TELKOMNIKA Telecommunication, Computing, Electronics and Control
Vol. 19, No. 6, December 2021, pp. 1847~1856
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v19i6.19791

An effective new iterative CG-method to solve unconstrained non-linear optimization issues

Rana Z. Al-Kawaz[1], Abbas Y. Al-Bayati[2]
[1]Department of Mathematics, College of Basic Education, University of Telafer, Tall'Afar, Mosul, Iraq
[2]University of Telafer, Tall'Afar, Mosul, Iraq

Article Info

Article history:

Received Jan 25, 2021
Revised May 25, 2021
Accepted Jun 14, 2021

Keywords:

Double-search direction
Gamma-parameter
Large-scale
Non-linear problems
Standard HWY-algorithm

ABSTRACT

In this paper, we proposed a matrix-free double-search direction based on the updated parameter file of the double-search direction with a new mathematical formula for the gamma parameter. When comparing the numerical results of this algorithm with the standard (HWY) algorithm which given by Halilu, Waziri and Yusuf in 2020. We get very robust numerical results. The proposed algorithm is devoid of derivatives to solve large-scale non-linear problems by combining two search directions in one search direction. We demonstrated the overall convergence of the proposed algorithm under certain conditions. The numerical results presented in this paper show that the new search direction is useful for solving widespread non-linear test problems.

Corresponding Author:

Rana Z. Al-Kawaz
Department of Mathematics
College of Basic Education
University of Telafer
Mosul, Iraq
Email: rana.alkawaz@yahoo.com, ranazidan@uotelafer.edu.iq

1. INTRODUCTION

Most of the applied problems derived from the following branches: engineering, biological, mathematics, physical, chemical, and the rest of the scientific branches are non-linear. Researchers have continued to develop numerical methods that solve this type of problem as in [1], [2]. In this article, we will discuss how to solve a system of nonlinear equations, which we can represent by:

$$F(x) = 0 \tag{1}$$

So, we know the function $F: R^n \to R^n$ is a nonlinear function. The premise of this function R^n refers to the real space of the dimension-n measured by the Euclidean standard $\|\cdot\|$. As for the methods that can solve such a non-linear system, they are the iterative methods, such as the Newton method, Quasi-Newton method [3]-[8], and the derivative-free method [9]-[11]. When solving in (1) using the most commonly used iterative method through the linear sequence as in [12], as the search direction d_k is obtained from (2).

$$F(x_k) + J(x_k)d_k = 0 \tag{2}$$

Where $J(x_k)$ is the jacobian matrix that equal to $F'(x_k)$ or an approximation of it. One of the good characteristics of Newton's method is the speed in reaching the optimal solution and its rapid convergence, but it requires the computation of the Jacobian matrix. The first idea of the double-search direction suggested in [13] relies on generating duplicates from the (3).

$$x_{k+1} = x_k + \alpha_k e_k + \alpha_k^2 c_k \tag{3}$$

Where $\alpha_k > 0$ is the step-size, x_{k+1} is the new point of (3), x_k is the previous point while e_k and c_k are the first and the second search directions, respectively. In this article, we will shed light on approximating a matrix $J(x_k)$ using a diagonal matrix (I is the identity matrix) i.e., that (4).

$$J(x_k) \approx \psi_k I \tag{4}$$

We can define $f(x)$ as a modular function that defines it (5).

$$f(x) = \frac{1}{2}\|F(x)\|^2 \tag{5}$$

Note that the problem of equations in (1) is equivalent to the following global optimization problem (6).

$$\min f(x), \quad x \in R^n \tag{6}$$

In the double search direction method (3), the iterative information is used in multiple steps, and curves are searched to generate new iterative points. Researchers continue to develop a special type of search direction, for example, Petrovic and Stanimirovic [14] deal with a double-direction to solve unconstrained optimization issues. The transformation of the double-step length scheme is suggested in [15], [16] to boost the numerical efficiency and global convergence properties of double-direction methods. It is also possible to compute the step-length alpha, either by using exact or inexact line searches. Thus, inexact line search [17], [18] is the most commonly used approach in this field. A fundamental requirement of line search is to minimize function values properly, i.e. to evaluate the function values (7).

$$\|F(x_{k+1})\| \leq \|F(x_k)\| \tag{7}$$

We organized the article in the following order: section 2, deals with the two new algorithms (S-RA and D-RA). Section 3 deals with introducing some new theorems that prove the convergence of the newly proposed algorithms (S-RA and D-RA). Section 4, concerns the numerical results which demonstrate the efficiency of the newly proposed algorithms when compared to the standard (HWY) algorithm. Section 5 deals with general conclusions.

2. TWO NEW ALGORITHMS (S-RA AND D-RA)

In this section, we suggest reducing the two vector directions (3) into a single that with relying on the projection technique to find that direction of research. This is made possible by allowing the two directions to be identical, i.e. $e_k = c_k$. We propose that the e_k and c_k in (3), the unique search direction is described as (8)

$$e_k = c_k = -\psi_k^{-1} F(x_k) \tag{8}$$

Now, put (8) into (3), we get (9).

$$x_{k+1} = x_k + \alpha_k(-\psi_k^{-1}(1+\alpha_k))F(x_k) \tag{9}$$

Through (9), we can conclude that the double-direction will become (10)

$$d_k = -\psi_k^{-1}(1+\alpha_k)F(x_k) \tag{10}$$

Set the acceleration parameter used in (10) as (11),

$$\psi_{k+1} = e^{\frac{\left[\|s_k\|^2 - \|y_k\|^2\right]}{\|F(x_{k+1})\|^2}} \tag{11}$$

where $y_k = F_{k+1} - F_k$ and the difference between the two-point is $s_k = x_{k+1} - x_k$. Therefore, through the (9) and (10), we can get (12).

$$x_{k+1} = x_k + \alpha_k d_k \tag{12}$$

The projection approach relies on the use of a monotone case F to accelerate and change the new point using repetition. As in the (13).

$$z_k = x_k + \alpha_k d_k \tag{13}$$

The hyperplane, as an original iterative, is (14).

$$H_k = \{x \in R^n | F(z_k)^T (x_k - z_k) = 0\} \tag{14}$$

To start using the projection technique, we use the update of the new point x_{k+1} as given in the [19], [20] to be the projection of x_k onto the hyperplane H_k. So, can be evaluated as:

$$x_{k+1} = P_\Omega[x_k - \varsigma_k F(z_k)] \tag{15}$$

$$\varsigma_k = \frac{F(z_k)^T (x_k - z_k)}{\|F(z_k)\|^2} \tag{16}$$

In the next paragraph, we will present the standard (HWY) algorithm [12] and the newly proposed algorithms which are divided into two parts: first of one (S-RA) algorithm which uses (12), (8), and (11) and the second (D-RA) algorithm which uses (12), (10), and (11). To clarify the idea of the numerical algorithms used in this research, we present the steps of each of these algorithms in detail.

2.1. Algorithm (HWY) [12]

Input: Given x_0, $\psi_0 \in (0,1)$, $\alpha > 0$, $\varepsilon = 10^{-4}, w_1$ and $w_2 > 0$, set k=0.

- Compute $F_k = F(x_k)$.
- Test the stopping criterion. If yes, then stop; otherwise, continue to the next step.
- Compute search direction d_k using (10).
- Compute step length α_k using this line-search:

$$f(x_k + \alpha_k d_k) - f(x_k) \le -w_1 \|\alpha_k F_k\|^2 - w_2 \|\alpha_k d_k\|^2 + \eta_k f(x_k)$$

- Set $x_{k+1} = x_k + \alpha_k d_k$.
- Compute $F(x_{k+1})$.
- Determine ψ_{k+1} using $\psi_{k+1} = \frac{\|y_k\|^2}{y_k^T s_k}$
- Set k=k+1, and go to 2.

2.2. New single search direction algorithm (S-RA)

Input: Given $x_0 \in \Omega$, ψ_0, r, $\sigma, \mu \in (0,1)$, $\alpha > 0$, $\varepsilon > 0$, set k=0.

- Compute $F_k = F(x_k)$ and test If $\|F_k\| \le \varepsilon$ yes, then stop; otherwise, continue to the next step.
- Compute search direction d_k (using (8)).
- Set z_k from (13) and compute step length α_k using this line-search:

$$-F(x_k + \mu\, r_k^m d_k)^T d_k \ge \sigma\, \mu\, r_k^m \|F(x_k + \mu\, r_k^m d_k)\| \|d_k\|^2 \tag{17}$$

- If $z_k \in \Omega$ and $\|F(z_k)\| \le \varepsilon$ stop, else compute x_{k+1} from (12).
- Determine ψ_{k+1} using (11).
- Set k=k+1, and go to 2.

2.3. New double search direction algorithm (D-RA)

Input: Given $x_0 \in \Omega$, ψ_0, r, σ, μ $\in$(0,1), $\alpha > 0$, $\varepsilon > 0$, set k=0.

- Compute $F_k = F(x_k)$ and test If $\|F_k\| \leq \varepsilon$ yes, then stop; otherwise, continue to the next step.
- Compute search direction d_k (using (10)).
- Set z_k from (13) and compute step length α_k using this line-search from (17).
- If $z_k \in \Omega$ and $\|F(z_k)\| \leq \varepsilon$ stop, else compute x_{k+1} from (12).
- Determine ψ_{k+1} using (11).
- Set k=k+1, and go to 2.

3. CONVERGENCE ANALYSIS

In the previous section, we proposed two new algorithms (S-RA and D-RA) depending on the parameter ψ_{k+1}. Now in this section, we will present an affinity analysis for the second algorithm, which is more general than the first as in the coming theorems, but before that, we must give the basic assumptions a space of attention which is:

3.1. Assumption A

Assumption A means that the special solution of (1) in N stands for x^*. Since $F'(x_k)$ is approximated by $\psi_k I$ along the direction s_k, we might mention another assumption of the same idea.

- Suppose there is a set level defined by:

$$\Omega = \{x \,|\, \|F(x)\| \leq \|F(x_0)\|\}$$

- There is an x^* that belongs to R^n, where $F(x^*) = 0$ is true.
- Let the function F be differentiable and continuous in some neighborhood, that is, N of x^* contained in Ω.
- On N, i.e., there is a Jacobian of F restricted and positive definite, i.e. there are a positive constants M > m > 0 are such that:

$$\|F'(x)\| \leq M, \ \forall\, x \in N. \tag{18}$$

And

$$m\|d\|^2 \leq d^T F'(x) d, \ \forall\, x \in N, d \in R^n. \tag{19}$$

3.2. Assumption B

If we consider that $\psi_k I$ is a good approximation of $F'(x_k)$, which means that:

$$\|(F'(x_k) - \psi_k I)d_k\| \leq \varepsilon \| F(x_k)\| \tag{20}$$

where $\varepsilon \in (0,1)$ is a small quantity [6].

3.3. Theorem (descent direction)

Suppose assumption B holds and that new algorithm (S-RA) and (D-RA) produces $\{x_k\}$. Then, d_k in (8) in the direction of the descent of $f(x_k)$ at x_k i.e.

$$\nabla f(x_k)^T d_k < 0 \tag{21}$$

Proof: We will divide the proof into two parts, each part concerned with an algorithm to change the search direction in each of them as follows:
Part 1: when dealing with the first algorithm, we will need a search direction from (8), as:

$$\nabla f(x_k)^T d_k = F(x_k)^T F'(x_k) d_k = F(x_k)^T [(F'(x_k) - \psi_k I)d_k - F(x_k)]$$

$$\nabla f(x_k)^T d_k = F(x_k)^T (F'(x_k) - \psi_k I)d_k - \|F(x_k)\|^2 \tag{22}$$

by Cauchy-Schwarz inequality, we have:

$$\nabla f(x_k)^T d_k \ \leq \| F(x_k)\| \|(F'(x_k) - \psi_k\, I)d_k\| - \|F(x_k)\|^2 \tag{23}$$

If (20) satisfy then,

$$\nabla f(x_k)^T d_k \ \leq \ \varepsilon\|F(x_k)\|^2 - \|F(x_k)\|^2 \leq -(1-\ \varepsilon)\|F(x_k)\|^2 \tag{24}$$

Hence for $\varepsilon \in (0,1)$, this proves of part 1 is true.
Part 2: when dealing with the second algorithm, we will need a search direction from (8) and (10), as:

$$\nabla f(x_k)^T d_k = F(x_k)^T F'(x_k) d_k = F(x_k)^T (F'(x_k) - \psi_k\, I)d_k - (1+\alpha_k)\|F(x_k)\|^2 \tag{25}$$

by Cauchy-Schwarz inequality, we have:

$$\nabla f(x_k)^T d_k \ \leq \| F(x_k)\| \|(F'(x_k) - \psi_k\, I)d_k\| - (1+\alpha_k)\|F(x_k)\|^2$$

If (20) satisfy then,

$$\nabla f(x_k)^T d_k \ \leq \ \varepsilon\|F(x_k)\|^2 - (1+\alpha_k)\|F(x_k)\|^2 \leq -(1-\ \varepsilon + \alpha_k)\|F(x_k)\|^2$$

Hence this proves part 2. This means that the two new proposed algorithms have descent search directions. We can deduce from the theorem (descent direction) that the norm function $f(x_k)$ is a decline for d_k, which implies that $\|F(x_{k+1})\| \leq \ \|F(x_k)\| \leq \ldots \leq \ \|F(x_0)\|$. This implies that $x_k \in \Omega$.

3.4. Lemma (bounded ψ_{k+1})

Suppose that assumption A holds and $\{x_k\}$ is generated by an algorithm (S-RA) and (D-RA). Then there exists a constants *M > m > 0* such that for all k:

$$\frac{[\|s_k\|^2 - \|y_k\|^2]}{\|F(x_{k+1})\|^2} \leq \frac{M - M^3}{m^2} \tag{26}$$

Proof:
From assumption A we get:

$$y_k^T\, s_k \ \geq \ m\, \|s_k\|^2 \tag{27}$$

from [18], we have:

$$M^2\, y_k^T\, s_k \ \geq \ m\, \|y_k\|^2 \tag{28}$$

then,

$$[m\, \|s_k\|^2 - m\, \|y_k\|^2] \leq [y_k^T\, s_k - M^2\, y_k^T\, s_k] \Rightarrow [\|s_k\|^2 - \|y_k\|^2] \leq \frac{[1-M^2]}{m} y_k^T s_k$$

from the theorem we have:

$$m\, \|s_k\| \ \leq \ \|F_{k+1}\| \leq \| F_{k+1} - F_k\| \leq \ M\, \|s_k\| \tag{29}$$

hence,

$$\frac{[\|s_k\|^2 - \|y_k\|^2]}{\|F(x_{k+1})\|^2} \leq \frac{\frac{[1-M^2]}{m} y_k^T s_k}{m^2 \|s_k\|^2} \ \leq \ \frac{[1-M^2]M}{m^2} \tag{30}$$

The inequality (26) is true. Using (26), ψ_{k+1} is generated by the update of (11) and we can deduce that $\psi_{k+1} I$ inherit the positive definiteness of $\psi_k I$.

3.5. Lemma (bounded d_k)

Suppose that assumption A and B holds and $\{x_k\}$ is generated by an algorithm (S-RA) and (D-RA). Then there exists a constant b>0 such that $\forall k > 0$,

$$\|d_k\| \leq b_i \tag{31}$$

where i=1,2.

Proof: We will present two parts in this lemma, each of which depends on the search direction resulting from an algorithm as in:
Part 1: From (8), (11), and assumption A we have:

$$\|d_k\| = \left\| -\frac{F(x_k)}{e^{\frac{\|s_k\|^2-\|y_k\|^2}{\|F(x_{k+1})\|^2}}} \right\| \tag{32}$$

and using the result of Lemma (bounded ψ_{k+1}),

$$\|d_k\| \leq e^{\frac{[1-M^2]M}{m^2}}\|F(x_k)\| \leq [B_1\|F(x_0)\|] \leq b_1 \tag{33}$$

where B_1>0 and $b_1 = B_1\|F(x_0)\|$.
Part 2: From (10), (11), and assumption A we have:

$$\|d_k\| = \left\| -\frac{(1+\alpha_k)F(x_k)}{e^{\frac{\|s_k\|^2-\|y_k\|^2}{\|F(x_{k+1})\|^2}}} \right\| \tag{34}$$

and using the result of Lemma (bounded ψ_{k+1}),

$$\|d_k\| \leq (1+\alpha_k)e^{\frac{[1-M^2]M}{m^2}}\|F(x_k)\| \leq [(\|F(x_0)\| + \alpha_k\|F(x_k)\|)e^{\frac{[1-M^2]M}{m^2}}] \tag{35}$$

$$\|d_k\| \leq [(\|F(x_0)\| + B_2)e^{\frac{[1-M^2]M}{m^2}}] \leq b_2 \tag{36}$$

where B_2>0 and $b_2 = (\|F(x_0)\| + B_2)e^{\frac{[1-M^2]M}{m^2}}$. The following theorem deals with the global convergence property. To prove that under a few suitable conditions, there exist an accumulation point of x_k which is a solution to the problem (1).

3.6. Theorem (global convergence)

Suppose that assumption B holds, $\{x_k\}$ is generated by an algorithm (S-RA) and (D-RA). Assume further $\forall k > 0$,

$$\alpha_k \geq \tau\frac{|F(x_k)^T d_k|}{\|d_k\|^2} \tag{37}$$

where τ is some positive constant. Then

$$\lim_{k\to\infty}\|F(x_k)\| = 0 \tag{38}$$

Proof: From (31), and (Descent Direction Theorem) we have:

$$\lim_{k\to\infty}\|s_k\| = 0 \tag{39}$$

and the bounded of $\|d_k\|$, we have:

$$\lim_{k\to\infty}\alpha_k\|d_k\|^2 = 0 \tag{40}$$

From (37) and (40) it follows that:

$$\lim_{k\to\infty} |F(x_k)^T d_k| = 0 \tag{41}$$

In this stage of the proof, we will take two parts according to the two new algorithms as in:
Part 1: According to the (S-RA) algorithm and from (8), we have:

$$F(x_k)^T d_k = -\psi_k^{-1} \|F(x_k)\|^2 \Longrightarrow |\psi_k| |F(x_k)^T d_k| = \|F(x_k)\|^2$$

and as we imposed in the theorem:

$$|\psi_k| \frac{1}{c} \alpha_k \|d_k\|^2 \geq \|F(x_k)\|^2 \tag{42}$$

While $|\psi_k| = e^{\frac{[\|s_{k-1}\|^2 - \|y_{k-1}\|^2]}{\|F(x_k)\|^2}} \leq e^{\frac{[1-M^2]M}{m^2}} \leq \delta$

so, from the (42), then

$$0 \leftarrow \frac{\delta}{c} \alpha_k \|d_k\|^2 \geq \|F(x_k)\|^2 \geq 0 \tag{43}$$

therefore, in (42) is true and the proof for part 1 is completed.
Part 2: According to the (D-RA) algorithm and using (10), we have:

$$F(x_k)^T d_k = -\psi_k^{-1} (1 + \alpha_k) \|F(x_k)\|^2$$

$$\|F(x_k)\|^2 = \|-\psi_k F(x_k)^T d_k\| - \|\alpha_k F(x_k)\|^2 \leq |\psi_k| |F(x_k)^T d_k| \tag{44}$$

while $|\psi_k| \leq \delta$ as in part 1, so from the (50), then

$$0 \leftarrow \delta |F(x_k)^T d_k| \geq \|F(x_k)\|^2 \geq 0 \tag{45}$$

therefore, the (38) is true and the proof for part 2 is completed.

4. NUMERICAL PERFORMANCE

In this section, we will present our numerical results for comparisons between the two new proposed algorithms (S-RA) and (D-RA) and the standard (HWY) algorithm which is devoid of the derivative to solve certain nonlinear test problems. In our implementing all three algorithms, we used the Matlab R2018b program in a laptop calculator with its Corei5 specifications. As for the tools used in the two algorithms, they are as follows: $\psi_0 = 0.6 \; and \; 1, r = 0.9, \sigma = 0.02, \mu = 1, w_1 = w_2 = 10^{-4}, \|F(x_k)\| < 10^{-8}$. The program finds the results on several non-derivative functions through several two initial points indicated in the Tables 1 and 2.

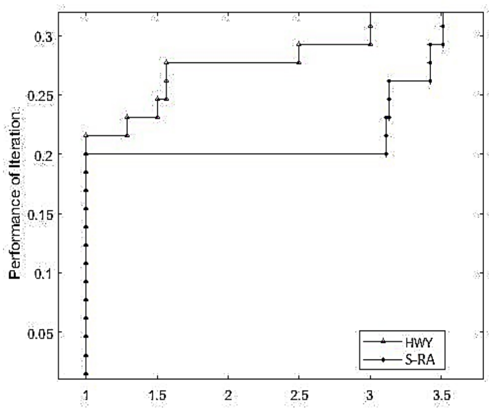
Table 1. The initial points

Name of Variable	Initial point
x_1	$(1, 1, 1, .., 1)^T$
x_2	$(0.2, 0.2, 0.2, .., 0.2)^T$
x_3	$(20, 20, 20, .., 20)^T$
x_4	$(rand, rand, rand, .., rand)^T$

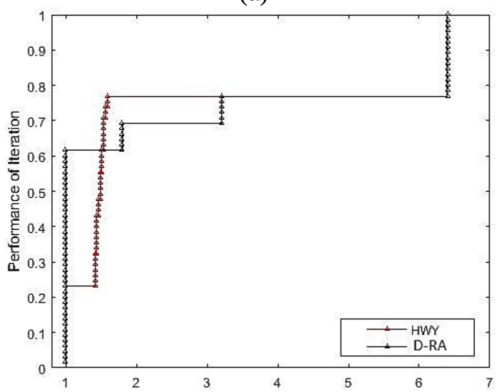
These algorithms we implemented within dimensions n (1000, 2000, 5000, 7000, 12000). All such algorithms are recognized by their performance in (Iter) the number of iterations, (Eval-F) the number of evaluations of functions, (Time) in second CPU time, (Norm) approximation solution norm. The test problems $F(x) = (f_1, f_2, f_3, \dots, f_n)^T$ where $x = (x_1, x_2, x_3, \dots, x_n)^T$, for $i = 1,2, \dots, n$ and $\Omega = R_+^n$ are from [21]-[24] and listed as shown in Table 2. Using Dolan and Mor´e style [25], the Figures 1-3 are used for comparison between the (HWY) with (S-RA) and (D-RA) algorithms when switching the search direction. The Figures 1-3 are about the initial point 1 because it is the best performance.

Table 2. Define the problems

No.	Problems
1	$F_i(x) = x_i - \sin x_i$
2	$F_i(x) = e^{x_i} - 1$
3	$F_i(x) = \sqrt{c}\,(x_1 - 1), i = 2,3,..,n-1.$ $F_n(x) = \frac{1}{4n}\sum_{j=1}^{n} x_j^2 - 1/4, c = 1 * 10^{-5}$
4	$F_i(x) = \ln(\lvert x_i \rvert + 1) - \frac{x_i}{n}$
5	$F_i(x) = \min(\min(\lvert x_i \rvert, x_i^2), \max(\lvert x_i \rvert, x_i^3))$
6	$F_1(x) = x_1 - e^{\frac{\cos(x_1 + x_2)}{n+1}}$ $F_i(x) = x_i - e^{\frac{\cos(x_{i+1} + x_i + x_{i-1})}{n+1}}, for\ i = 2,3,..,n-1$ $F_n(x) = x_n - e^{\frac{\cos(x_{n-1} + x_n)}{n+1}}$
7	$F_i(x) = \frac{i}{n} e^{x_i} - 1$
8	$F_1(x) = e^{x_1} - 1$ $F_i(x) = e^{x_i} - x_{i-1} - 1$
9	$F_i(x) = \sum_{i=1}^{n} \lvert x_i \rvert^{i}$
10	$F_i(x) = \sum_{i=1}^{n} \lvert x_i \rvert$
11	$F_i(x) = \max_{i=1,..,n} \lvert x_i \rvert$
12	$F_i(x) = \sum_{i=1}^{n} \lvert x_i \rvert \; e^{-\sum_{i=1}^{n} \sin(x_i^2)}$
13	$F_i(x) = \sum_{i=1}^{n} \lvert x_i \rvert^{i+1}$

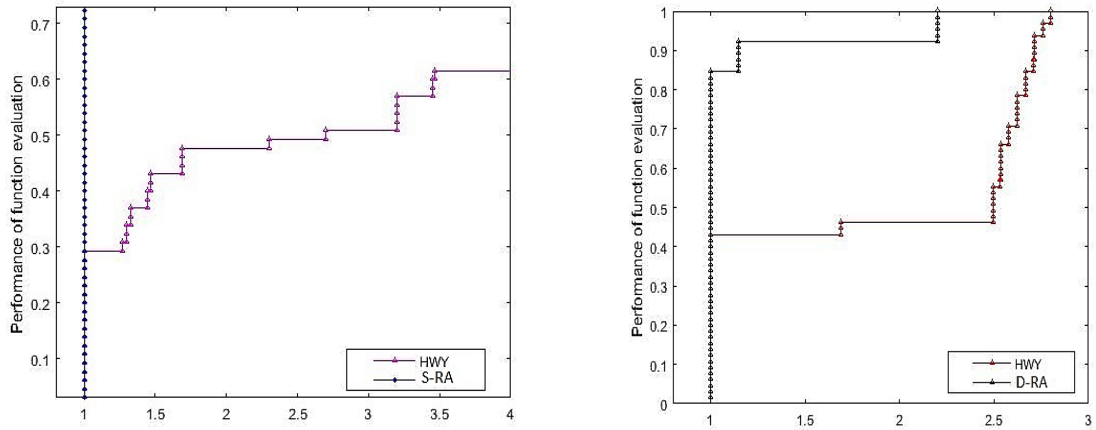


(a)



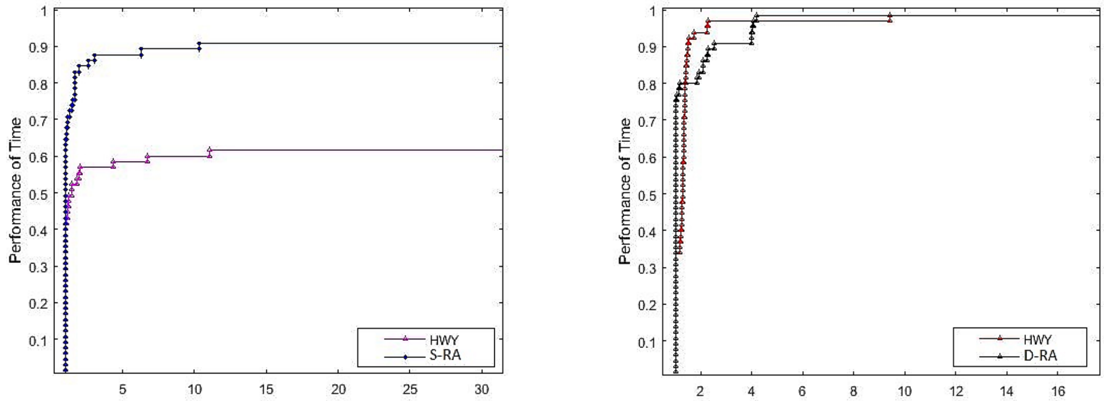
(b)

Figure 1. Performance of iterations for the (S-RA and D-RA vs. HWY) algorithms: (a) S-RA and HWY and (b) D-RA and HWY



(a) (b)

Figure 2. Performance of function evaluations for the (S-RA and D-RA vs. HWY) algorithms: (a) S-RA and HWY and (b) D-RA and HWY



(a) (b)

Figure 3. Performance of time for the (S-RA and D-RA vs. HWY) algorithms: (a) S-RA and HWY and (b) D-RA and HWY

5. CONCLUSIONS

The results, presented in the six figures show the efficiency of the two new algorithms (S-RA) and (D-RA) when compared with the previous standard (HWY) algorithm, and their efficiency is better by taking the first initial point and increase when increasing the dimensions in the variables used. The new algorithms have given a clear convergence in reaching the optimal point for solving non-linear functions.

ACKNOWLEDGEMENTS

The research is supported by the College of Computer Sciences and Mathematics, University of Mosul, and College of Basic Education, University of Telafer, Republic of Iraq. The authors declare that there are no conflicts of interest regarding this work.

REFERENCES

[1] R. Z. Al-Kawaz, A. Y. Al-Bayati, and M. S. Jameel, "Interaction between un-updated FR-CG algorithms with an optimal Cuckoo algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1497-1504, 2020, doi: 10.11591/ijeecs.v19.i3.pp1497-1504.

[2] A. A. Al-Arbo and R. Z. Al-Kawaz, "Implementation of a combined new optimal cuckoo algorithm with a gray wolf algorithm to solve unconstrained optimization nonlinear problems," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no 3, pp. 1582-1589, 2020, doi: 10.11591/ijeecs.v19.i3.pp1582-1589.

[3] J. E. Dennis and R. B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Philadelphia: SIAM, 1996, doi: 10.1137/1.9781611971200. [Online]. Available: https://epubs.siam.org/doi/pdf/10.1137/1.9781611971200.fm
[4] M. W. Yusuf, L. W. June, and M. A. Hassan, "Jacobian-free diagonal Newton's method for solving nonlinear systems with singular Jacobian," *Malaysian Journal of Mathematical Sciences*, vol. 5, no. 2, pp. 241-255, 2011. [Online]. Available: https://einspem.upm.edu.my/journal/fullpaper/vol5no2/7.%20waziri.pdf
[5] Q. R. Yan, X. Z. Peng and D. H. Li, "A globally convergent derivative-free method for solving large-scale nonlinear monotone equations," *Journal of computational and applied mathematics*, vol. 234, no. 3, pp. 649-657, June 2010, doi: 10.1016/j.cam.2010.01.001.
[6] G. Yuan and X. Lu, "A new backtracking inexact BFGS method for symmetric nonlinear equations," *Computers and Mathematics with Application*, vol. 55, no. 1, pp. 116-129, January 2008, doi: 10.1016/j.camwa.2006.12.081.
[7] D. Li and M. Fukushima, "A global and superlinear convergent Gauss-Newton based BFGS method for an asymmetric nonlinear equation," *SIAM Journal of Numerical Analysis*, vol. 37, no. 1, pp. 152-172, 1998, doi: 10.1137/S0036142998335704. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.48.2857&rep=rep1&type=pdf
[8] G. Fasano, F. Lampariello and M. Sciandrone, "A truncated non-monotone Gauss-Newton method for large-scale nonlinear least-squares problems," *Computational Optimization and Application*, vol. 34, pp. 343-358, 2006, doi: 10.1007/s10589-006-6444-2.
[9] A. B. Abubakar, P. Kumam, and H. Mohammad. "A note on the spectral gradient projection method for nonlinear monotone equations with applications," *Computational and Applied Mathematics*, vol. 39, no. 129, pp. 1-35, 2020, doi: 10.1007/s40314-020-01151-5.
[10] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Indust. Appl. Math.*, vol. 11, no. 2, pp. 431-441, 1963, doi: 10.1137/0111030.
[11] K. Levenberg, "A method for the solution of certain non-linear problems in the least-squares," *Quarterly Applied Mathematics*, vol. 2, no. 2, pp. 164-166, 1944. [Online]. Available: https://www.jstor.org/stable/43633451
[12] I. Yusuf, A. S. Halilu and M. Y. Waziri, "Efficient matrix-free direction method with line search for solving a large-scale system of nonlinear equations," *Yugoslav Journal of Operations Research*, vol. 30, no. 4, pp. 394-412, 2020, doi: 10.2298/YJOR160515005H.
[13] N. I. Duranovic-Milicic, "A multi-step curve search algorithm in nonlinear optimization," *Yugoslav Journal of Operations Research*, vol. 18, no. 1, pp. 47-52, 2008, doi: 10.2298/YUJOR0801047D. [Online]. Available: http://yujor.fon.bg.ac.rs/index.php/yujor/article/view/300/191
[14] M. J. Petrović and P. S. Stanimirović, "Accelerated Double Direction Method for Solving Unconstrained Optimization Problems," *Mathematical Problems in Engineering*, vol. 2014, 2014, doi: 10.1155/2014/965104.
[15] C.G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Mathematics of Computation*, vol. 19, no. 92, pp. 577-593, 1965. [Online]. Available: https://www.jstor.org/stable/2003941
[16] A. S. Halilu, M. K. Dauda, M. Y. Waziri, and M. Mamat, "A derivative-free decent method via acceleration parameter for Solving systems of nonlinear equations," *Open Journal of Science and Technology*, vol. 2, no. 3, pp. 1-4, 2019, doi: 10.31580/ojst.v2i3.931.
[17] A. Bouaricha and R. B. Schnabel, "Tensor methods for large sparse systems of nonlinear equations," *Mathematical Programming*, vol. 82, pp. 377-400, 1998, doi: 10.1007/BF01580076.
[18] W. Zhou and D. Shen, "An inexact PRP conjugate gradient method for symmetric nonlinear equations," *Numerical Functional Analysis and Optimization*, vol. 35, no. 3, pp. 370-388, 2014, doi: 10.1080/01630563.2013.871290.
[19] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, University of Auckland, NZ: SIAM, 1970; 30, doi: 10.1137/1.9780898719468.
[20] M. V. Solodov and B. F. Svaiter, "A globally convergent inexact Newton method for systems of monotone equations," In: *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*. *Applied Optimization*, M. Fukushima and L. Qi, Eds., Boston, MA, vol. 22, 1998, pp. 355-369, doi: 10.1007/978-1-4757-6388-1_18.
[21] X. S. Yang, "Test Problems in Optimization*," Engineering Optimization: An Introduction with Metaheuristic Applications*, Wiley, 2010, doi: 10.1002/9780470640425.
[22] M. Jamil and X. S. Yang, "A literature survey of benchmark functions for global optimization problems," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 4, no. 2, pp. 150-194, 2013, doi: 10.1504/IJMMNO.2013.055204.
[23] Dai, Z., & Zhu, H. "A modified Hestenes-Stiefel-type derivative-free method for large-scale nonlinear monotone equations," *Mathematics*, vol. 8, no. 2, pp. 168, 2020, doi.org/10.3390/math8020168.
[24] R. Z. Al-Kawaz and A. Y. Al-Bayati, "Show off the efficiency of the dai-Liao method in merging technology for monotonous non-linear problems," *Indonesian Journal of Electrical Engineering and Computer Sciences*, vol. 21, no. 1, pp. 505-515, 2021, doi: 10.11591/ijeecs.v21.i1.pp505-515.
[25] E. D. Dolan and J. J. Mor´e, "Benchmarking optimization software with performance profiles," *Math. Program.*, vol. 91, pp. 201-213, 2002, doi: 10.1007/s101070100263.