IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 5, 2022, accepted January 12, 2022, date of publication January 25, 2022, date of current version January 31, 2022.
*Digital Object Identifier 10.1109/ACCESS.2022.3144875*

# 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

**MAHMOUD MUHAMMAD YAHAYA[1,2], POOM KUMAM[1,2,3,4], (Member, IEEE), ALIYU MUHAMMED AWWAL[3,4], PARIN CHAIPUNYA[1,4], SANI AJI[1,5], AND SANI SALISU[1,6]**
[1]KMUTTFixed Point Research Laboratory, Room SCL 802 Fixed Point Laboratory, Science Laboratory Building, Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), Thung Khru, Bangkok 10140, Thailand
[2]Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), Thung Khru, Bangkok 10140, Thailand
[3]Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 40402, Taiwan
[4]KMUTTFixed Point Research Laboratory, Room SCL 802 Fixed Point Laboratory, Center of Excellence in Theoretical and Computational Science (TaCS-CoE), Science Laboratory Building, Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), Thung Khru, Bangkok 10140, Thailand
[5]Department of Mathematics, Faculty of Science, Gombe State University, Gombe 760214, Nigeria
[6]Department of Mathematics and Computer Science, Sule Lamido University, Kafin Hausa, Jigawa 741103, Nigeria

Corresponding author: Poom Kumam (poom.kum@kmutt.ac.th)

This work was supported in part by the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), King Mongkut's University of Technology, Thonburi (KMUTT); in part by the Center under Computational and Applied Science for Smart Innovation research Cluster (CLASSIC), Faculty of Science, KMUTT; in part by the National Council of Thailand (NRCT) through the Research Grants for Talented Mid-Career Researchers under Contract N41A640089; and in part by the Thailand Science Research and Innovation (TSRI) Basic Research Fund Fiscal Year 2022 (FF65). The work of Mahmoud Muhammad Yahaya was supported by the Petchra Pra Jom Klao Masters Research Scholarship from the KMUTT under Contract 5/2562.

**ABSTRACT** Many problems in science and engineering can be formulated as nonlinear least-squares (NLS) problems. Thus, the need for efficient algorithms to solve these problems can not be overemphasized. In that sense, we introduce a generalized structured-based diagonal Hessian algorithm for solving NLS problems. The formulation associated with this algorithm is essentially a generalization of a similar result in Yahaya *et al.* (Journal of Computational and Applied Mathematics, pp. 113582, 2021). However, in this work, the structured diagonal Hessian update is derived under a weighted Frobenius norm; this allows other choices of the weighted matrix analogous to the Davidon-Fletcher-Powell (DFP) method. Moreover, to theoretically fill the gap in Yahaya *et al.* (Journal of Computational and Applied Mathematics, pp. 113582, 2021), we have shown that the proposed algorithm is R-linearly convergent under some standard conditions devoid of any safeguarding strategy. Furthermore, we experimentally tested the proposed scheme on some standard benchmark problems in the literature. Finally, we applied this algorithm to solve robotic motion control problem consisting of 3DOF (degrees of freedom).

**INDEX TERMS** Nonlinear least squares, quasi-Newton, diagonal updating, least change secant, robotic motion control.

## I. INTRODUCTION

In this research article, we propose generalized structured-based quasi-Newton algorithm for nonlinear least-squares problems of the following form:

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

where the residual, $r_i : \mathbb{R}^n \to \mathbb{R}$ is a smooth function for each $i = 1, 2, \cdots, m$. We assume that for higher-dimensional

The associate editor coordinating the review of this manuscript and approving it for publication was Felix Albu.

problems, i.e., when (*n is large*), the Jacobian matrix of $r$, $J(x)^T$ is not stored explicitly; however, we can evaluate the matrix-vector product say, $J^T v$, where $v \in \mathbb{R}^m$. Moreover, the gradient, $g(x)$ and Hessian, $H(x)$ of $f$ are defined as follows:

$$g(x) = \sum_{i=1}^{m} r_i(x)\nabla r_i(x) = J(x)^T r(x), \tag{2}$$

and

$$\begin{aligned} H(x) &= \sum_{i=1}^{m} \nabla r_i(x)\nabla r_i(x)^T + \sum_{i=1}^{m} r_i(x)\nabla^2 r_i(x) \\ &= J(x)^T J(x) + Q(x). \end{aligned} \tag{3}$$

Algorithms for solving (1) are paramount because of their wide range of applications, since problems of the form (1) arise in robotic motion, imaging, parameter estimation, data fitting, and also when solving systems of nonlinear equations (for more information, kindly see [1]–[19]).

In recent times, there are some algorithms developed for solving (1) considering its structure. The approach adopted in formulating these algorithms is mostly toward approximating the action of the Hessian of (1) by a *structured vector*, $z \in \mathbb{R}^n$ which can be derived through Taylor series expansion of $r^i$ or it's gradient $g^i$, for $i = 1, 2, \cdots, m$ such that a secant condition, $Hs \approx z$ or weak secant condition, $s^T Hs \approx s^T z$ is satisfied, where $s$ is a difference between successive iterates. For instance, in [20] the authors approximate the Hessian in (3) with a scalar multiple of an identity matrix such that the secant condition is satisfied. They incorporated this approximation into the well-known Barzilai and Borwein [21] (BB) spectral parameters and their convex combinations, as reported in [22]. Similarly, although using a different paradigm, Mohammed and Santos [23] came up with diagonal-based approximations of Hessian's (3) first and second matrix terms. The derived approximations satisfied the modified secant condition. However, despite approximating these matrices in (3), their search directions require several safeguarding techniques before the sufficient descent condition is satisfied.

To mitigate some of the shortcomings of their proposal, recently, Yahaya *et al.* [24] proposed structured, quasi-Newton-based algorithms for solving (1). First, the two formulations of the *structured vector* were derived. Both derivations approximate only the second term of (3), where the first formulation is estimated using first-order Taylor series expansion. On the other hand, the second term is approximated to higher-order Taylor series expansion on $r^i$ and its $g^i$ for each $i = 1, 2, \cdots, m$ by using the Richardson extrapolation technique to get rid of the tensor terms. These derived formulations are such that a modified weak secant condition of Dennis and Wolkowicz [25] is satisfied. Thus, they used the formulations to develop two diagonal updating schemes. These are then independently used in generating the search directions. Interestingly, their algorithm requires fewer user-defined parameters in the search direction.

This paper used the formulation in [24] to derive a generalized diagonal updating mechanism using a weighted Frobenius norm defined as

$$\|A\|_W^2 = tr(W^{-1}AW^{-1}A^T),$$

for solving (1), where $A \in \mathbb{R}^{nxn}$, $tr(\cdot)$ is trace operator, and $W$ is a weighted matrix which changes at every update and often different choices of it, leads to other updates. Some well-known updates include Davidon-Fletcher-Powell (DFB) and Powell-Symmetric-Broyden (PSB). Motivated by the previous works, this paper also aims to fill in the gap of the recent work [24] by giving the rate of convergence results under some standard assumptions with the aid of an Armijo line search strategy.

Inspire by the work of Yahaya *et al.*, [24] this paper gives the following contributions:

1) We propose a generalized structured diagonal approximation of the Hessian of the objective function.
2) Under some standard assumptions and with the aid of the chosen line search technique, we show the $R-$linear convergence of the algorithm.
3) We apply the proposed algorithm to a robotic motion control model with 3DOF.

We divided the remainder of the article into the following sections: We will state the algorithm's formulation and its steps in section 2. Next, we describe the algorithm's convergence under some conditions in section 3, and finally, we present experimental results of the algorithm and its application in section 4. In this article, $\|\cdot\|$ means a Euclidean norm.

## II. DESIGN AND STATEMENT OF THE PROPOSED ALGORITHM

From the second term of (3), we can observe that computing the residuals' second-order derivative is required. This second term is computationally expensive; thus, approximating the term may be a reasonable idea since it helps to evaluate the Hessian of the objective function.

Suppose at an iteration say, $k$ the second term of equation (3) is as follows:

$$Q(x_{k+1}) = \sum_{i=1}^{m} r_i(x_{k+1})K_i(x_{k+1}), \tag{4}$$

in which $r_i(x_{k+1})$, and $K_i(x_{k+1})$ denote the $i^{th}-$ component of the residual vector $r(x_{k+1})$, and Hessian of $r_i(x_{k+1})$, respectively.

Thus, the goal is to find a diagonal matrix say, $B(x_{k+1})$ that satisfies the following weak secant condition stated as follows:

$$\begin{aligned} s_k^T B(x_{k+1})s_k &\approx s_k^T H(x_{k+1})s_k \\ &= s_k^T J(x_{k+1})^T J(x_{k+1})s_k + s_k^T Q(x_{k+1})s_k > 0, \end{aligned}$$

where $s_k = x_{k+1} - x_k$, $B(x_{k+1})$ denoted by $B_{k+1}$ is defined using the least change secant condition and the term

$s_k^T J(x_{k+1})^T J(x_{k+1}) s_k = \|J(x_{k+1})s_k\|^2 \geq 0$. Therefore, we are now left with approximating the term $s_k^T Q(x_{k+1})s_k$.

Now, post-multiplying (4) by $s_k$ gives

$$Q(x_{k+1})s_k = \sum_{i=1}^{m} r_i(x_{k+1})K_i(x_{k+1})s_k, \tag{5}$$

where for notational simplicity, we represent $Q(x_{k+1}) = Q_{k+1}$, $r_i(x_{k+1}) = r_{k+1}^i$, and $K_i(x_{k+1}) = K_{k+1}^i$, this is essentially approximating the action of the second order term $K_{k+1}^i$ on $s_k$ without explicitly computing the $K_{k+1}^i$.

Suppose that the gradient of the residual $r_{k+1}^i$ at $i^{th}-$ component is denoted by $g_{k+1}^i$. We now, use Taylor's series expansion on $g_{k+1}^i$ to approximate the term $K_{k+1}^i s_k$ as follows:

$$g_k^i \approx g_{k+1}^i - K_{k+1}^i s_k, \quad i = 1, 2, 3, \cdots m,$$

this implies,

$$K_{k+1}^i s_k \approx g_{k+1}^i - g_k^i. \tag{6}$$

Now, we have

$$Q_{k+1}s_k = \sum_{i=1}^{m} r_{k+1}^i K_{k+1}^i s_k. \tag{7}$$

Therefore, plugging-in equation (6) into equation (7) and summing over all $i = 1, 2, 3, \cdots, m$ gives

$$s_k^T Q_{k+1}s_k \approx s_k^T (J_{k+1} - J_k)^T r_{k+1} = r_{k+1}^T (J_{k+1} - J_k)s_k. \tag{8}$$

Hence, we aim to obtain diagonal matrix, $B_{k+1}$ which satisfies the property that

$$s_k^T B_{k+1}s_k \approx s_k^T H(x_{k+1})s_k = s_k^T (J_{k+1}^T J_{k+1})s_k + s_k^T Q_{k+1}s_k.$$

Thus, the requirement is that

$$s_k^T B_{k+1}s_k = s_k^T (J_{k+1}^T J_{k+1})s_k + s_k^T Q_{k+1}s_k. \tag{9}$$

The diagonal approximation, $B_{k+1}$ of the Hessian, $H_{k+1}$ in the above modified weak secant condition is defined as $B_{k+1} = B_k + C_k$, in which $C_k$ is a diagonal correction matrix, where $B_k$ is a diagonal approximation of $H_k$ and both $B_k$ and $B_{k+1}$ are required to positive definite. Next, we state *Lemma* with which we derive the diagonal entries of the correction matrix.

*Lemma 1: Let $C_k$ and $B_k$ be two diagonal matrices containing the elements $c_k^i$ and $b_k^i$ for $i = 1, 2, \cdots, m$ respectively. Then the entries, $c_k^i$ of the solution of the following optimization problem*

$$\min_{C_k} \frac{1}{2}\|C_k\|_{W_k}^2 + tr(B_k + C_k), \tag{10}$$

$$s.t\ s_k^T (B_k + C_k)s_k = \gamma_k, \tag{11}$$

*satisfies*

$$c_k^i = \left[\frac{[s_k^T W_k^2 s_k - s_k^T B_k s_k + \gamma_k]}{\sum_{i=1}^m (s_k^i)^4 (w_k^i)^2}(s_k^i)^2 - 1\right](w_k^i)^2 \qquad i = 1, 2, \cdots, m \tag{12}$$

*where*

$$\gamma_k = s_k^T (J_{k+1}^T J_{k+1})s_k + r_{k+1}^T (J_{k+1} - J_k)s_k, \tag{13}$$

*$\|\cdot\|_W$, is a weighted Frobenius norm and $tr(\cdot)$ is trace of a matrix.*

*Proof:* The optimization problem (10) can be reformulated as

$$\min_{c} \frac{1}{2}\sum_{i=1}^{m} (c_k^i)^2 (w_k^i)^2 + \sum_{i=1}^{m} (b_k^i + c_k^i) \tag{14}$$

$$\text{s.t} \sum_{i=1}^{m} (s_k^i)^2 (b_k^i + c_k^i) = \gamma_k. \tag{15}$$

Since the problem (10) is convex. So, the Lagrangian function of (12) is as follows:

$$L(c_k, \beta_k) = \frac{1}{2}\sum_{i=1}^{m} (c_k^i)^2 (w_k^i)^{-2} + \sum_{i=1}^{m} (b_k^i + c_k^i) + \beta_k \left(\sum_{i=1}^{m} (s_k^i)^2 (b_k^i + c_k^i) - \gamma_k\right),$$

in which, $\beta_k$ is a Lagrangian multiplier. Now, evaluating $\frac{\partial L}{\partial c_k^i}$ and setting $\frac{\partial L}{\partial c_k^i} = 0$ we have

$$\frac{\partial L}{\partial c_k^i} = c_k^i (w_k^i)^{-2} + 1 + \beta_k (s_k^i)^2 = 0 \quad for\ i = 1, 2, \cdots, m,$$

this implies,

$$c_k^i = [-\beta_k (s_k^i)^2 - 1](w_k^i)^2 \quad for\ i = 1, 2, \cdots, m \tag{16}$$

pre-multiplying equation (16) by $(s_k^i)^2$ and calling up the constraint (15) we have

$$\sum_{i=1}^{m} (s_k^i)^2 c_k^i = \sum_{i=1}^{m} (s_k^i)^2 [-\beta_k (s_k^i)^2 - 1](w_k^i)^2 = \gamma_k - \sum_{i=1}^{m} (s_k^i)^2 b_k^i, \quad for\ i = 1, 2, \cdots, m.$$

Therefore, solving for $\beta_k$ from the above expression gives

$$\beta_k = \frac{\left[\sum_{i=1}^m (s_k^i)^2 b_k^i - \gamma_k - \sum_{i=1}^m (s_k^i)^2 (w_k^i)^2\right]}{\sum_{i=1}^m (s_k^i)^4 (w_k^i)^2}, \qquad i = 1, 2, \cdots, m. \tag{17}$$

Thus, plugging equation (17) into equation (16), gives the entries of the correction matrix, $C_k$ as

$$c_k^i = \left[\frac{\left[\sum_{i=1}^m (s_k^i)^2 (w_k^i)^2 - \sum_{i=1}^m (s_k^i)^2 b_k^i + \gamma_k\right]}{\sum_{i=1}^m (s_k^i)^4 (w_k^i)^2}(s_k^i)^2 - 1\right] \times (w_k^i)^2 \text{for } i = 1, 2, \cdots, m. \tag{18}$$

Now, by setting $S_k = diag(s_k)$ and $W_k = diag(w_k)$ and substituting these terms in (18), the diagonal correction matrix, $C_k$ can simply be written as

$$C_k = \left[\frac{(s_k^T W_k^2 s_k - s_k^T B_k s_k + \gamma_k)}{s_k^T (S_k^2 W_k^2)s_k} S_k^2 - I\right] W_k^2. \tag{19}$$

□

*Note: The motivation behind adding the trace operator in equation* (10) *is that we intend to find the correction matrix that clusters the eigenvalues of the updated diagonal matrix, $B_{k+1}$, in such away that its condition number is improved.*

Moreover, in what follows, we look at some possible options of the weighting matrix, $W_k$ in (19). Some of the apparent choices of $W_k$ are as follows:

1) **Choice** Take $W_k = I$, leads to the standard formulation proposed in [24].
2) **Choice** Another alternative choice of $W_k$, motivated from Davidon-Fletcher-Powell (DFP) update can be obtain by setting $W_k = B_k$. This will yield the following correction matrix

$$C_k = \left[ \frac{(s_k^T B_k^2 s_k - s_k^T B_k s_k + \gamma_k)}{s_k^T (S_k^2 B_k^2) s_k} S_k^2 - I \right] B_k^2. \tag{20}$$

It can be observed that by choosing a weighting that varies, the denominator of (20) may become too small as the iteration progresses. To remedy this, as was similarly suggested in [26], we use the above correction matrix in the update if $s_k^T (S_k^2 B_k^2) s_k \geq \nu_1 \|s_k\|^2 tr(S_k^2 B_k^2)$, where $\nu_1$ is some small values in the interval, (0, 1).

Therefore, the search direction say, $d_k$ of the propose method can simply be defined as

$$d_{k+1} = \begin{cases} -B_0^{-1} g_0, & \text{for } k = 0, \\ -B_{k+1}^{-1} g_{k+1}, & \text{for } k = 1, 2, 3, \cdots, \end{cases} \tag{21}$$

where $B_0 = \text{diag}(b_0^i)$, $b_0^i = 1$ for all $i$, and the entries of the diagonal matrix $B_{k+1}$ are given as follows

$$b_{k+1}^i = \nu_2 b_k^i + c_k^i, \quad \text{for } k = 0, 1, 2, \cdots, \tag{22}$$

where $c_k^i$ is given by (18) and $\nu_2 \in (0, 1)$. The parameter $\nu_2$, is introduce into (22) just to aid in showing the convergence result.

We employed a monotone line search couple with backtracking strategy for selecting a suitable step length. The step say, $\alpha$ that satisfies Armijo line search conditions together with backtracking strategy, is computed as follows:

**Algorithm 1** Armijo Line Search With Backtracking

**Input:** Objective function $f_k$, the search direction vector, $d_k$ at the point, $x_k$ and positive real numbers $\varsigma \in (0, 1)$

**Step 1:** Set $\alpha = 1$, if

$$f(x_k + \alpha d_k) \leq f_k + \varsigma \alpha g_k^T d_k \tag{23}$$

then $\alpha_k = \alpha$. Else, set $\alpha = \alpha/2$ and test (23) again.

**Output:** $\alpha_k$

In what follows, we state the steps of our proposed algorithm as follows:

*Remark 1: The above Algorithm 2 is composed of two algorithms depending on the choice of $W$. If $W = I$ for all $k$, then in evaluating the entries, $c_k^i$ for $i = 1, 2, \cdots, m$ of the correction matrix, $C_k$, $w_k^i = 1$ for all $k$, however, if $W = B$ for all $k$, then the entries $c_k^i$ for $i = 1, 2, \cdots, m$ are computed using* (20).

**Algorithm 2** **G**eneralized **S**tructured **D**iagonal-Based **A**lgorithm (GSDA)

**Input:** Choose an initial approximation $x_0 \in \mathbb{R}^n$, $B_0 = I$, $W_0 = I$, $\nu_2 \in (0, 1)$, $\varsigma \in (0, 1)$. Set $k = 0$, and $Tol > 0$.

Compute $r_k$, $f_k$ and $g_k$; and then compute $d_k = -B_k^{-1} g_k$.

**Step 1:** If $\|g_k\| \leq Tol$, stop. Else, go to **Next step**

**Step 2:** Compute $\alpha_k$ using Algorithm 1.

**Step 3:** Evaluate the next iterate using

$$x_{k+1} = x_k + \alpha_k d_k. \tag{24}$$

**Step 4:** Evaluate the update of the entries, $b_{k+1}^i$ of the diagonal matrix, $B_{k+1}$ as follows:

$$b_{k+1}^i = \nu_2 b_k^i + c_k^i,$$

where $c_k^i$ is defined in (18).

**Step 5:** Update as follows:

$B_{k+1} = diag(b_{k+1}^i)$.
$d_{k+1} = -B_{k+1}^{-1} g_{k+1}$.
$W_{k+1} = diag(w_k^i)$ where $w_k \in W_k$

**Step 6:** Set $k := k + 1$ and go to **Step 1.**

## III. CONVERGENCE ANALYSIS

For the convergence analysis of the proposed algorithm, we first present the following useful assumption:

*Assumption 1: The objective function $f$ is twice continuously differentiable on a set, $\chi = \{x \in \mathbb{R}^n | f(x) \leq f(x_o)\}$.*

*Assumption 2: There exist some positive constants $N_1$ and $N_2$ where $N_1 \leq N_2$ such that*

$$N_1 \|u\|^2 \leq u^T \nabla^2 f(x) u \leq N_2 \|u\|^2, \tag{25}$$

*for all $u \in \mathbb{R}^n$ and $x \in \chi$, holds.*

Next, we state an underline assumption on the Jacobian matrix and residual as follows:

*Assumption 3: We also assume that the Jacobian, denoted by $J(x)$ and the residual $r(x)$ are Lipschitz continuous in some neighborhood $N$ of $\chi$ with Lipschitz constants $l_1 > 0$ and $l_2 > 0$ i.e $\|J(x) - J(y)\| \leq l_1 \|x - y\|$, and $\|r(x) - r(y)\| \leq l_2 \|x - y\|$, $\forall x, y \in \chi$.*

It can be deduced from the above Assumption 3 that there exist some positive constants $l_3, c_1, c_2, c_3$ such that $\forall x, y \in \chi$, we obtain

$$\|g(x) - g(y)\| \leq l_3 \|x - y\|,$$
$$\|J(x)\| < c_1, \quad \|r(x)\| < c_2, \quad \|g(x)\| \leq c_3.$$

*Lemma 2: Suppose Assumptions 1, 3 and 2 hold, then there exists some positive constants $N_1$ and $\bar{N}$ such that, $\forall k > 0$,*

$$N_1 \|s_k\|^2 \leq |\gamma_k| \leq \bar{N} \|s_k\|^2. \tag{26}$$

*Proof:* Recall, that $\gamma_k$ is defined in (13) as follows

$$\begin{aligned}|\gamma_k| &= |s_k^T J_{k+1}^T J_k s_k + s_k^T (J_{k+1} - J_k)^T r_{k+1}| \\ &\le |s_k^T J_{k+1}^T J_k s_k| + |s_k^T (J_{k+1} - J_k)^T r_{k+1}| \\ &\le \|J_{k+1}\|^2 \|s_k\|^2 + \|s_k\| \|J_{k+1} - J_k\| \|r_{k+1}\| \\ &\le c_1^2 \|s_k\|^2 + l_1 \|s_k\|^2 \|r_{k+1}\| \\ &\le c_1^2 \|s_k\|^2 + c_1 l_2 \|s_k\|^2 \\ &= (c_1^2 + l_1 c_2) \|s_k\|^2 \\ &= L \|s_k\|^2,\end{aligned}$$

where $L := c_1^2 + l_1 c_2$.

Now, from (25) and the above inequality, we have

$$\begin{aligned}N_1 \|s_k\|^2 &\le s_k^T y_k \\ &\le s_k^T z_k \\ &= s_k^T y_k + |\gamma_k| \\ &\le (N_2 + L) \|s_k\|^2,\end{aligned}$$

where $y_k = g_{k-1} - g_k$ and $z_k$ is a structured vector. Hence, by setting $\bar{N} = N_2 + L$, the inequality (26) holds. □

*Lemma 3: Suppose that the step-size $\alpha_k$ is established by Algorithm 1, and assume that Assumption 2 is satisfied. Then either $\alpha_k = 1$ or there exist some positive constants $p_1$ and $p_2$ such that*:

$$p_1 \frac{s_k^T B_k s_k}{\|s_k\|^2} \le \alpha_k \le p_2 \frac{s_k^T B_k s_k}{\|s_k\|^2}. \tag{27}$$

*Proof:* Suppose 23 is satisfied by $\alpha_k = 1$, then the first segment of the proof is achieved. Let $\alpha_k < 1$, which simply mean that the relation (23) failed, for a step-size $\alpha_k < \alpha \le 2\alpha_k$. This implies

$$f(x_k + \alpha d_k) - f(x_k) > \varsigma \alpha g_k^T d_k.$$

Then by using mean-value theorem, we can have

$$f(x_k + \alpha d_k) - f(x_k) = g(x_k + \delta_1 \alpha d_k)^T (\alpha d_k) > \varsigma \alpha g_k^T d_k,$$

in which $\delta_1 \in (0, 1)$. Thus, this leads to

$$g(x_k + \delta_1 \alpha d_k)^T (\alpha d_k) - \alpha g_k^T d_k > \varsigma \alpha g_k^T d_k - \alpha g_k^T d_k. \tag{28}$$

This implies, that

$$\alpha [g(x_k + \delta_1 \alpha d_k) - g_k]^T d_k > \alpha (\varsigma - 1) g_k^T d_k.$$

Now, using (28) and Assumption 2, yield

$$\alpha (\varsigma - 1) g_k^T d_k < \alpha [g(x_k + \delta_1 \alpha d_k) - g_k]^T d_k \le N_2 \alpha \|d_k\|^2.$$

Then,

$$\begin{aligned}\alpha_k &\ge \frac{1}{2} \alpha > \frac{(1-\varsigma)(-g_k^T d_k)}{2N_2 \|d_k\|^2} \\ &= \frac{(1-\varsigma)}{2N_2} \frac{s_k^T B_k s_k}{\|s_k\|^2}.\end{aligned}$$

Thus, the lower bound on $\alpha_k$ is established, when we set $p_1 = \frac{(1-\varsigma)}{2N_2}$.

The condition in (23) gives the upper bound on $\alpha_k$, by Taylor's theorem, we have

$$f(x_{k+1}) - f(x_k) = g_k^T s_k + \frac{1}{2} s_k^T H(\psi) s_k,$$

for some $\psi$ that lie in the line segment joining $x_{k+1}$ and $x_k$. Therefore,

$$\begin{aligned}g_k^T s_k + \frac{1}{2} s_k^T H(\psi) s_k &= f(x_{k+1}) - f(x_k) \le \varsigma g_k^T s_k, \\ 2 g_k^T s_k + s_k^T H(\psi) s_k &\le 2 \varsigma g_k^T s_k, \\ \text{this implies } s_k^T H(\psi) s_k &\le 2(\varsigma - 1) g_k^T s_k = 2(1-\varsigma)(-g_k^T s_k), \\ \text{this implies } s_k^T H(\psi) s_k &\le 2(\varsigma - 1) \frac{s_k^T B_k s_k}{\alpha_k}, \\ \text{this implies } \alpha_k &\le 2(1-\varsigma) \frac{s_k^T B_k s_k}{s_k^T H(\psi) s_k} \\ \alpha_k &\le 2(1-\varsigma) \frac{\frac{s_k^T B_k s_k}{\|s_k\|^2}}{\frac{s_k^T H(\psi) s_k}{\|s_k\|^2}} \\ &= \frac{2(1-\varsigma)}{N_1} \frac{s_k^T B_k s_k}{\|s_k\|^2}.\end{aligned}$$

The required inequality in (27) is obtain by setting $p_2 = \frac{2(1-\varsigma)}{N_1}$. □

*Lemma 4: Suppose the sequence $\{B_k\}$ is generated by Algorithm 2 and let Assumptions 1 and 2 hold, if the entries of diagonal matrix $B_0$ are bounded. Then, there exist some positive constants $\Delta_1$, $\Delta_2$ and $\Delta_3$ such that*

$$tr(B_{k+1}) \le \Delta_1^{k+1} \quad \textit{for appropriately large } k.$$

*Furthermore, if $\nu_2 < 1$, then*

$$tr(B_k) < \Delta_2 \quad \forall k.$$

*Moreover, $B_{k+1}^{(i)} \ge \Delta_3$ for $i = 1, 2, \cdots, m$.*

*Proof:* Now, suppose we define $\sigma_1 = \|W_k\| = \max\{|w_k^i|\}$, for $i = 1, 2, \cdots, m$, and also $\|s_k\|^2 = s_k^T s_k = \sum_{i=1}^{m} (s_k^i)^2 \le \sum_{i=1}^{m} (s_k^{\max})^2 = m\sigma_2^2$, where $s_k^{\max}$, is a component of $s_k$ with largest term and $\beta_1 \|s_k\|^2 \le s_k^T B_k s_k \le \beta_2 \|s_k\|^2$.

Consider $\beta_3 = \max\{|\beta_1|, |\beta_2|\}$

Now, from the diagonal matrix form of (22) we have,

$$\begin{aligned}&tr(B_{k+1}) \\ &\le \nu_2 tr(B_k) \\ &\quad + \left[ \frac{(s_k^T W_k^2 s_k + s_k^T B_k s_k + \gamma_k)}{s_k^T (S_k^2 W_k^2) s_k} tr(S_k^2 W_k^2) - tr(I) tr(W_k^2) \right] \\ &\le \nu_2 tr(B_k) + \left[ \frac{|s_k^T W_k^2 s_k + s_k^T B_k s_k + \gamma_k|}{s_k^T (S_k^2 W_k^2) s_k} tr(S_k^2 W_k^2) \right] \\ &\le \nu_2 tr(B_k) + \left[ \frac{|s_k^T W_k^2 s_k| + |s_k^T B_k s_k| + |\gamma_k|}{s_k^T (S_k^2 W_k^2) s_k} tr(S_k^2 W_k^2) \right].\end{aligned}$$

Now, using Lemma (2), we have

$$\begin{aligned}
tr(B_{k+1}) & \\
&\leq \nu_2 tr(B_k) + \left[\frac{|s_k^T W_k^2 s_k| + |s_k^T B_k s_k| + |\gamma_k|}{s_k^T(S_k^2 W_k^2)s_k} tr(S_k^2 W_k^2)\right] \\
&\leq \nu_2 tr(B_k) + \left[\frac{\|s_k\|^2\|W_k^2\| + |s_k^T B_k s_k| + |\gamma_k|}{s_k^T(S_k^2 W_k^2)s_k} tr(S_k^2 W_k^2)\right] \\
&\leq \nu_2 tr(B_k) + \left[\frac{\sigma_1^2\|s_k\|^2 + \beta_3\|s_k\|^2 + |\gamma_k|}{\nu_1\|s_k\|^2 tr(S_k^2 W_k^2)} tr(S_k^2 W_k^2)\right] \\
&\leq \nu_2 tr(B_k) + \left[\frac{(\sigma_1^2 + \beta_3 + \bar{N})\|s_k\|^2}{\nu_1\|s_k\|^2 tr(S_k^2 W_k^2)} tr(S_k^2 W_k^2)\right] \\
&\leq \nu_2 tr(B_k) + \left[\frac{\sigma_1^2 + \beta_3 + \bar{N}}{\nu_1}\right] \\
&\vdots \\
&\leq \nu_2^{k+1} tr(B_0) + \left(1 + \sum_{j=1}^{k} \nu_2^j\right)\left[\frac{\sigma_1^2 + \beta_3 + \bar{N}}{\nu_1}\right] \\
&\leq tr(B_0) + (k+1)\left[\frac{\sigma_1^2 + \beta_3 + \bar{N}}{\nu_1}\right] \\
&\leq n + (k+1)\left[\frac{\sigma_1^2 + \beta_3 + \bar{N}}{\nu_1}\right] \\
&\leq \Delta_1^{k+1},
\end{aligned}$$

where $\Delta_1 = \max\{2, n + \left[\frac{\sigma_1^2+\beta_3+\bar{N}}{\nu_1}\right]\}$.

Furthermore, if $\nu_2 < 1$, we have

$$\begin{aligned}
tr(B_{k+1}) &< tr(B_0) + \left(1 + \lim_{k\to\infty}\sum_{j=1}^{k} \nu_2^j\right)\left[\frac{\sigma_1^2 + \beta_3 + \bar{N}}{\nu_1}\right] \\
&\leq n + \left(\frac{1}{1-\nu_2}\right)\left[\frac{\sigma_1^2 + \beta_3 + \bar{N}}{\nu_1}\right] \\
&= \Delta_2.
\end{aligned}$$

On the other-hand, we have

$$B_{k+1}^{(i)} \geq \nu_2 B_k^{(i)} \geq \Delta_3, \quad \forall i.$$

Hence, the diagonal matrix $B_k^{(i)}$ is bounded both above and below by some positive constants for $i = 1, 2, \cdots m$. □

We now state and prove the theorem that shows the convergence rate of the proposed algorithm.

*Theorem 1: Suppose the Algorithm 2 generates sequence* $\{x_{k+1}\}$ *using* (24) *where the search direction,* $d_k = -B_k^{-1} g_k$, *whose elements of* $B_k$ *are evaluated using* (22) *and let Assumptions 1 and 2 hold. Then for any positive definite matrix* $B_0$, *which possesses bounded diagonal entries, the Algorithm generate sequence of iterates which converges to the minimizer say,* $x^*$ *and*

$$\sum_{k=0}^{\infty}\|x_k - x^*\| < \infty.$$

*Moreso, there is a positive constant,* $\nu_3$ *in* $[0, 1)$ *such that*

$$f(x_{k+1}) - f(x_k) \leq \nu_3^{k+1}(f(x_0) - f(x^*)),$$

*where* $x_0$ *is a starting point of* $f$.

*Proof:* Suppose we define $\theta_k$ to be the angle between the search direction, $d_k$ and negative gradient, $-g_k$ stated as

$$\begin{aligned}
\cos\theta_k &= \frac{g_k^T B_k^{-1} g_k}{\|g_k\|\|B_k^{-1} g_k\|} \\
&= \frac{s_k^T B_k s_k/\alpha_k^2}{\|B_k s_k\|\|s_k\|/\alpha_k^2} \\
&= \frac{s_k^T B_k s_k}{\|s_k\|\|B_k s_k\|}.
\end{aligned}$$

Therefore, using the line-search condition in Algorithm 1, the lower boundedness of $\alpha_k$ in Lemma 3 and the assumptions on $f$ we have.

$$\begin{aligned}
f(x_{k+1}) - f(x^*) &\leq f(x_k) - f(x^*) - \varsigma\alpha_k g_k^T B_k^{-1} g_k, \\
&\leq f(x_k) - f(x^*) - \varsigma p_2 \frac{(s_k^T B_k s_k)^2}{\|s_k\|^2} \frac{\|g_k\|^2}{\|g_k\|^2}, \\
&= f(x_k) - f(x^*) - \varsigma p_2 \frac{(s_k^T B_k s_k)^2}{\|s_k\|^2\|B_k s_k\|^2}\|g_k\|^2, \\
&= f(x_k) - f(x^*) - \varsigma p_2 \cos^2\theta_k\|g_k\|^2,
\end{aligned}$$

where, $p_2 = \frac{(1-\varsigma)}{2N_1}$. Now, from Assumption 2 i.e the boundedness of $\nabla^2 f$, we have

$$N_1\|x_k - x^*\|^2 \leq (x_k - x^*)^T(g(x_k) - g(x^*)),$$

where $g(x^*) = 0$ and hence, applying the Cauchy-Schwartz, on the above expression, we have

$$N_1\|x_k - x^*\|^2 \leq \|x_k - x^*\|\|g_k\|. \tag{29}$$

Hence,

$$f(x_k) - f(x^*) \leq (x_k - x^*)^T(g(x_k) - g(x^*)) \leq \|x_k - x^*\|\|g_k\|. \tag{30}$$

Therefore, using (29) and (30), we have

$$\begin{aligned}
f(x_k) - f(x^*) &\leq \frac{1}{N_1}\|g_k\|^2 \\
&\Longrightarrow N_1[f(x_k) - f(x^*)] \leq \|g_k\|^2, \\
f(x_{k+1}) - f(x^*) &\leq f(x_k) - f(x^*) - \varsigma p_2\cos^2\theta_k\|g_k\|^2, \\
&\leq [1 - N_1\varsigma p_2\cos^2\theta_k](f(x_k) - f(x^*))
\end{aligned} \tag{31}$$

Now, using the upper bound of $\alpha_k$ and the inequality which states that $\frac{\|B_k s_k\|}{\|s_k\|} \leq tr(B_k)$.

Therefore, $\frac{\alpha_k}{\cos\theta_k} \leq \frac{p_2\|B_k s_k\|}{\|s_k\|} \leq p_2 tr(B_k) = p_2\Delta_2 = \Delta_4$. Since $B_k$ is bounded as was shown in Lemma 3, therefore, we have

$$\sum_{i=0}^{k}\frac{\alpha_i}{\cos\theta_i} \leq (k+1)\Delta_4. \tag{32}$$

Now, applying the geometric/arithmetic mean inequality (i.e $det(B_{k+1})^{\frac{1}{m}} \leq \frac{tr(B_{k+1})}{m}$) to (32) gives

$$\prod_{i=0}^{k} \frac{\alpha_i}{\cos\theta_i} \leq \Delta_4^{k+1}. \tag{33}$$

Similarly, using the upper bound for $B_k^{(i)}$ obtained from Lemma 4 we have

$$\frac{\|s_k\|^2}{s_k^T B_k s_k} = \frac{\|d_k\|^2}{g_k^T d_k} \leq \Delta_2.$$

Hence, using the above relation, together with bounds of $\alpha_k$, we have

$$\alpha_k \geq \frac{p_1 s_k^T B_k s_k}{\|s_k\|^2} \geq \frac{p_1}{\Delta_2}. \tag{34}$$

Thus, using the relations (33) and (34), we have

$$\prod_{i=0}^{k} \cos\theta_i \geq \frac{\prod_{i=0}^{k}\alpha_i}{\Delta_3^{k+1}} \geq \left(\frac{p_1}{\Delta_2\Delta_3}\right)^{k+1}. \tag{35}$$

Moreover, when induction is apply to (31), we have

$$f(x_{k+1}) - f(x^*) \leq \prod_{i=0}^{k}[1 - N_1\varsigma p_2\cos^2\theta_i](f(x_0) - f(x^*)) \tag{36}$$

Now, re-applying the geometric/arithmetic mean and using (35), we get

$$\begin{aligned} f(x_{k+1}) - f(x^*) &\leq \left(\frac{1}{k+1}\sum_{i=0}^{k}(1 - N_1\varsigma p_2\cos^2\theta_i)\right)^{k+1} (f(x_0) - f(x^*)) \\ &\leq \left(1 - N_1\varsigma p_2 (\prod_{i=0}^{k}\cos^2\theta_i)^{\frac{1}{k+1}}\right)^{k+1} (f(x_0) - f(x^*)) \\ &\leq \nu_3^{k+1}(f(x_0) - f(x^*)), \end{aligned} \tag{37}$$

where $\nu_3 = 1 - N_1\varsigma p_2\left(\frac{p_1}{\Delta_2\Delta_3}\right) < 1$.

Furthermore, with the aid of Assumption 2, we can easily achieve as follows:

$$\begin{aligned} \frac{1}{2}m\|x_{k+1} - x_k\|^2 &\leq f(x_{k+1}) - f(x^*) \\ &\leq \nu_3^{k+1}(f(x_0) - f(x^*)) \\ &\leq \frac{1}{m}\|g_{k+1}\|^2. \end{aligned} \tag{38}$$

The above relation (38), together with (37) gives

$$\|x_{k+1} - x_k\|^2 \leq \frac{2}{m}\nu_3^{k+1}(f(x_0) - f(x^*)).$$

Hence,

$$\begin{aligned} \sum_{k=0}^{\infty}\|x_{k+1} - x_k\| &\leq \left(\frac{2}{m}\right)^{\frac{1}{2}} \sum_{k=0}^{\infty}(f(x_{k+1}) - f(x^*))^{\frac{1}{2}}, \\ &\leq \left(\frac{2}{m}(f(x_0) - f(x^*))\right)^{\frac{1}{2}} \sum_{k=0}^{\infty}(\nu_3^{\frac{1}{2}})^{k+1} < \infty. \end{aligned}$$

Therefore, the sequence $\{x_k\}$ is convergent. □

## IV. NUMERICAL EXPERIMENTS

This section explores the proposed algorithm's numerical performance compared to other recent structured algorithms. We segmented the experiment into two components. The first part is composed of/discuss testing the algorithm on some benchmark test problems. On the other hand, the second segment comprises applying the proposed algorithm to solve some data fitting problems in the literature. We conducted these experiments on a MATLAB R2019b programming packet installed on a PC with a processor speed of 1.60 GHz, intel CORE i5-8265U, and 8 GB of RAM.

### A. EXPERIMENTATION ON SOME BENCHMARK TEST PROBLEMS

This subsection presented some numerical results on solving a set of benchmark test problems. These results verify the numerical efficiency of the proposed algorithm in comparison to ASDA1 and ASDA2 (which are essentially the proposed algorithm when $W = I$) algorithms developed in [24]. The extracted problems are from various sources in the literature; we cited each problem's reference and their respective standard initial starting point (see Table 1).

The set of problems considered in this experiment comprises twenty(20) large-scales while the remaining three (3) are small-scales. Each of these large-scale problems had varying dimensions. This dimensions are 3000, 6000, 9000, 12000, 15000. The parameters used in implementing the proposed GSDA algorithm are as follows:

- Algorithm GSDA: $\epsilon = 10^{-2}$, $\varsigma = 10^{-4}$, $\varepsilon = 10^{-3}$, $Tol = 10^{-4}$

On the other hand, we took the parameters of ASDA1 and ASDA2 from [24]. Furthermore, unlike ASDA1 and ASDA2 algorithms, where a monotone line search strategy is adopted, we used a simple Armijo line search technique based on Algorithm 1. An approximate solution is achieved when the stopping criterion $\|g_k\| \leq 10^{-6}$ is satisfied. However, a failure by an algorithm reported as $F$ occurs when either the number of iteration surpasses 1000 and the stopping criterion mentioned above has not been satisfied. The standard metrics of comparison used are the number of iterations, number of functions evaluations, number of matrix-vector products, and computing time. These are represented by *#niter*, *#nfval*, *#nmvp* and *#ncpu* respectively. The results of the numerical experiments are tabulated and made available in this link: https://github.com/MAHMOUDPD/Experimental_Results_of_GSDA_Alg

**TABLE 1. List of test problems with references and their respective starting points.**

| Problems | Function name | Starting point |
|---|---|---|
| Large scale | | |
| P1 | Penalty function I [28] | $(1/3, 1/3, \cdots, 1/3)^T$ |
| P2 | Trigonometric function [29] | $(1/n, \cdots, 1/n)^T$ |
| P3 | Discrete boundary value [29] | $(\frac{1}{n+1}(\frac{1}{n+1}-1), \cdots, \frac{1}{n+1}(\frac{n}{n+1}-1))^T$ |
| P4 | Linear function full rank [29] | $(1, 1, \cdots, 1)^T$ |
| P5 | Problem 202 [30] | $(2, 2, \cdots, 2)^T$ |
| P6 | Problem 206 [30] | $(1/n, \cdots, 1/n)^T$ |
| P7 | Problem 212 [30] | $(0.5, \cdots, 0.5)^T$ |
| P8 | Strictly convex function I [31] | $(1/n, 2/n, \cdots, 1)^T$ |
| P9 | Sine function 2 [32] | $(1, 1, \cdots, 1)^T$ |
| P10 | Exponential function I [28] | $\left(\frac{n}{n-1}, \cdots, \frac{n}{n-1}\right)^T$ |
| P11 | Exponential function II [28] | $(1/n^2, 1/n^2, \cdots, 1/n^2)^T$ |
| P12 | Logarithmic function [28] | $(1, 1, \cdots, 1)^T$ |
| P13 | Trigonometric Exponential System [28] | $(0, 0, \cdots, 0)^T$ |
| P14 | Extended Powell singular [28] | $(1.5E-4, \cdots, 1.5E-4)^T$ |
| P15 | Broyden tridiagonal function [29] | $(-1, -1, \cdots, -1)^T$ |
| P16 | Extended Himmelblau function [33] | $(1, 1/n, 1, 1/n, \cdots, 1, 1/n)^T$ |
| P17 | Function 27 [28] | $(100, 1/n^2, 1/n^2, \cdots, 1/n^2)^T$ |
| P18 | Trigonometric logarithmic function [23] | $(1, 1, \cdots, 1)^T$ |
| P19 | Zero Jacobian function [28] | $\left(\frac{(100(n-100)}{n}, \cdots, \frac{(n-500)(n-1000)}{(60n)^2}\right)^T$ |
| P20 | Exponential function [28] | $(1, 1, \cdots, 1)^T$ |
| P21 | Brown almost linear function [34] | $(1/n, 1/n, \cdots, 1/n)^T$ |
| Small scale | | |
| P22 | Jennrich and Sampson [29] | $(0.2, 0.2)^T$ |
| P23 | Rank deficient Jacobian [35] | $(-1, 1)^T$ |
| P24 | Beale function [34] | $(1, 1)^T$ |

orithm. It can be observed from the results that our proposed algorithm GSDA (with $W = B$) solved all the test problems successfully. The ASDHA1 algorithm subsequently follows this. However, the ASDHA1 and ASDHA2 algorithms recorded some failure cases in problems named P2, P15, and P20. Moreover, for a concrete visual representation of the result, each metric considered for all the problems is summarized using the well-known performance profile of Dolan and Moré [27]. That is, for each algorithm, we plot a fraction, say, $\rho$ of problems for which the algorithm performed well within a factor, say $\tau$. One can easily see from the Figs. 1-4, that the performance of GSDA is superior to all of its competitors. Since the curves formed by the proposed GSDA topped all the algorithms thus, these results indicate that the GSDA algorithm could provide a better alternative for solving NLS problems. Thus, this further accentuates the efficiency of the GSDA algorithm.

*Remark 2: To mitigate the possible generation of non-positive and singular updated diagonal matrix, $B_{k+1}$, we obviously require that the entries, $b^i_{k+1} > 0$ for all $i = 1, 2, \cdots, m$. In practical implementation, the direction, $d^i_{k+1} = -g^i_{k+1}/b^i_{k+1}$ if $b^i_{k+1} \geq \epsilon^*$, for every $i$, else we set $d^i_{k+1} = -g^i_{k+1}$, where $\epsilon^*$ is a positive parameter.*

### B. APPLICATION IN 3DOF MOTION CONTROL OF ROBOTIC MANIPULATOR

In this segment, we apply the proposed GSDA algorithm to solve a real-robotic model with three degrees of freedom (3DOF) that was describe in [36]. We describe the three joint kinematic model in a planar, and the discrete kinematic model equation with 3DOF can be represented using the following equations

$$r(\theta) = \begin{bmatrix} l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) \\ l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) \end{bmatrix}, \quad (39)$$

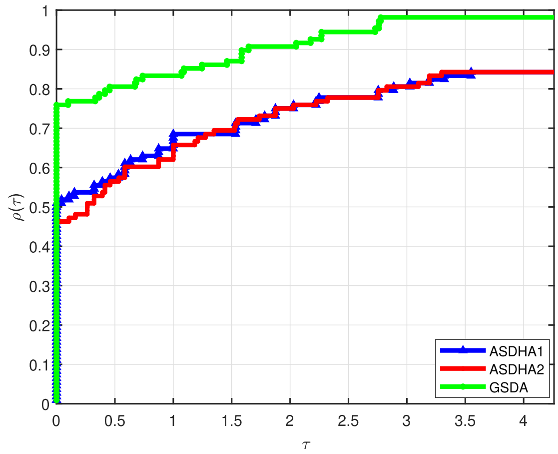


**FIGURE 1. Performance profile based on number of iteration.**

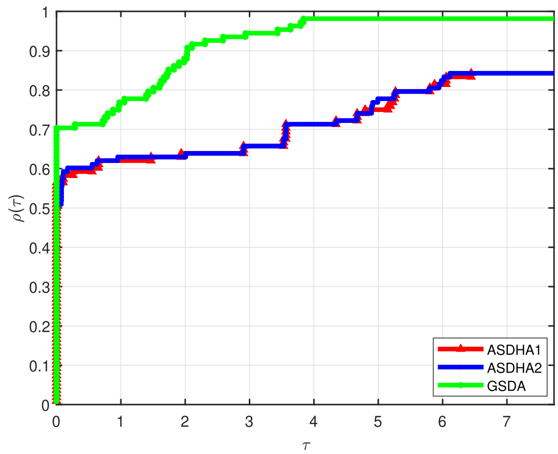


**FIGURE 2. Performance profile based on function evaluations.**

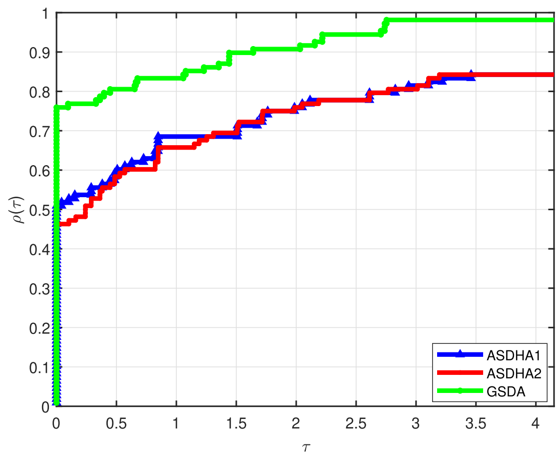


**FIGURE 3. Performance profile based on number of matrix-vector product.**

where $r(\cdot)$ is kinematic mapping function which relate the position and orientation of a robot's end-effector or any part of the robot to an active joint displacements, $\theta \in \mathbb{R}^3$, $l_i$ (for $i = 1, 2, 3$) denotes the length of each link, and in a context

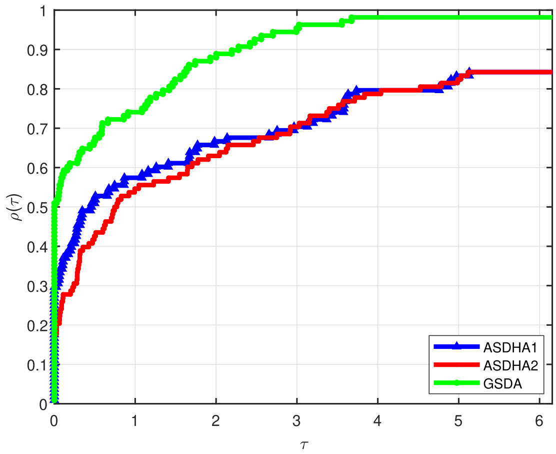


**FIGURE 4.** Performance profile based on CPU-TIME.

of robotic motion control, $r(\theta)$ is an end effector position vector. Suppose, $\delta_{t_k} \in \mathbb{R}^2$ denotes the desired path vector at any given time say, $t_k$. We formulated the following least-squares problem which is solved at every time interval say, $t_k \in [0, t_f]$. The problem is stated as follows:

$$\min_{\theta \in \mathbb{R}^3} \frac{1}{2}\|r(\theta) - \delta_{t_k}\|^2, \tag{40}$$

where $\delta_{t_k}$ as reported in [37] is the desired path at $t_k$ of a Lissajous curve express as

$$\delta_{t_k} = \begin{bmatrix} 1.5 + 0.4\sin(\frac{\pi t_k}{5}) \\ \frac{\sqrt{3}}{2} + 0.4\sin(\frac{\pi t_k}{5} + \frac{\pi}{3}) \end{bmatrix}. \tag{41}$$

It can be observed that the above equation (40) resembles the structure of (1). Thus the GSDA Algorithm can be used to solve it.

**Algorithm 3** GSDA for Solving (40)

**Input:** Initial time duration, $t_0$, Initial joint angle, $\theta_{t_0}$, maximum time duration, $t_{max}$, sampling period, $g$, and maximum iteration, $K_{max}$.
**for** $k = 1 : K_{max}$ **do**
  $t_k = kg$;
  Evaluate $\delta_{t_k}$ using (41).
  Compute $\theta_{t_k}$ using $GDSA(\theta_{t_0}, \delta_{t_k})$ stated in 2.
  Set $\theta_{new} = [\theta_{t_0}; \theta_{t_k}]$
**Output:** $\theta_{new}$

Now, to solve the model and subsequently simulate the results, we initialize the joint at time instant, $t = 0$ to be $\theta_{t_0} = [0, \frac{\pi}{3}, \frac{\pi}{2}]$, the link length as $l_i = 1$ (for $i = 1, 2, 3$) and the maximum duration it takes as, $t_{max} = 10s$, in the above Algorithm 3.

Finally, we can observe from the figures that portray the results obtained from solving (40) using the proposed algorithm. These results are plotted in Figs. 5- 6. It can be seen that from 6, the task of synthesizing the robot trajectories is successfully achieved, and the error rate of the residuals is about $10^{-6}$ which can be observed from Figs. 7 and 8.

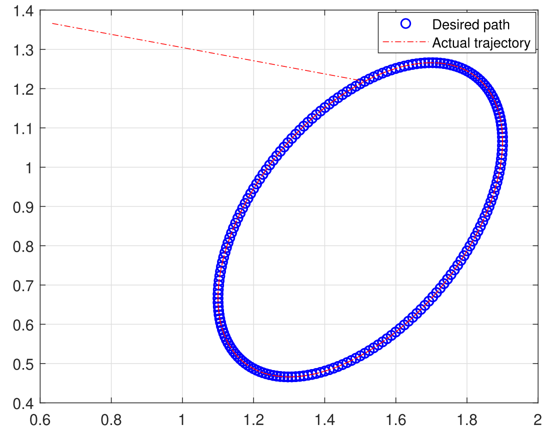


**FIGURE 5.** End effector trajectory and desired path.

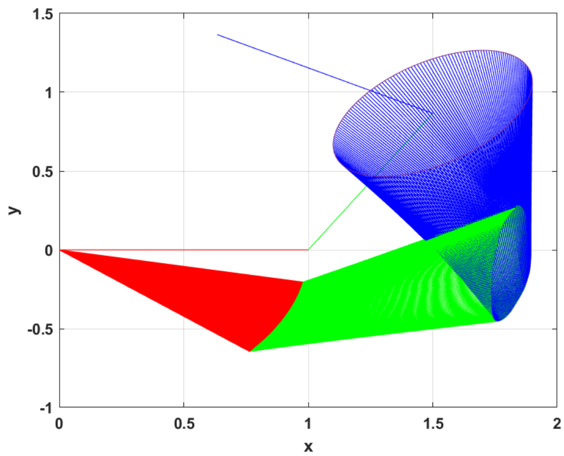


**FIGURE 6.** Synthesized robot trajectories.

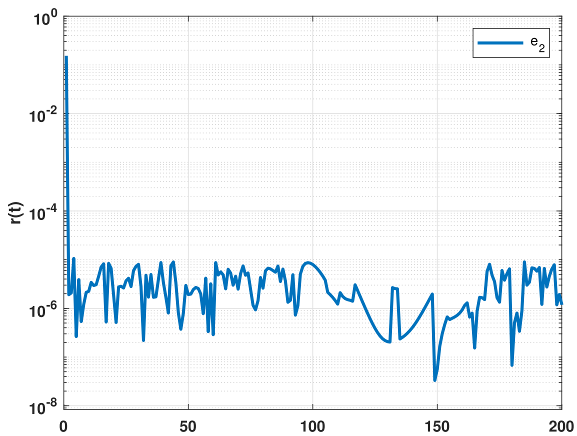


**FIGURE 7.** Tracking residual error on the horizontal x-axis.

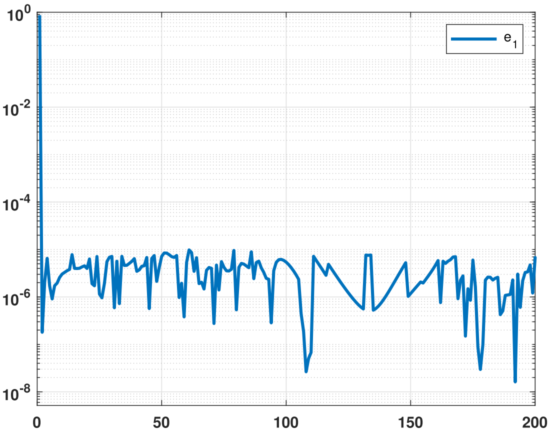


**FIGURE 8.** Tracking residual error on the horizontal y-axis.

## V. CONCLUSION

We have proposed an algorithm for computing a minimizer of nonlinear least-squares problems. The developed algorithm is essentially based on a standard quasi-Newton class of algorithms; we called the algorithm 'generalized structured based diagonal algorithm' (GSDA). It was derived based upon a structured weak secant, the least-change secant updating scheme coupled with the trace of the correction matrix of the updated matrix. The least-change secant is under a weighted Frobenius norm. Thus, the algorithm is matrix-free and straightforward; this simplicity, of course, yields its low computational cost in each iteration. Furthermore, it should be noted; this algorithm is a generalization of the algorithms proposed in [24]. We have also presented the convergence result of the proposed algorithm. In addition, we have shown that the algorithm with monotone(Armijo-line search) is $R-$linearly convergent; this fills the gap that existed in [24] for a convex class of NLS problems. Moreover, the proposal was numerically shown to be efficient and comparatively better than those proposed in [24] when the associated weighted matrix, $W$, is taken as the previous diagonal update $B_k$. However, if the weighted matrix is an identity, $I$, the proposed structured formulation of the diagonal update becomes the one presented in [24]. Finally, we have shown that the algorithm can be applied successfully to robotic planar motion control manipulators with 3DOF; this underscores the applicability of the proposed algorithm. Our GSDA MATLAB codes are available on the first Author's GitHub page through this link: https://github.com/MAHMOUDPD/GSDA_for_Robotic_Arm

## REFERENCES

[1] J.-F. Bao, C. Li, W.-P. Shen, J.-C. Yao, and S.-M. Guu, "Approximate Gauss–Newton methods for solving underdetermined nonlinear least squares problems," *Appl. Numer. Math.*, vol. 111, pp. 92–110, Jan. 2017.

[2] R. J. Renka, "Nonlinear least squares and sobolev gradients," *Appl. Numer. Math.*, vol. 65, pp. 91–104, Mar. 2013.

[3] G. Golub and V. Pereyra, "Separable nonlinear least squares: The variable projection method and its applications," *Inverse Problems*, vol. 19, no. 2, pp. R1–R26, Apr. 2003.

[4] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale -Regularized least squares," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 606–617, Dec. 2007.

[5] M. M. Yahaya, P. Kumam, A. M. Awwal, and S. Aji, "Alternative structured spectral gradient algorithms for solving nonlinear least-squares problems," *Heliyon*, vol. 7, no. 7, Jul. 2021, Art. no. e07499.

[6] J. Li, F. Ding, and G. Yang, "Maximum likelihood least squares identification method for input nonlinear finite impulse response moving average systems," *Math. Comput. Model.*, vol. 55, nos. 3–4, pp. 442–450, Feb. 2012.

[7] A. Cornelio, "Regularized nonlinear least squares methods for hit position reconstruction in small gamma cameras," *Appl. Math. Comput.*, vol. 217, no. 12, pp. 5589–5595, 2011.

[8] C. L. C. Diana, T. Barz, S. Körkel, and G. Wozny, "Nonlinear ill-posed problem analysis in model-based parameter estimation and experimental design," *Comput. Chem. Eng.*, vol. 77, pp. 24–42, Jun. 2015.

[9] L. M. Tang, "A regularization homotopy iterative method for ill-posed nonlinear least squares problem and its application," *Appl. Mech. Mater.*, vols. 90–93, pp. 3268–3273, Sep. 2011.

[10] D. Jukiá and R. Scitovski, "Least squares fitting Gaussian type curve," *Appl. Math. Comput.*, vol. 167, no. 1, pp. 286–298, Aug. 2005.

[11] J. C. Ferreira and M. C. Baquião, "A least square point of view to reproducing kernel methods to solve functional equations," *Appl. Math. Comput.*, vol. 357, pp. 206–221, Sep. 2019.

[12] H. Ren and I. K. Argyros, "Local convergence of a secant type method for solving least squares problems," *Appl. Math. Comput.*, vol. 217, no. 8, pp. 3816–3824, Dec. 2010.

[13] A. B. Abubakar, P. Kumam, M. Malik, and A. H. Ibrahim, "A hybrid conjugate gradient based approach for solving unconstrained optimization and motion control problems," *Math. Comput. Simul.*, Jun. 2021, doi: 10.1016/j.matcom.2021.05.038.

[14] A. B. Abubakar, P. Kumam, M. Malik, P. Chaipunya, and A. H. Ibrahim, "A hybrid fr-dy conjugate gradient algorithm for unconstrained optimization with application in portfolio selection," *AIMS Math.*, vol. 6, no. 6, pp. 6506–6527, 2021.

[15] S. Enshaei, W. June Leong, and M. Farid, "Diagonal quasi-Newton method via variational principle under generalized Frobenius norm," *Optim. Methods Softw.*, vol. 31, no. 6, pp. 1258–1271, Nov. 2016.

[16] C. Yang, C. Chen, N. Wang, Z. Ju, J. Fu, and M. Wang, "Biologically inspired motion modeling and neural control for robot learning from demonstrations," *IEEE Trans. Cogn. Develop. Syst.*, vol. 11, no. 2, pp. 281–291, Jun. 2019.

[17] R.-E. Precup, R.-C. David, R.-C. Roman, A.-I. Szedlak-Stinean, and E. M. Petriu, "Optimal tuning of interval type-2 fuzzy controllers for nonlinear servo systems using slime mould algorithm," *Int. J. Syst. Sci.*, vol. 4, pp. 1–16, Jun. 2021.

[18] A. Adamu, D. Kitkuan, A. Padcharoen, C. E. Chidume, and P. Kumam, "Inertial viscosity-type iterative method for solving inclusion problems with applications," *Math. Comput. Simul.*, vol. 194, pp. 445–459, Apr. 2022.

[19] C. E. Chidume, A. Adamu, P. Kumam, and D. Kitkuan, "Generalized hybrid viscosity-type forward-backward splitting method with application to convex minimization and image restoration problems," *Numer. Funct. Anal. Optim.*, vol. 4, pp. 1–22, Aug. 2021.

[20] H. Mohammad and M. Y. Waziri, "Structured two-point stepsize gradient methods for nonlinear least squares," *J. Optim. Theory Appl.*, vol. 181, no. 1, pp. 298–317, Apr. 2019.

[21] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA J. Numer. Anal.*, vol. 8, pp. 141–148, Jan. 1988.

[22] A. M. Awwal, P. Kumam, L. Wang, M. M. Yahaya, and H. Mohammad, "On the Barzilai–Borwein gradient methods with structured secant equation for nonlinear least squares problems," *Optim. Methods Softw.*, vol. 7, pp. 1–20, Dec. 2020.

[23] H. Mohammad and S. A. Santos, "A structured diagonal Hessian approximation method with evaluation complexity analysis for nonlinear least squares," *Comput. Appl. Math.*, vol. 37, no. 5, pp. 6619–6653, Nov. 2018.

[24] M. M. Yahaya, P. Kumam, A. M. Awwal, and S. Aji, "A structured quasi-Newton algorithm with nonmonotone search strategy for structured NLS problems and its application in robotic motion control," *J. Comput. Appl. Math.*, vol. 395, Oct. 2021, Art. no. 113582.

[25] J. E. Dennis and J. J. Moré, "Quasi-Newton methods, motivation and theory," *SIAM Rev.*, vol. 19, no. 1, pp. 46–89, 1977.

[26] H. F. Khalfan, R. H. Byrd, and R. B. Schnabel, "A theoretical and experimental study of the symmetric rank-one update," *SIAM J. Optim.*, vol. 3, no. 1, pp. 1–24, Feb. 1993.

[27] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Math. Program.*, vol. 91, no. 2, pp. 201–213, 2002.

[28] W. L. Cruz, J. Martínez, and M. Raydan. (2004). *Spectral Residual Method Without Gradient Information for Solving Large-Scale Nonlinear Systems of Equations: Theory and Experiments*. [Online]. Available: https://www.researchgate.net/publication/238730856

[29] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "Testing unconstrained optimization software," Argonne National Lab., Lemont, IL, USA, Tech. Rep. (324), 1978, p. 96.

[30] L. Lukšan and J. Vlcek, "Test problems for unconstrained optimization," Acad. Sci. Czech Republic, Inst. Comput. Sci., Chennai, India, Tech. Rep. 1(897), 2003.

[31] M. Raydan, "The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem," *SIAM J. Optim.*, vol. 7, no. 1, pp. 26–33, 1997.

[32] J. K. Liu and S. J. Li, "A projection method for convex constrained monotone nonlinear equations with applications," *Comput. Math. Appl.*, vol. 70, no. 10, pp. 2442–2453, Nov. 2015.

[33] M. Jamil and X.-S. Yang, "A literature survey of benchmark functions for global optimization problems," *Int. J. Math. Model. Numer. Optim.*, vol. 4, no 2, pp. 150–194, 2013.

[34] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "Testing unconstrained optimization software," *ACM Trans. Math. Softw.*, vol. 7, no. 1, pp. 17–41, Mar. 1981.

[35] D. S. Gonçalves and S. A. Santos, "Local analysis of a spectral correction for the gauss-Newton model applied to quadratic residual problems," *Numer. Algorithms*, vol. 73, no. 2, pp. 407–431, Oct. 2016.

[36] A. Renfrew, "Introduction to robotics: Mechanics and control," *Int. J. Electr. Eng. Educ.*, vol. 41, no. 4, p. 388, 2004.

[37] Y. Zhang, L. He, C. Hu, J. Guo, J. Li, and Y. Shi, "General four-step discrete-time zeroing and derivative dynamics applied to time-varying nonlinear optimization," *J. Comput. Appl. Math.*, vol. 347, pp. 314–329, Feb. 2019.

**MAHMOUD MUHAMMAD YAHAYA** received the B.S. degree in pure/applied mathematics from Gombe State University, Gombe, in September 2016. He is currently pursuing the master's degree by research in applied mathematics with the King Mongkut's University of Technology, Thonburi, under Prof. Poom Kumam. He participated in a one-year mandatory national service called NYSC, from January 2017 to January 2018. His research interests include convex/non-convex optimization, least-squares methods, unconstrained optimization algorithms and their applications, and applied ML/DL algorithms.

**POOM KUMAM** (Member, IEEE) received the Ph.D. degree in mathematics from Naresuan University, Thailand. He is currently a Full Professor with the Department of Mathematics, King Mongkut's University of Technology Thonburi (KMUTT). He is also the Head of the Fixed Point Theory and Applications Research Group, KMUTT, and also with the Theoretical and Computational Science Center (TaCS-Center), KMUTT. He is also the Director of the Computational and Applied Science for Smart Innovation Cluster (CLASSIC Research Cluster), KMUTT. He has provided and developed many mathematical tools in his fields productively over the past years. He has over 800 scientific articles and projects either presented or published. He also delivers many invited talks at different international conferences every year all around the world. His research interests include fixed point theory, variational analysis, random operator theory, optimization theory, approximation theory, fractional differential equations, differential game, entropy and quantum operators, fuzzy soft set, mathematical modeling for fluid dynamics and areas of interest inverse problems, dynamic games in economics, traffic network equilibria, bandwidth allocation problem, wireless sensor networks, image restoration, signal and image processing, game theory, and cryptology.

**ALIYU MUHAMMED AWWAL** received the B.Sc. degree from Gombe State University, the M.Sc. degree from Bayero University Kano, and the Ph.D. degree in applied mathematics from the King Mongkut's University of Technology Thonburi (KMUTT). He has authored or coauthored a number of research articles in high-impact journals. His current research interests include iterative algorithms for solving nonlinear problems, such as numerical optimization problems, nonlinear least squares problems, and system of nonlinear equations with applications in signal recovery, image deblurring, and motion control.

**PARIN CHAIPUNYA** received the Ph.D. degree from the King Mongkut's University of Technology Thonburi (KMUTT). He is currently affiliated with the Department of Mathematics, KMUTT. He also works with the Center of Excellence in Theoretical and Computational Science (TaCS), KMUTT. He authored or coauthored many high-quality research articles and projects in high-impact journals. His research interests include nonlinear functional analysis, general topology, and differential/metric geometry. Moreover, he also likes to work with colleagues worldwide.

**SANI AJI** received the B.Sc. degree in mathematics from Gombe State University, Nigeria, in 2015, the master's degree in interdisciplinary mathematics and mathematical modeling from the University of L'Aquila, Italy, and the master's degree in interdisciplinary mathematics and mathematical modeling from the University of Silesia, Katowice, Poland. He is currently pursuing the Ph.D. degree with the King Mongkut University of Technology Thonburi, Thailand. His research interests include solving system of nonlinear equations with applications to signal recovery, robotic motion control, image processing, machine learning, and computer vision.

**SANI SALISU** received the B.Sc. degree in mathematics from Sokoto State University, Sokoto, in October 2017, and the M.Sc. degree in pure and applied mathematics from the African University of Science and Technology, Abuja, in July 2019. He is currently pursuing the Ph.D. degree in applied mathematics with the King Mongkut's University of Technology Thonburi, under the supervision of Prof. Poom Kumam and Dr. Songpon Sriwongsa. His research interests include fixed point theory, operator theory, and nonlinear convex analysis and optimization.