}$$

From (2.2) and (2.5), we obtain

$$A_i = \frac{1}{2} C_i + \frac{1}{6} D_i + O(\|s_{k-1}\|^4), \tag{2.9}$$

$$B_i = C_i + \frac{1}{2} D_i + O(\|s_{k-1}\|^4). \tag{2.10}$$

Multiplying (2.9) by 6 and (2.10) by 2, we have

$$6A_i = 3C_i + D_i + O(\|s_{k-1}\|^4), \tag{2.11}$$

$$2B_i = 2C_i + D_i + O(\|s_{k-1}\|^4). \tag{2.12}$$

Subtracting to eliminate D_i and solving for C_i:

$$C_i = s_{k-1}^\top \nabla^2 r_i(x_{k-1})\, s_{k-1} = 6A_i - 2B_i + O(\|s_{k-1}\|^4). \tag{2.13}$$

Suppose a simple diagonal approximation of $\nabla^2 r_i(x_k)$ is used, i.e., if we require $\nabla^2 r_i(x_k) \approx \alpha_i I$. Following the work of Yahaya et al. [1], the requirement becomes

$$C_i = 6A_i - 2B_i + O(\|s_{k-1}\|^4) \approx \alpha_i s_{k-1}^\top s_{k-1}.$$

This gives

$$\alpha_i \approx \frac{C_i}{\|s_{k-1}\|^2}, \tag{2.14}$$

and thus, the approximation $\nabla^2 r_i(x_k) s_{k-1}$ is then given as

$$\nabla^2 r_i(x_k)\, s_{k-1} \approx \frac{C_i}{\|s_{k-1}\|^2}\, s_{k-1}. \tag{2.15}$$

Recall that

$$(x_k)s_{k-1} = \sum_i r_i(x_k)\nabla^2 r_i(x_k)s_{k-1},$$

and summing over i:

$$\begin{aligned} Q(x_k)\, s_{k-1} &= \sum_{i=1}^{m} r_i(x_k)\nabla^2 r_i(x_k)\, s_{k-1} \\ &\approx \frac{\sum_{i=1}^{m} r_i(x_k)C_i}{\|s_{k-1}\|^2}\, s_{k-1}, \end{aligned}$$

$$Q(x_k)\, s_{k-1} \approx \frac{\vartheta_k}{\|s_{k-1}\|^2}\, s_{k-1}, \tag{2.16}$$

where the structured curvature scalar is

$$\vartheta_k = \sum_{i=1}^{m} r_i(x_k)\, C_i = \sum_{i=1}^{m} r_i(x_{k+1})(6A_i - 2B_i). \tag{2.17}$$

Substituting (2.6) and (2.7) into (2.17) and using $\sum_i r_i(x_k)\nabla r_i(x_{k-1})^\top s_{k-1} = r_k^\top J_{k-1}s_{k-1}$:

$$6\sum_i r_i(x_k)\, A_i = 6[\|r_k\|^2 - r_k^\top r_{k-1} - r_k^\top J_{k-1}s_{k-1}], \tag{2.18}$$

$$2\sum_i r_i(x_k)\, B_i = 2r_k^\top (J_k - J_{k-1})s_{k-1}. \tag{2.19}$$

Combining (2.18)–(2.19), we have

$$\vartheta_k = 6\|r_k\|^2 - 6\, r_k^\top r_{k-1} - 4\, r_k^\top J_{k-1}s_{k-1} - 2\, r_k^\top J_k s_{k-1}. \tag{2.20}$$

Combining the Gauss-Newton term with the approximation (2.16):

$$\nabla^2 f(x_k)\, s_{k-1} \approx J_k^\top J_k\, s_{k-1} + \frac{\vartheta_k}{\|s_{k-1}\|^2}\, s_{k-1}. \tag{2.21}$$

We define the structured secant vector as follows:

$$\hat{y}_{k-1} = J_k^\top J_k\, s_{k-1} + \rho_k \frac{\vartheta_k}{\|s_{k-1}\|^2}\, s_{k-1}, \tag{2.22}$$

with an adaptive guard parameter $\rho_k \in \{0, 1\}$:

$$\rho_k = \begin{cases} 1, & \text{if } \|J_k s_{k-1}\|^2 + \vartheta_k \geq \omega\|s_{k-1}\|^2, \\ 0, & \text{otherwise,} \end{cases} \tag{2.23}$$

for a small constant $\omega > 0$. The structured secant condition for the approximate Hessian G_k is

$$G_k s_{k-1} = \hat{y}_{k-1}. \tag{2.24}$$

Remark 2.1. *The safeguard parameter ρ_k ensures that*

$$s_{k-1}^{\top}\hat{y}_{k-1} > 0$$

at every iteration, which is required for the CG parameter β_k to be well-defined. When $\rho_k = 0$, the correction is suppressed and the secant vector reduces to the Gauss-Newton form

$$y_k = J_k^{\top} J_k s_{k-1},$$

for which positivity follows from the full column rank assumption on J_k.

2.2. Derivation of the SFDL and adaptive parameter t_k

2.2.1. The Dai-Liao conjugacy condition

Since its introduction, the Dai-Liao conjugacy condition has motivated extensive research on the selection of t_k; see Babaie-Kafaki [25] for a comprehensive survey. Generalizing the standard conjugacy requirement $d_k^{\top}\hat{y}_{k-1} = 0$, Dai and Liao [26] imposed

$$d_k^{\top}\hat{y}_{k-1} = -t_k\, g_k^{\top} s_{k-1}, \qquad t_k \geq 0, \tag{2.25}$$

where t_k is a nonnegative parameter and $\hat{y}_{k-1}$ is the structured secant vector (2.22). Substituting

$$d_k = -g_k + \beta_k d_{k-1}$$

and solving for β_k, we have

$$\beta_k^{SFDL}(t_k) = \frac{g_k^{\top}(\hat{y}_{k-1} - t_k s_{k-1})}{d_{k-1}^{\top}\hat{y}_{k-1}}. \tag{2.26}$$

We now derive t_k from the first principles.

2.2.2. Optimal inverse-Hessian scaling

If $G_k s_{k-1} = \hat{y}_{k-1}$ and G_k is invertible, then

$$G_k^{-1}\hat{y}_{k-1} = s_{k-1}$$

A simple choice is $G_k^{-1} \approx \tau I$ with $\tau > 0$. We picked τ to minimize $\|\tau\hat{y}_{k-1} - s_{k-1}\|^2$.

Let

$$\phi(\tau) = \|\tau\hat{y}_{k-1} - s_{k-1}\|^2. \tag{2.27}$$

By expanding the expression above, we obtain the following:

$$\phi(\tau) = \tau^2\|\hat{y}_{k-1}\|^2 - 2\tau\, s_{k-1}^{\top}\hat{y}_{k-1} + \|s_{k-1}\|^2. \tag{2.28}$$

Differentiating with respect to τ, it follows that

$$\phi'(\tau) = 2\tau\|\hat{y}_{k-1}\|^2 - 2s_{k-1}^{\top}\hat{y}_{k-1}. \tag{2.29}$$

Therefore, setting $\phi'(\tau) = 0$, we obtain the following result:

$$\tau_k = \frac{s_{k-1}^{\top}\hat{y}_{k-1}}{\|\hat{y}_{k-1}\|^2}. \tag{2.30}$$

Proposition 2.2 (Optimal scalar approximation of G_k^{-1})**.** *The unique minimizer of*

$$\phi(\tau) = \|\tau\hat{y}_{k-1} - s_{k-1}\|^2$$

over $\tau > 0$ *is*

$$\tau_k = \frac{s_{k-1}^\top \hat{y}_{k-1}}{\|\hat{y}_{k-1}\|^2}, \tag{2.31}$$

which follows directly from the stationary condition $\phi'(\tau) = 0$ *derived above.*

Remark 2.3. τ_k *is the well-known second Barzilai-Borwein (BB2) step [27]; consequently,* $t_k = 1/\tau_k$ *coincides with the Dai-Kou [28] spectral parameter, here evaluated on the structured pair* $(s_{k-1}, \hat{y}_{k-1})$ *rather than the classical* (s_{k-1}, y_{k-1})*.*

The Dai-Liao condition (2.25) is motivated by the quasi-Newton approximation $G_k d_k \approx -t_k g_k$, equivalently, $d_k \approx -t_k G_k^{-1} g_k$. Under the spectral approximation $G_k^{-1} \approx \tau_k I$ from Proposition 2.2, this becomes

$$d_k \approx -t_k \tau_k g_k. \tag{2.32}$$

The product $t_k\tau_k$ is the effective step scale. Imposing the unit quasi-Newton normalization $t_k\tau_k = 1$ (so that $d_k \approx -g_k$ is the baseline quasi-Newton step) gives

$$t_k = \frac{1}{\tau_k}. \tag{2.33}$$

Substituting (2.30) into (2.33), we obtain the following result:

$$t_k = \frac{\|\hat{y}_{k-1}\|^2}{s_{k-1}^\top \hat{y}_{k-1}}. \tag{2.34}$$

Thus, t_k is the adaptive Dai-Liao parameter for the SFDL method. Now, substituting (2.34) into (2.26) gives the SFDL conjugate parameter:

$$\beta_k^{\mathrm{SFDL}} = \frac{g_k^\top \hat{y}_{k-1}}{d_{k-1}^\top \hat{y}_{k-1}} - \frac{\|\hat{y}_{k-1}\|^2 g_k^\top s_{k-1}}{(s_{k-1}^\top \hat{y}_{k-1})(d_{k-1}^\top \hat{y}_{k-1})}. \tag{2.35}$$

Both terms involve only the gradient g_k and the secant pair $(s_{k-1}, \hat{y}_{k-1})$; no additional information is required. The following exact cancelation, which permits sufficient descent without the need for an external correction term, is the main algebraic consequence of the adaptive choice (2.34).

2.3. The SFDL algorithm

The proposed method is summarized in Algorithm 1.

Algorithm 1 Structured family Dai-Liao conjugate gradient method (SFDL).

Require: Starting point $x_0 \in \mathbb{R}^n$; constants $\omega \in (0, \lambda_{\min})$, $0 < \delta < \sigma < 1$; tolerance $\varepsilon > 0$.
Ensure: Approximate solution x_k.

Set $k \leftarrow 0$.
Compute $r_0 = r(x_0)$, $J_0 = J(x_0)$, $g_0 = J_0^\top r_0$, and set $d_0 = -g_0$.
while $\|g_k\|_\infty > \varepsilon(1 + |f_k|)$ **do**
    Find $\alpha_k > 0$ satisfying the strong Wolfe conditions:

$$f(x_k + \alpha_k d_k) \le f(x_k) + \delta \alpha_k g_k^\top d_k,$$
$$\left| g(x_k + \alpha_k d_k)^\top d_k \right| \le -\sigma g_k^\top d_k.$$

    Set

$$x_{k+1} = x_k + \alpha_k d_k.$$

    Compute

$$r_k = r(x_k), \qquad J_k = J(x_k), \qquad g_k = J_k^\top r_k.$$

    Compute ϑ_k via (2.20) and ρ_k via (2.23).
    Form the structured secant vector $\hat{y}_{k-1}$ via (2.22).
    Compute t_k via (2.34).
    Compute the SFDL conjugate parameter β_k^{SFDL} via (2.35).
    Set

$$d_k = -g_k + \beta_k^{\mathrm{SFDL}} d_{k-1}.$$

    Set $k \leftarrow k + 1$.
end while
return x_k.

3. Global convergence analysis

In this section, we present the global convergence properties of the proposed SFDL method. The analysis is carried out under the following assumptions.

Assumption 3.1.
(1) *The level set* $\mathcal{S} = \{x \in \mathbb{R}^n : f(x) \le f(x_0)\}$ *is bounded.*
(2) $J(x)$ *has full column rank on* $\mathcal{S}$*, so* $J(x)^\top J(x)$ *is positive definite with the smallest eigenvalue* $\lambda_{\min} > 0$ *uniformly on* $\mathcal{S}$*.*
(3) *There exists an open neighborhood* $\mathcal{U} \supseteq \mathcal{S}$ *and a constant* $L > 0$ *such that*

$$\|J(x) - J(z)\| \le L\|x - z\|$$

for all $x, z \in \mathcal{U}$.

From Assumption 3.1, there exists $\gamma > 0$ such that, $\forall\, x \in \mathcal{S}$:

$$\|r(x)\| \le \gamma, \quad \|J(x)\| \le \gamma, \quad \|g(x)\| = \|J(x)^\top r(x)\| \le \gamma^2. \tag{3.1}$$

Also, by the mean-value theorem, $\|r(x) - r(z)\| \le \gamma\|x - z\|$ for all $x, z \in \mathcal{S}$.

Lemma 3.2. *Under Assumption 3.1, there exists a constant $C_\vartheta > 0$ such that*

$$|\vartheta_k| \le C_\vartheta \|s_{k-1}\|^2, \quad \forall\, k \ge 0. \tag{3.2}$$

Proof. Rewrite (2.20) using $s_{k-1} = x_k - x_{k-1}$:

$$\vartheta_k = 6r_k^\top (r_k - r_{k-1} - J_k s_{k-1}) + 4r_k^\top (J_k - J_{k-1}) s_{k-1}, \tag{3.3}$$

obtained by adding and subtracting $6r_k^\top J_k s_{k-1}$ in (2.20).
First term. Fix $i \in \{1, \dots, m\}$. By the mean-value theorem applied to r_i along the segment joining x_{k-1} and x_k, there exists $\xi_i \in (x_{k-1}, x_k)$ such that

$$r_i(x_k) - r_i(x_{k-1}) = \nabla r_i(\xi_i)^\top s_{k-1}.$$

Since $J_k = J(x_k)$ has the i-th row $\nabla r_i(x_k)^\top$, the i-th component of $J_k s_{k-1}$ is $\nabla r_i(x_k)^\top s_{k-1}$, so

$$\begin{aligned} \left[r_k - r_{k-1} - J_k s_{k-1}\right]_i &= r_i(x_k) - r_i(x_{k-1}) - \nabla r_i(x_k)^\top s_{k-1} \\ &= \left(\nabla r_i(\xi_i) - \nabla r_i(x_k)\right)^\top s_{k-1}. \end{aligned}$$

By Assumption 3.1 and $\xi_i \in (x_{k-1}, x_k)$,

$$\|\xi_i - x_k\| \le \|s_{k-1}\|,$$

so

$$\left|[r_k - r_{k-1} - J_k s_{k-1}]_i\right| \le L\|\xi_i - x_k\|\, \|s_{k-1}\| \le L\|s_{k-1}\|^2.$$

Hence,

$$\|r_k - r_{k-1} - J_k s_{k-1}\| \le \sqrt{m}\, L\|s_{k-1}\|^2,$$

and by the Cauchy-Schwarz and inequality (3.1):

$$\begin{aligned} |6r_k^\top (r_k - r_{k-1} - J_k s_{k-1})| &\le 6\|r_k\|\, \|r_k - r_{k-1} - J_k s_{k-1}\| \\ &\le 6\gamma\, \sqrt{m}\, L\|s_{k-1}\|^2. \end{aligned} \tag{3.4}$$

Second term. By 3.1, $\|J_k - J_{k-1}\| \le L\|s_{k-1}\|$, so

$$|4r_k^\top (J_k - J_{k-1}) s_{k-1}| \le 4\gamma L\|s_{k-1}\|^2. \tag{3.5}$$

Combining (3.3)–(3.5):

$$|\vartheta_k| \le (6\sqrt{m} + 4)\gamma L\|s_{k-1}\|^2.$$

Set $C_\vartheta = (6\sqrt{m} + 4)\gamma L$. □

Lemma 3.3. *Suppose that Assumption 3.1 holds. Then there exists $M_1 > 0$ such that*

$$\|\hat{y}_{k-1}\| \le M_1 \|s_{k-1}\|, \quad \forall\, k \ge 1. \tag{3.6}$$

Proof.
Case $\rho_k = 1$. By the definition of $\hat{y}_{k-1}$, (3.1), and Lemma 3.2:

$$\begin{aligned}\|\hat{y}_{k-1}\| &\le \|J_k^\top J_k\, s_{k-1}\| + \frac{|\vartheta_k|}{\|s_{k-1}\|^2}\,\|s_{k-1}\| \\ &\le \gamma^2\|s_{k-1}\| + C_\vartheta\|s_{k-1}\| \\ &= (\gamma^2 + C_\vartheta)\|s_{k-1}\|.\end{aligned}$$

Case $\rho_k = 0$. $\hat{y}_{k-1} = J_k^\top J_k s_{k-1}$, so

$$\|\hat{y}_{k-1}\| \le \|J_k\|^2\|s_{k-1}\| \le \gamma^2\|s_{k-1}\|.$$

Setting $M_1 = \gamma^2 + C_\vartheta$ covers both cases. □

Lemma 3.4 (Positivity and the lower bound of $s_{k-1}^\top \hat{y}_{k-1}$)**.** *Suppose that Assumption 3.1 holds. Then there exists* $M_2 > 0$ *such that*

$$s_{k-1}^\top \hat{y}_{k-1} \ge M_2\|s_{k-1}\|^2 > 0, \quad \forall\, k \ge 0. \tag{3.7}$$

Consequently, $d_k^\top \hat{y}_{k-1} > 0$ *for all* $k \ge 0$*, and the adaptive parameter* t_k *is well-defined.*

Proof.
Case $\rho_k = 1$. The trigger condition in (2.23) gives

$$\|J_k s_{k-1}\|^2 + \vartheta_k \ge \omega\|s_{k-1}\|^2,$$

so

$$s_{k-1}^\top \hat{y}_{k-1} = \|J_k s_{k-1}\|^2 + \vartheta_k \ge \omega\|s_{k-1}\|^2.$$

Case $\rho_k = 0$. $\hat{y}_{k-1} = J_k^\top J_k s_{k-1}$, so

$$s_{k-1}^\top \hat{y}_{k-1} = \|J_k s_{k-1}\|^2 \ge \lambda_{\min}\|s_{k-1}\|^2 \ge \omega\|s_{k-1}\|^2,$$

where $\lambda_{\min} > \omega$ by assumption. Setting $M_2 = \min\{\omega, \lambda_{\min}\} = \omega > 0$ covers both cases.

Since $s_{k-1} = \alpha_k d_k$ with $\alpha_k > 0$,

$$d_k^\top \hat{y}_{k-1} = \alpha_k^{-1} s_{k-1}^\top \hat{y}_{k-1} \ge M_2\alpha_k^{-1}\|s_{k-1}\|^2 > 0.$$

This also shows t_k has a strictly positive denominator; hence, it is well-defined and positive. □

Corollary 3.5. *Under Assumption 3.1, the angle* θ_k *between* s_{k-1} *and* $\hat{y}_{k-1}$ *satisfies*

$$\cos\theta_k \ge \frac{M_2}{M_1} > 0 \quad \textit{for all } k,$$

where M_1 *and* M_2 *are the constants from Lemmas 3.3 and 3.4.*

Proof. This follows immediately by combining Lemmas 3.3 and 3.4 with the definition of $\cos\theta_k$.

Corollary 3.6. *For any $k \geq 1$ and any $\alpha_{k-1} > 0$, the t_k contribution in $g_k^\top d_k$ satisfies*

$$\frac{\|\hat{y}_{k-1}\|^2 \, (g_k^\top s_{k-1}) \, (g_k^\top d_{k-1})}{(s_{k-1}^\top \hat{y}_{k-1}) \, (d_{k-1}^\top \hat{y}_{k-1})} = \frac{\|\hat{y}_{k-1}\|^2 \, (g_k^\top d_{k-1})^2}{(d_{k-1}^\top \hat{y}_{k-1})^2}. \tag{3.8}$$

Proof. Since $s_{k-1} = \alpha_{k-1} d_{k-1}$ with $\alpha_{k-1} > 0$, both $g_k^\top s_{k-1} = \alpha_{k-1}(g_k^T d_{k-1})$ and $s_{k-1}^\top \hat{y}_{k-1} = \alpha_{k-1}(d_{k-1}^\top \hat{y}_{k-1})$ (the latter being strictly positive by Lemma 3.4). The left side of (3.8) becomes

$$\frac{\|\hat{y}_{k-1}\|^2 \cdot \alpha_{k-1} \, (g_k^\top d_{k-1}) \cdot (g_k^\top d_{k-1})}{\alpha_{k-1} \, (d_{k-1}^\top \hat{y}_{k-1}) \cdot (d_{k-1}^\top \hat{y}_{k-1})} = \frac{\|\hat{y}_{k-1}\|^2 \, (g_k^\top d_{k-1})^2}{(d_{k-1}^\top \hat{y}_{k-1})^2}.$$

□

The following lemma shows that the search direction of the SFDL method satisfies the sufficient descent condition.

Lemma 3.7. *The search direction d_k generated by Algorithm 1 satisfies*

$$g_k^T d_k \leq -\left(\frac{3}{4}\right)\|g_k\|^2, \quad \forall \, k > 0.$$

Proof. For $k = 0$, the inequality holds by the initialization rule of the method. Now, assume $k \geq 1$. From Eq (1.4), we have

$$d_k = -g_k + \beta_k^{\text{SFDL}} d_{k-1}.$$

Multiplying both sides by $g_k^\top$ gives

$$g_k^\top d_k = -\|g_k\|^2 + \beta_k^{\text{SFDL}}(g_k^\top d_{k-1}).$$

Using the definition of β_k^{SFDL} from (2.35) (with $\hat{y}_{k-1}$ in place of y_{k-1}), we obtain

$$g_k^\top d_k = -\|g_k\|^2 + \frac{(g_k^\top \hat{y}_{k-1})(g_k^\top d_{k-1})}{d_{k-1}^\top \hat{y}_{k-1}} - \frac{\|\hat{y}_{k-1}\|^2 \, (g_k^\top s_{k-1})(g_k^\top d_{k-1})}{(s_{k-1}^\top \hat{y}_{k-1})(d_{k-1}^\top \hat{y}_{k-1})}. \tag{3.9}$$

By applying the inequality

$$a^T b \leq \frac{1}{2}(\|a\|^2 + \|b\|^2)$$

with

$$a = \frac{1}{\sqrt{2}} g_k,$$

$$b = \frac{\sqrt{2}(g_k^\top d_{k-1})}{d_{k-1}^\top \hat{y}_{k-1}} \, \hat{y}_{k-1},$$

it follows that

$$\frac{(g_k^\top \hat{y}_{k-1})(g_k^\top d_{k-1})}{d_{k-1}^\top \hat{y}_{k-1}} \leq \frac{1}{4}\|g_k\|^2 + \frac{\|\hat{y}_{k-1}\|^2 \, (g_k^\top d_{k-1})^2}{(d_{k-1}^\top \hat{y}_{k-1})^2}. \tag{3.10}$$

Substituting (3.10) into (3.9) gives

$$g_k^\top d_k \le -\|g_k\|^2 + \frac{1}{4}\|g_k\|^2 + \frac{\|\hat{y}_{k-1}\|^2 \left(g_k^\top d_{k-1}\right)^2}{\left(d_{k-1}^\top \hat{y}_{k-1}\right)^2} - \frac{\|\hat{y}_{k-1}\|^2 \left(g_k^\top s_{k-1}\right)\left(g_k^\top d_{k-1}\right)}{\left(s_{k-1}^\top \hat{y}_{k-1}\right)\left(d_{k-1}^\top \hat{y}_{k-1}\right)}.$$

By using Corollary 3.6, the last two terms cancel, giving

$$g_k^\top d_k \le -\|g_k\|^2 + \frac{1}{4}\|g_k\|^2 = -\left(\frac{3}{4}\right)\|g_k\|^2.$$

Hence, the direction d_k satisfies the desired inequality. □

Lemma 3.8. *Suppose that Assumption 3.1 holds. Then there exists a constant $M > 0$ such that*

$$\|d_k\| \le M, \quad \forall\, k \ge 0. \tag{3.11}$$

Proof. $k = 0$: $\|d_0\| = \|g_0\| \le \gamma^2$ by (3.1).

For $k \ge 1$, by the triangle inequality:

$$\|d_k\| \le \|g_k\| + |\beta_k^{\mathrm{SFDL}}|\|d_{k-1}\|. \tag{3.12}$$

We bound $|\beta_k^{\mathrm{SFDL}}|\|d_{k-1}\|$ uniformly. Write $\beta_k^{\mathrm{SFDL}} = T_1 - T_2$ from (2.35) with $\hat{y}_{k-1}$.

Denominator lower bounds: Since $s_{k-1} = \alpha_{k-1} d_{k-1}$, Lemma 3.4 (with index $k-1$) gives:

$$s_{k-1}^\top \hat{y}_{k-1} \ge M_2 \|s_{k-1}\|^2 = M_2\, \alpha_{k-1}^2 \|d_{k-1}\|^2, \tag{3.13}$$

$$d_{k-1}^\top \hat{y}_{k-1} = \alpha_{k-1}^{-1}\, s_{k-1}^\top \hat{y}_{k-1} \ge M_2\, \alpha_{k-1} \|d_{k-1}\|^2. \tag{3.14}$$

T_1: Using (3.1), Lemma 3.3, and (3.14):

$$|T_1|\|d_{k-1}\| = \frac{|g_k^\top \hat{y}_{k-1}|}{d_{k-1}^\top \hat{y}_{k-1}}\|d_{k-1}\| \le \frac{\gamma^2 M_1 \alpha_{k-1}\|d_{k-1}\|^2}{M_2 \alpha_{k-1}\|d_{k-1}\|^2} = \frac{M_1 \gamma^2}{M_2}. \tag{3.15}$$

T_2: Using (3.1), Lemma 3.3, (3.13), and (3.14):

$$\begin{aligned} |T_2|\|d_{k-1}\| &= \frac{\|\hat{y}_{k-1}\|^2 |g_k^\top s_{k-1}|}{(s_{k-1}^\top \hat{y}_{k-1})(d_{k-1}^\top \hat{y}_{k-1})}\|d_{k-1}\| \\ &\le \frac{M_1^2 \alpha_{k-1}^2 \|d_{k-1}\|^2 \cdot \gamma^2 \alpha_{k-1}\|d_{k-1}\| \cdot \|d_{k-1}\|}{M_2 \alpha_{k-1}^2 \|d_{k-1}\|^2 \cdot M_2 \alpha_{k-1}\|d_{k-1}\|^2} \\ &= \frac{M_1^2 \gamma^2}{M_2^2}. \end{aligned} \tag{3.16}$$

Adding (3.15) and (3.16): $|\beta_k^{\mathrm{SFDL}}|\|d_{k-1}\| \le C_\beta$, where $C_\beta = \gamma^2(M_1/M_2 + M_1^2/M_2^2)$ is independent of k. Substituting into (3.12):

$$\|d_k\| \le \gamma^2 + C_\beta.$$

Set $M = \max\{\gamma^2,\ \gamma^2 + C_\beta\} = \gamma^2 + C_\beta$, which completes the proof. □

Theorem 3.9. *Suppose that f is continuously differentiable and bounded below on the level set S, and that $\{x_k\}$ is generated by an iterative scheme* (1.3) *satisfying the strong Wolfe conditions, with $g_k^\top d_k < 0$ for all k. If*

$$\sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} = \infty,$$

then

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

The following theorem demonstrates that Algorithm 1 is globally convergent under the stated assumptions.

Theorem 3.10. *Suppose Assumption 3.1 holds, and let $\{x_k\}$ be generated by Algorithm 1. Then*

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

Proof. It follows immediately from Lemma 3.8 that $\|d_k\| \leq M$ for all $k \geq 0$, for some constant $M > 0$. Hence

$$\sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} \geq \sum_{k=0}^{\infty} \frac{1}{M^2} = \infty.$$

Since Lemma 3.7 guarantees

$$g_k^\top d_k \leq -\frac{3}{4}\|g_k\|^2 < 0$$

for all $k > 0$, and α_k satisfies the strong Wolfe conditions by the construction of Algorithm 1, Theorem 3.9 applies directly, giving

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

□

4. Numerical experiments and applications

4.1. Benchmark problems

This section presents the numerical results obtained from solving NLS problems using the proposed SFDL algorithm. A series of numerical experiments was conducted to assess the effectiveness of the proposed method. In total, 27 benchmark test problems were considered. The details of the test problems, including their initial points and corresponding references, are provided in Table 1. Each problem was solved for five different dimensions, namely, 3000, 6000, 9000, 12,000, and 15,000, resulting in a total of 135 test instances. The performance of the proposed SFDL method was compared with those of Dehghani and Mahdavi-Amiri [20], SSCG in [18], SSHS in [24], and the SRMILCG method of Yunus et al. [22].

Table 1. List of test problems: starting points and corresponding references.

Problems	Test functions	Starting point	References
1	Penalty Function I	$x_0 = (3, \ldots, 3)^\top$	[29]
2	Trigonometric Function	$x_0 = (1/n, \ldots, 1/n)^\top$	[29]
3	Discrete Boundary Value Problem	$x_0 = \left(\frac{\frac{1}{n+1}-1}{n+1}, \frac{\frac{2}{n+1}-1}{n+1}, \ldots, \frac{\frac{n}{n+1}-1}{n+1}\right)^\top$	[30]
4	Linear Function – Full Rank	$x_0 = (1, \ldots, 1)^\top$	[30]
5	P202 (Hock & Schittkowski)	$x_0 = (25, 5, -5, 1)^\top$	[31]
6	P206 (Hock & Schittkowski)	$x_0 = (1, 1, 1)^\top$	[31]
7	P212 (Hock & Schittkowski)	$x_0 = (\frac{1}{2}, \ldots, \frac{1}{2})^\top$	[31]
8	Almost Rosenbrock	$x_0 = (\frac{1}{n}, \ldots, \frac{1}{n})^\top$	[30]
9	Raydan 1	$x_0 = (1, \ldots, 1)^\top$	[32]
10	Sine Function 2	$x_0 = (\frac{1}{n}, \ldots, \frac{1}{n})^\top$	[30]
11	Exponential Function 1	$x_0 = (\frac{n}{n-1}, \ldots, \frac{n}{n-1})^\top$	[30]
12	Exponential Function 2	$x_0 = (\frac{1}{n^2}, \ldots, \frac{1}{n^2})^\top$	[30]
13	Logarithmic Function	$x_0 = (1, \ldots, 1)^\top$	[30]
14	Trigonometric-Exponential Function	$x_0 = (1, \ldots, 1)^\top$	[30]
15	Extended Freudenstein & Roth	$x_0 = (0.5, -2.0, 0.5, -2.0, \ldots)^\top$	[33]
16	Extended Powell Singular Function	$x_0 = (1, \ldots, 1)^\top$	[30]
17	Function 21	$x_0 = (-1, \ldots, -1)^\top \in \mathbb{R}^{6n/5}$	[29]
18	Broyden Tridiagonal Function	$x_0 = (-1, \ldots, -1)^\top$	[34]
19	Generalized Broyden Tridiagonal	$x_0 = (-1, \ldots, -1)^\top$	[34]
20	Extended Rosenbrock	$x_0 = (-1.2, 1.0, -1.2, 1.0, \ldots)^\top$	[35]
21	Extended Himmelblau Function	$x_0 = (0, \ldots, 0)^\top$	[36]
22	Function 27	$x_0 = (3, -1, 0, 1, \ldots)^\top$	[37]
23	Trigonometric-Logarithmic Function	$x_0 = (1, \ldots, 1)^\top$	[30]
24	Zero Jacobian Function	$x_0 = \left(\frac{100(n-100)}{n}, \underbrace{\frac{(n-1000)(n-500)}{(60n)^2}, \ldots, \frac{(n-1000)(n-500)}{(60n)^2}}_{n-1 \text{ entries}}\right)^\top$	[30]
25	Exponential Function	$x_0 = (\frac{1}{2}, \ldots, \frac{1}{2})^\top$	[30]
26	Extended Wood Function	$x_0 = (-3, -1, -3, -1, \ldots)^\top$	[38]
27	Sine Function 1	$x_0 = (1/n, \ldots, 1/n)^\top$	[30]

We carried out the experiment using MATLAB R2023a on a personal computer with an Intel (R) CORE(TM) i7-3537U processor running at 2.00 GHz and having 8 GB of RAM. The algorithms were executed with the specified configuration of parameters:

- SFDL algorithm: $\delta = 10^{-4}$, $\sigma = 0.9$, and $\omega = 10^{-8}$.
- Dehghani algorithm: The parameter settings for this method remained unchanged, as in [20].
- Nosrati algorithm: The parameter settings for this method remained unchanged, as in [18].
- SSHS algorithm: The parameter settings for this method remained unchanged, as in [24].
- SRMILCG algorithm: $\epsilon = 10^{-4}$, $\eta_{min} = 0.1$, $\eta_{max} = 0.85$, $\kappa = 10^{-8}$ [22].

Furthermore, throughout the iterative process, the stopping criterion $\|g_k\| \leq 10^{-4}$ was used. An algorithm was considered to have successfully converged once this condition was satisfied. Conversely, a failure was denoted by "F" whenever any of the following conditions was encountered:

(1) The number of iterations exceeded 1000.
(2) The number of function evaluations exceeded 5000.

To assess the performance of the proposed algorithm, the following performance metrics were considered: (i) the number of iterations (Iter), (ii) the number of function evaluations (Feval), (iii) the number of gradient evaluations (Ngrad), (iv) the CPU time (CPU) required to obtain an approximate solution, and (v) the residual function value (ValueF). The complete numerical results are available at: https://github.com/Rabiu-Bashir/Numerical-Results-SFDL-Method.

Furthermore, to provide a more comprehensive comparison of the competing methods, the results are further analyzed using the well-known performance profile proposed by Dolan and Moré [39]. This graphical tool facilitates a standardized and objective comparison of optimization algorithms. Let S denote the set of solvers and P the set of test problems. For a given solver $s \in S$ and problem $p \in P$, let $b_{p,s}$ represent a performance measure, such as the number of iterations, the number of function evaluations, the number of gradient evaluations, the CPU time, or the residual value. The performance ratio of the solver s on the problem p is defined as

$$r_{p,s} = \frac{b_{p,s}}{\min\{b_{p,s} : s \in S\}}, \tag{4.1}$$

which measures the performance of the solver s relative to the best-performing solver for the same problem. The performance profile of a solver s is then given by

$$\rho_s(\tau) = \frac{1}{|P|} \text{size}\left\{p \in P : \log_2(r_{p,s}) \leq \tau\right\}, \tag{4.2}$$

where $\tau \in \mathbb{R}^+$ is a prescribed performance factor. The function $\rho_s(\tau)$ represents the probability that the solver s performs within a factor τ of the best solver in the set of test problems. According to Dolan and Moré [39], a solver with a larger value of $\rho_s(\tau)$ is considered more efficient and robust. In particular, the solver whose performance profile attains the highest probability is regarded as the overall best-performing method.

The performance profiles corresponding to the considered metrics are presented in Figures 1–5. A careful examination of these figures reveals that the proposed SFDL algorithm consistently exhibits superior performance when compared with the competing methods, namely, Dehghani, SSCG, SSHS, and SRMILCG. Specifically, Figure 1 shows that, for $\tau \geq 1$, the SFDL method solves approximately 88% of the test problems with the fewest iterations, outperforming Dehghani, SSCG, SSHS, and SRMILCG, which solve about 65%, 60%, 55%, and 72% of the problems, respectively. Similarly, Figure 2 demonstrates that SFDL achieves the highest probability of success in terms of the number of function evaluations. At $\tau \geq 2$, SFDL successfully solves nearly 92% of the test problems with the least number of function evaluations, whereas SRMILCG, Dehghani, SSCG, and SSHS solve approximately 68%, 62%, 58%, and 52% of the problems, respectively.

A comparable trend is observed in Figure 3 for the number of gradient evaluations. The SFDL algorithm solves nearly 90% of the test problems with the fewest gradient evaluations, while SRMILCG, Dehghani, SSCG, and SSHS achieve approximately 70%, 60%, 55%, and 50%, respectively. Furthermore, the CPU-time performance profile in Figure 4 highlights the computational efficiency of the proposed method. For $\tau \geq 2$, SFDL successfully solves about 91% of the test problems in the shortest computational time, compared with approximately 73%, 61%, 57%, and

53% for SRMILCG, Dehghani, SSCG, and SSHS, respectively. The residual-value performance profile shown in Figure 5 further confirms the effectiveness of the proposed approach. The SFDL method attains the highest probability of achieving the smallest residual values, indicating that its iterates converge to highly accurate approximations of the solutions. Although SRMILCG exhibits competitive performance and generally ranks as the second-best solver, SFDL consistently maintains a clear advantage across all performance measures. Here, τ denotes the proportion of problems for which a solver achieves the best residual value. Performance profile analysis confirms that SFDL outperforms all competing methods in every evaluated metric. By consistently requiring fewer iterations, function and gradient evaluations, and less computational time while delivering more precise solutions, SFDL proves to be a practically effective and memory-efficient solver for large-scale NLS problems.

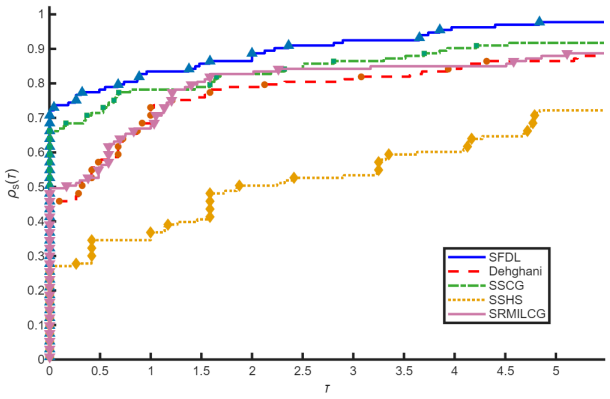


Figure 1. Performance profile based on the number of iterations.

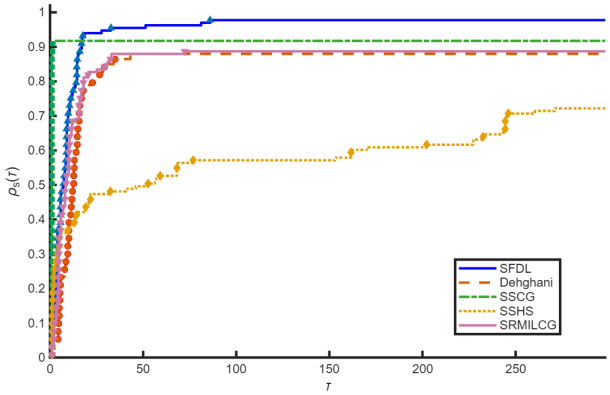


Figure 2. Performance profile based on feval.

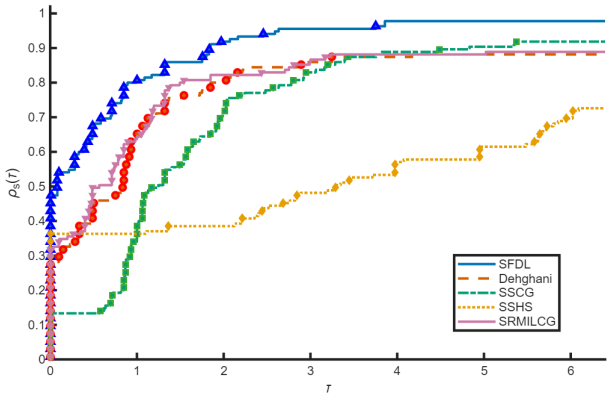


Figure 3. Performance profile based on the number of gradient evaluations.

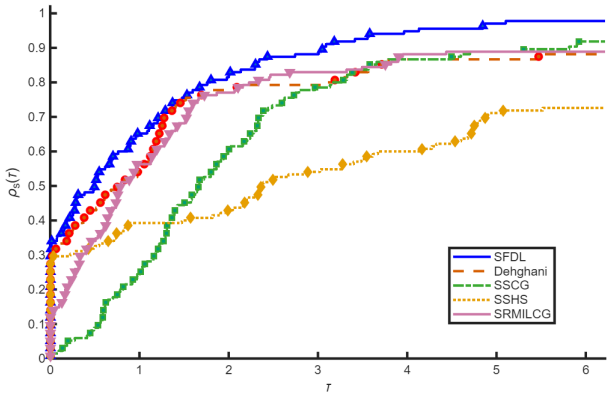


Figure 4. Performance profile based on CPU time.

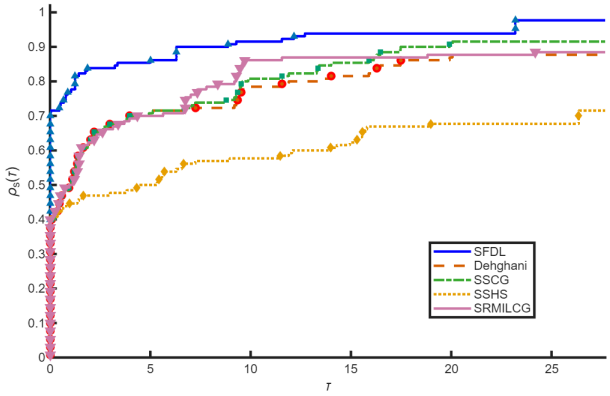


Figure 5. Performance profile based on the function value.

4.2. Real-world data-fitting applications

The fundamental idea behind any NLS problem is to determine n unknown parameters $x = [x_1, x_2, \ldots, x_n]^\top$ of a nonlinear fitting model $y = M(x, t)$ that provides the best fit to a set of m observed data points $(t_1, y_1), (t_2, y_2), \ldots, (t_m, y_m)$, with $m > n$ [40]. The best fit is achieved when the residual functions $r_i(x)$, defined as the discrepancy between each observed data point y_i and the corresponding value predicted by the fitted model $M(x, t_i)$ [5],

$$r_i(x) = y_i - M(x, t_i), \qquad i = 1, 2, \ldots, m, \tag{4.3}$$

are collectively minimized in the least-squares sense. That is, x^* solves

$$\min_{x\in\mathbb{R}^n} \; f(x) = \frac{1}{2}\sum_{i=1}^{m} r_i(x)^2 = \frac{1}{2}\,\|r(x)\|^2,$$

where $r(x) = [r_1(x), r_2(x), \ldots, r_m(x)]^\top$ is the residual vector and $J(x) = \nabla r(x)^\top \in \mathbb{R}^{m\times n}$ is its Jacobian.

To further assess the effectiveness and practical applicability of the SFDL method, we consider three classical NLS data-fitting problems selected from the well-known Moré-Garbow-Hillstrom (MGH) test collection [29]. All numerical experiments were performed using the standard MGH initial points reported in [29]. The stopping tolerance was set to $\varepsilon = 10^{-6}$, while the strong Wolfe line-search parameters were chosen as $\delta = 10^{-4}$ and $\sigma = 0.9$. In addition, the positivity safeguard constant was fixed at $\omega = 10^{-8}$, and the maximum number of iterations was limited to 5000.

4.2.1. Gaussian function (MGH Problem 9)

The first experiment considers the Gaussian peak-fitting problem, which involves unknown parameters' $n = 3$ and $m = 15$ residuals. The nonlinear model is defined by

$$m(x, t) = x_1 \exp\!\left(-\frac{1}{2}x_2(\theta_t - x_3)^2\right), \qquad \theta_t = \frac{8 - t}{2}, \qquad t = 1, \ldots, 15. \tag{4.4}$$

Here, x_1 represents the peak amplitude, x_2 controls the inverse squared width of the Gaussian curve, and x_3 denotes the peak location. The observed data correspond to a bell-shaped symmetric distribution centered at $\theta = 0$ (equivalently, $t = 8$), with values varying from approximately 0.0009 at the tails to 0.3989 at the peak. The algorithm was initialized with the standard starting point

$$x_0 = (0.398,\ 1.0,\ 2.5)^\top.$$

The SFDL method converged successfully to the approximate minimizer

$$x^* \approx (0.3989,\ 1.0000,\ 0.0000)^\top,$$

which agrees closely with the reference solution reported in [29]. The resulting fitted Gaussian profile interpolates the data points accurately and yields a residual norm of approximately $\|r(x^*)\| \approx 1.13 \times 10^{-3}$, indicating that the problem is essentially a zero-residual problem. The quantity of structured curvature ϑ_k effectively captures the local curvature information along the search direction, enabling the SFDL update parameter β_k^{SFDL} to exploit the intrinsic least-squares structure and, thereby, accelerate convergence. The graphical representation of the fitting result is presented in Figure 6.

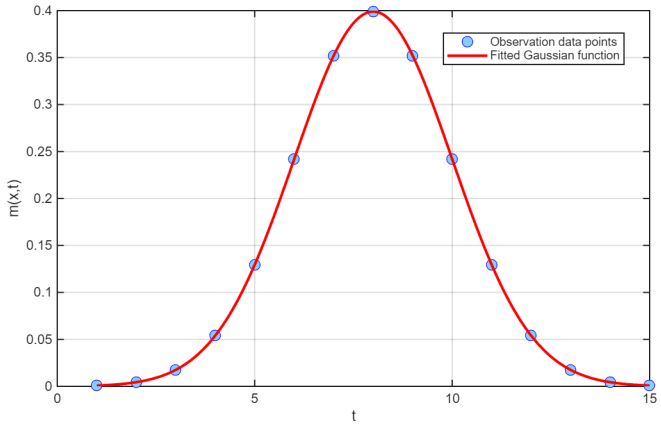


Figure 6. A nonlinear Gaussian fitting problem in which the red curve denotes the Gaussian model with $n = 3$ unknown parameters fitted to $m = 15$ observed data points.

4.2.2. Osborne II function (MGH Problem 20)

The second test problem is the Osborne II function, which is regarded as one of the most difficult problems in the MGH collection. This problem contains $n = 11$ unknown parameters and $m = 65$ observations. The underlying model is given by

$$m(x,t) = x_1 e^{-x_5 t} + x_2 e^{-x_6(t-x_9)^2} + x_3 e^{-x_7(t-x_{10})^2} + x_4 e^{-x_8(t-x_{11})^2}, \tag{4.5}$$

where

$$t_i = \frac{i-1}{10}, \qquad i = 1, \ldots, 65.$$

The first exponential term represents a decaying component, while the remaining three terms correspond to Gaussian profiles centered at x_9, x_{10}, and x_{11}, respectively. Their widths are controlled by the parameters x_6, x_7, and x_8. Due to its high dimensionality, multiple Gaussian components, and nearly flat objective landscape during initial iterations, this problem provides a demanding benchmark for structured NLS algorithms. The SFDL method was initialized at

$$x_0 = (1.3,\ 0.65,\ 0.65,\ 0.7,\ 0.6,\ 3.0,\ 5.0,\ 7.0,\ 2.0,\ 4.5,\ 5.5)^\top.$$

Starting from this point, the algorithm converged successfully and produced a solution satisfying

$$\|r(x^*)\| \approx 0.4014,$$

which is consistent with the benchmark value reported in [29]. In this experiment, the adaptive safeguard parameter ρ_k plays an essential role. Whenever the second-order correction term ϑ_k fails to satisfy the positivity condition

$$s_{k-1}^\top \hat{y}_{k-1} \geq \omega \|s_{k-1}\|^2,$$

the correction is automatically suppressed by setting $\rho_k = 0$. Consequently, the secant vector reduces to the stable Gauss-Newton approximation:

$$\hat{y}_{k-1} = J_k^\top J_k s_{k-1}.$$

This mechanism ensures the well-definedness of β_k^{SFDL} throughout the highly oscillatory early iterations and contributes significantly to the robustness of the method. Figure 7 illustrates the fitting performance of the SFDL method for the Osborne II problem.

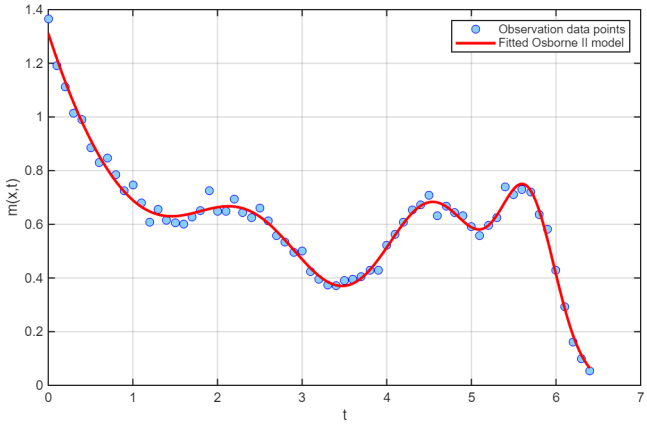


Figure 7. A nonlinear fitting of the Osborne II function, where the red curve represents the fitted Osborne model with $n = 11$ parameters (variables) and $m = 65$ observed data points.

4.2.3. Meyer function (MGH Problem 10)

The third example considers the Meyer thermocouple calibration problem, which consists of $n = 3$ parameters and $m = 16$ experimental measurements. The nonlinear model is expressed as

$$m(x,t) = x_1 \exp\left(\frac{x_2}{t_i + x_3}\right), \qquad t_i = 45 + 5i, \quad i = 1, \ldots, 16. \tag{4.6}$$

The measured electromotive-force data decrease from approximately $34,780$ mV at $t = 50°$C to 2872 mV at $t = 125°$C. The Meyer problem is widely recognized as severely ill-conditioned because the Jacobian matrix becomes nearly singular in the neighborhood of the solution. As a result, many conventional gradient-based algorithms either stagnate or diverge when starting from the standard initial estimate

$$x_0 = (0.02,\ 4000,\ 250)^\top.$$

Despite the strong ill-conditioning, the SFDL method converged successfully to the approximate solution

$$x^* \approx (5.61 \times 10^{-3},\ 6181.3,\ 345.2)^\top,$$

with the residual norm

$$\|r(x^*)\| \approx 0.8790,$$

which agrees with the reference solution in [29]. A major contributing factor to this performance is the adaptive Dai-Liao scaling parameter which automatically adjusts the CG correction according to the local inverse-Hessian geometry. In regions of large curvature, such as those near the nearly singular Jacobian of the Meyer function, the parameter t_k becomes relatively large, thus reducing the influence of the previous search direction and maintaining a stable descent trajectory. The resulting Meyer data-fitting profile is depicted in Figure 8.

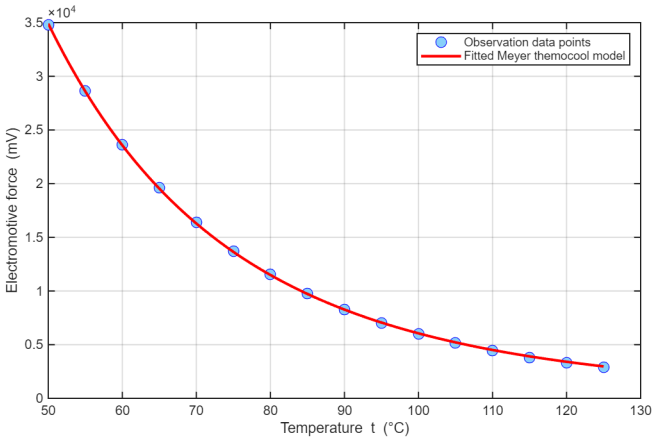


Figure 8. A nonlinear fitting of the Meyer function, where the red curve represents the fitted Meyer model with $n = 3$ parameters and $m = 16$ observed data points.

Table 2 summarizes the numerical performance of the SFDL method on the three nonlinear data-fitting problems. In all cases, the algorithm successfully converged to solutions that agree closely with the known MGH reference values.

Table 2. Performance of the SFDL method on nonlinear data-fitting problems.

Problem	n	m	Iter	Feval	$\|g^*\|_\infty$	$\|r(x^*)\|$
Gaussian (MGH 9)	3	15	60	147	$< 10^{-6}$	1.13×10^{-3}
Osborne II (MGH 20)	11	65	695	2686	$< 10^{-6}$	0.4014
Meyer (MGH 10)	3	16	52	111	$< 10^{-6}$	0.8790

Iter denotes the number of outer iterations, while Feval represents the total number of function evaluations. The above experiments collectively demonstrate several important properties of the proposed SFDL method when applied to NLS problems.

5. Conclusions

This paper proposed the structured family of the Dai-Liao (SFDL) CG method for NLS problems. The method combines a higher-order structured secant equation with an adaptive spectral Dai-Liao parameter, yielding a parameter-free search direction that satisfies the sufficient descent condition without additional safeguarding. Under standard assumptions, the global convergence of the proposed method was established. Extensive numerical experiments on benchmark NLS and MGH data-fitting problems demonstrated that SFDL consistently outperforms existing structured CG methods in terms of iteration count, function and gradient evaluations, computational time, and solution accuracy. These results indicate that SFDL provides an efficient, robust, and memory-efficient approach to solving large-scale NLS problems. A limitation of the proposed method is that the structured curvature correction relies on a simplifying diagonal Hessian approximation, which may be less effective on problems with highly coupled curvature. Future work will focus on extending the proposed adaptive framework to the two-parameter weighted Dai-Liao conjugacy condition, developing three-term hybrid

variants, and investigating applications to constrained and stochastic NLS problems arising in robotics, image restoration, and large-scale data fitting.

Author contributions

Bashir Danladi Garba: Conceptualization, methodology, software, writing—original draft; Mansur Hassan: Supervision, investigation, software, validation, writing—review and editing; Nooraini Zainuddin: Methodology, supervision, funding acquisition; Hadil Alhazmi: Investigation, analysis, methodology, resources; Abdulhadi Aminu: Methodology, supervision, analysis, validation; Nazreen Waeleh: Software, funding acquisition, writing—original draft; Rabiu Bashir Yunus: Conceptualization, investigation, resources, supervision, validation. All authors have read and approved the final version of the manuscript for publication.

Funding statements

This project received funding from Diamond Gas Holding (DGH) for a research project with cost center 015MFA-B10 at Universiti Teknologi PETRONAS, Malaysia, and in collaboration with Universiti Teknikal Malaysia Melaka (UTeM) under the fellowship scheme.

Acknowledgements

This is to acknowledge the support of the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R528), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors declare that they have no competing interests to disclose.

References

1. M. M. Yahaya, P. Kumam, A. M. Awwal, P. Chaipunya, S. Aji, S. Salisu, A new generalized quasi-Newton algorithm based on structured diagonal Hessian approximation for solving nonlinear least-squares problems with application to 3DOF planar robot arm manipulator, *IEEE Access*, **10** (2022), 10816–10826. https://doi.org/10.1109/ACCESS.2022.3144875
2. L. Meng, Y. Hao, Q. Yuan, Y. Zhan, D. Zhang, Safe human-centric shared control for consumer service robots in smart home environments via observer-based ADP, *IEEE Trans. Consum. Electron.*, 2026. https://doi.org/10.1109/TCE.2026.3718874

3. S. Henn, A Levenberg-Marquardt scheme for nonlinear image registration, *BIT Numer. Math.*, **43** (2003), 743–759. https://doi.org/10.1023/B:BITN.0000009940.58397.98
4. G. Rojas-Dueñas, J. R. Riba, M. Moreno-Eguilaz, Nonlinear least squares optimization for parametric identification of DC-DC converters, *IEEE Trans. Power Electron.*, **36** (2021), 654–661. https://doi.org/10.1109/TPEL.2020.3003075
5. M. M. Yahaya, P. Kumam, P. Chaipunya, A. M. Awwal, L. Wang, On diagonally structured scheme for nonlinear least squares and data-fitting problems, *RAIRO Oper. Res.*, **58** (2024), 2887–2905. https://doi.org/10.1051/ro/2024102
6. R. M. Passi, Use of nonlinear least squares in meteorological applications, *J. Appl. Meteorol. Climatol.*, **16** (1977), 827–832.
7. D. Zhang, L. Meng, L. Liang, C. Qin, D. Liu, Dynamic event-triggered control for human-machine cooperative systems based on dynamic authority allocation, *IEEE Trans. Syst. Man Cybern. Syst.*, **56** (2026), 3733–3744. https://doi.org/10.1109/TSMC.2026.3661716
8. D. Zhang, Y. Hao, Q. Yuan, C. Qin, Dynamic event-triggered approximate optimal consensus control for unknown nonlinear multi-agent systems via adaptive dynamic programming, *ISA Trans.*, **172** (2026), 21–32. https://doi.org/10.1016/j.isatra.2026.03.010
9. H. Wu, C. Fan, D. Zhang, An innovative neural network architecture designed for industrial fault diagnosis with hierarchical adaptive attention mechanism, *Process Saf. Environ. Prot.*, **205** (2026), 108162. https://doi.org/10.1016/j.psep.2025.108162
10. L. Liang, P. Xiang, H. Huang, N. Zhang, P. Qi, Z. Yin, et al., Sec-GNN-driven joint multidimensional anti-eavesdropping optimization for secure UAV-satellite communications, *IEEE Internet Things J.*, **13** (2026), 28694–28706. https://doi.org/10.1109/JIOT.2026.3685201
11. D. Zhang, Q. Yuan, L. Meng, R. Xia, W. Liu, C. Qin, Reinforcement learning for single-agent to multi-agent systems: from basic theory to industrial application progress, a survey, *Artif. Intell. Rev.*, **59** (2026), 46. https://doi.org/10.1007/s10462-025-11439-9
12. D. Zhang, C. Fan, Z. Li, C. Qin, J. Dan, Scale-spectrum-state collaborative fusion network for real-time extreme small object detection in intelligent sports analytics, *Expert Syst. Appl.*, **333** (2026), 133831. https://doi.org/10.1016/j.eswa.2026.133831
13. J. Nocedal, S. J. Wright, *Numerical optimization*, New York: Springer, 2006. https://doi.org/10.1007/978-0-387-40065-5
14. H. O. Hartley, The modified Gauss-Newton method for the fitting of non-linear regression functions by least squares, *Technometrics*, **3** (1961), 269–280. https://doi.org/10.1080/00401706.1961.10489945
15. D. W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *J. Soc. Ind. Appl. Math.*, **11** (1963), 431–441. https://doi.org/10.1137/0111030
16. J. E. Dennis Jr., D. M. Gay, R. E. Welsch, An adaptive nonlinear least-squares algorithm, *ACM Trans. Math. Softw.*, **7** (1981), 348–368. https://doi.org/10.1145/355958.355965
17. J. Z. Zhang, Y. Xue, K. Zhang, A structured secant method based on a new quasi-Newton equation for nonlinear least squares problems, *BIT Numer. Math.*, **43** (2003), 217–229. https://doi.org/10.1023/A:1023665409152

18. M. Nosrati, K. Amini, A new structured spectral conjugate gradient method for nonlinear least squares problems, *Numer. Algorithms*, **97** (2024), 897–914. https://doi.org/10.1007/s11075-023-01729-0
19. M. Kobayashi, Y. Narushima, H. Yabe, Nonlinear conjugate gradient methods with structured secant condition for nonlinear least squares problems, *J. Comput. Appl. Math.*, **234** (2010), 375–397. https://doi.org/10.1016/j.cam.2009.12.031
20. R. Dehghani, N. Mahdavi-Amiri, Scaled nonlinear conjugate gradient methods for nonlinear least squares problems, *Numer. Algorithms*, **82** (2019), 1–20. https://doi.org/10.1007/s11075-018-0591-2
21. R. B. Yunus, A. Ben Ghorbal, N. Zainuddin, S. M. Ibrahim, An accelerated diagonally structured CG algorithm for nonlinear least squares and inverse kinematics, *Mathematics*, **13** (2025), 2766. https://doi.org/10.3390/math13172766
22. R. B. Yunus, A. R. El-Saeed, N. Zainuddin, H. Daud, A structured RMIL conjugate gradient-based strategy for nonlinear least squares with applications in image restoration problems, *AIMS Math.*, **10** (2025), 14893–14916. https://doi.org/10.3934/math.2025668
23. R. B. Yunus, N. Zainuddin, H. Daud, R. Kannan, M. M. Yahaya, A structured Liu-Storey-type conjugate gradient approach for nonlinear least-squares with application to a 3DOF robot model, *J. Adv. Res. Appl. Sci. Eng. Technol.*, **64** (2025), 187–200. https://doi.org/10.37934/araset.64.187200
24. R. B. Yunus, N. Zainuddin, H. Daud, R. Kannan, S. A. Abdul Karim, M. M. Yahaya, A modified structured spectral HS method for nonlinear least squares problems and applications in robot arm control, *Mathematics*, **11** (2023), 3215. https://doi.org/10.3390/math11143215
25. S. Babaie-Kafaki, A survey on the Dai-Liao family of nonlinear conjugate gradient methods, *RAIRO Oper. Res.*, **57** (2023), 43–58. https://doi.org/10.1051/ro/2022213
26. Y. H. Dai, L. Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient methods, *Appl. Math. Optim.*, **43** (2001), 87–101. https://doi.org/10.1007/s002450010019
27. J. Barzilai, J. M. Borwein, Two-point step size gradient methods, *IMA J. Numer. Anal.*, **8** (1988), 141–148. https://doi.org/10.1093/imanum/8.1.141
28. Y. H. Dai, C. X. Kou, A nonlinear conjugate gradient algorithm with an optimal property and an improved Wolfe line search, *SIAM J. Optim.*, **23** (2013), 296–320. https://doi.org/10.1137/100813026
29. J. J. Moré, B. S. Garbow, K. E. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Softw.*, **7** (1981), 17–41. https://doi.org/10.1145/355934.355936
30. N. I. M. Gould, D. Orban, P. L. Toint, CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization, *Comput. Optim. Appl.*, **60** (2015), 545–557. https://doi.org/10.1007/s10589-014-9687-3
31. W. Hock, K. Schittkowski, *Test examples for nonlinear programming codes*, Berlin, Heidelberg: Springer-Verlag, 1981. https://doi.org/10.1007/978-3-642-48320-2
32. M. Raydan, The Barzilai and Borwein gradient method for the large-scale unconstrained minimization problem, *SIAM J. Optim.*, **7** (1997), 26–33. https://doi.org/10.1137/S1052623494266365

33. B. Roth, F. Freudenstein, Synthesis of path-generating mechanisms by numerical methods, *J. Eng. Ind.*, **85** (1963), 298–304. https://doi.org/10.1115/1.3669870

34. C. G. Broyden, A class of methods for solving nonlinear simultaneous equations, *Math. Comp.*, **19** (1965), 577–593. https://doi.org/10.1090/S0025-5718-1965-0198670-6

35. H. H. Rosenbrock, An automatic method for finding the greatest or least value of a function, *Comput. J.*, **3** (1960), 175–184. https://doi.org/10.1093/comjnl/3.3.175

36. D. M. Himmelblau, *Applied nonlinear programming*, New York: McGraw-Hill, 1972.

37. M. J. D. Powell, A Fortran subroutine for solving systems of nonlinear algebraic equations, In: *Numerical methods for nonlinear algebraic equations*, New York: Gordon and Breach, 1970.

38. Wood, J. Non-linear least squares methods for parameter estimation with applications in reactor physics, *J. Inst. Nucl. Eng.*, **14**, (1973), 106–110, 113–121.

39. E. D. Dolan, J. J. Moré, Benchmarking optimization software with performance profiles, *Math. Program.*, **91** (2002), 201–213. https://doi.org/10.1007/s101070100263

40. C. N. L. Eu, H. G. Harno, K. H. Lim, Nonlinear least squares problems using approximate greatest descent method, *2022 International Conference on Computer and Drone Application*s, 2022, 101–106. https://doi.org/10.1109/ICONDA56696.2022.10000382



AIMS Press

© 2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0)