An adaptive search direction algorithm for the modified projection minimization optimization

Rana Z. Al-Kawaz¹ and Abbas Y. Al-Bayati²*

¹Department of Mathematics, College of Basic Education, University of Telafer, Tall’Afar, Mosul, Iraq.
²University of Telafer, Tall’Afar, Mosul, Iraq.

E-mails: profabbasbayati@yahoo.com

Abstract. In this paper, we proposed a double-search direction algorithm with a free of derivatives based on a different parameter update (ηₖ) which gives us an update on the double search trend. We used anewly updated formula for the projection parameter within the formula for the proposed search direction. When comparing our numerical results for the proposed algorithm with some standard published algorithms we obtain efficient numerical results. The proposed algorithm, especially, is used to solve large-scale nonlinear problems by combining two search directions in one search direction. Also, to demonstrate the general convergence of the proposed new algorithm under some circumstances. The numerical performance of the new proposed algorithm on some nonlinear test functions proved the efficiency of this algorithm.

1. Introduction

Optimization issues took a lot of space in different disciplines such as engineering, geosciences, physics, mechanics, medicine, and so on, but the difficulty of these issues was that they were of the non-linear and complex type. At present, researchers have been able to overcome this problem through numerical optimization methods of derivative-free to solve such problems as in [1-6]. The most prominent and widespread use of derivative-free methods for solving a system of nonlinear equations, which we can represent through:

\[ F(x) = 0 \]  

(1)

Knowing that the function \( F: \mathbb{R}^n \rightarrow \mathbb{R}^n \) is a nonlinear function. The premise of this function \( \mathbb{R}^n \) refers to the real space of the dimension-\( n \) measured by the Euclidean standard \( ||.|| \). Researchers have given methods to solve this system in equation (1) through derivative-free methods as in [7-10]. Iterative methods start with a starting point \( x_0 \) and then continue to update the points through the:

\[ x_{k+1} = x_k + \alpha_k d_k \]  

(2)

But these iterations equation (2) dependent on the step \( \alpha_k \) and the search direction \( d_k \) may continue without reaching the goal, so there is a method for updating the points based on the quadratic form for the length of the step, known as:

\[ x_{k+1} = x_k + \alpha_k e_k + \alpha_k^2 c_k \]  

(3)

The first idea of the double-search direction equation (3) suggested in [11] where \( \alpha_k > 0 \) is the step-size, \( x_{k+1} \) is the new point of ‘equation (3)’, \( x_k \) is the previous point while \( e_k \) and \( c_k \) are the first and the second search directions, respectively. The search direction \( d_k \) in equation (2) is obtained from:

\[ F(x_k) + J(x_k)d_k = 0 \]  

(4)

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 fast convergence, but it requires the Jacobian matrix to be computed. It is best to use a suitable Jacobian matrix \( J(x_k) \) approximation such that transformation:

\[ J(x_k) \cong \eta_k I \]  

(5)
Where \( I \) is the identity matrix and \( f(x) \) is a modular function that defines:
\[
f(x) = \frac{1}{2} \| F(x) \|^2
\]  
(6)

Note that the problem of the equation in (1) is equivalent to the optimization problem:
\[
\min f(x), \quad x \in R^n
\]  
(7)

When the double-search direction method is observed in equation (3), the iterative information is used in multiple steps, and the curves are searched to generate new repeat points. Researchers continue to develop a special type of search trend, for example, Petrović and Stanimirović [12] deal with a double-direction to solve unconstrained optimization issues. The double-step length system transformation is suggested in [13] to boost the numerical efficiency and global convergence properties of double-direction methods. In [14], Halilu and Waziri recently suggested an improved matrix-free procedure for solving nonlinear equation structures by the approach of double-step length. Ibrahim’s [15] suggested method can be interpreted as an extension using the projection technique of the three-term modified Polak-Ribiére-Polyak and the three-term Hestenes-Stiefel conjugate gradient method. Researchers continued to develop as in [16–21]. A prerequisite of line search is to properly reduce function values, that is, to evaluate function values:
\[
\| F(x_{k+1}) \| \leq \| F(x_k) \|
\]  
(8)

The projection approach relies on the use of a monotone case \( F \) to accelerate equation (2) and change the new point using repetition. As in:
\[
z_k = x_k + \alpha_k d_k
\]  
(9)

The hyperplane, as an original iterative, is:
\[
H_k = \{ x \in R^n | F(z_k)^T (z_k - x_k) = 0 \}
\]  
(10)

To start using the projection technique, we use the new point \( x_{k+1} \) as given in the [22, 23] to be the projection of \( x_k \) onto the hyperplane \( H_k \). So, can be evaluated as:
\[
x_{k+1} = P_{H_k}[x_k - \delta_k F(z_k)]
\]  
(11a)
\[
\delta_k = \frac{F(z_k)^T (z_k - x_k)}{\| F(z_k) \|^2}
\]  
(11b)

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

2. Two New Modified Algorithms (MS-RA and MD-RA).

In this section, we suggest reducing the two search directions equation (3) into a single one to locate the study direction by depending on the projection technique. This is made possible by having the two directions be similar, i.e. \( e_k = c_k \). We say that the unique search path of the \( e_k \) and \( c_k \) in equation (3) is defined as:
\[
e_k = c_k = -\eta_k^{-1} F(x_k)
\]  
(12)

Put equation (13) into equation (3), we get:
\[
x_{k+1} = x_k + \alpha_k (-\eta_k^{-1} (1 + \alpha_k)) F(x_k)
\]  
(13)

From equation (13), we can conclude that the double-direction will become:
\[
d_k = -\eta_k^{-1} (1 + \alpha_k) F(x_k)
\]  
(14)

To get acceleration in the new algorithm we will impose the parameter \( \eta_{k+1} \) used in equation (14) as:
\[ \eta_{k+1} = e^{\eta(x_k)} \]  

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 two equations (13) and equation (14), we can get:

\[ x_{k+1} = x_k + \alpha_k d_k \]  

Sun and Liu and Wang[24] presented a multi-parameterized conjugated gradient (SLW) algorithm for solving monotone equations with convex constraints based on the projection update formula that:

\[ x_{k+1} = P_{\Omega} \left[ x_k - \beta_\delta_k \tau F(x_k) + F(z_k) \right] \]  

\[ \delta_k = \frac{F(x_k)^T(x_k - z_k)}{\|\tau F(x_k) + F(z_k)\|^2} \]

Where \( \theta \) and \( \tau \) any positive value. In this article, we used an improved projection method equation (17) and the new parameter \( \eta_{k+1} \) in equation (15) for two new algorithms to be designed in the points as the difference between the new algorithm and the algorithm (SLW):

1- Integration of the new single-search direction equation (12) with the newly updated parameter equation(15) in the new point equation (16) to obtain (MS-RA).

2- Merging the new double-search direction equation (14) with the newly updated parameter equation (15) in the new point equation (16) to obtain (MD-RA).

We present the steps of each of these algorithms in-depth to explain the concept of the numerical algorithms used in this research.

2.1. New Modified Single Search Direction Algorithm (MS-RA)

Step 0: Given \( x_0 \in \Omega \eta_0, r, \sigma, \mu, \tau \in (0,1), \alpha > 0, \epsilon > 0 \), set \( k=0 \).

Step 1: Compute \( F_k = F(x_k) \) and test if \( \|F_k\| \leq \epsilon \) yes, then stop; otherwise, continue to the next step.

Step 2: Compute search direction \( d_k \) ‘equation (12)’.

Step 3: Set \( z_k \) from equation (9) and compute step length \( \alpha_k \) using this line-search:

\[ -F(x_k + \mu r_k^m d_k)^T d_k \geq \sigma \mu r_k^m \|\tau F(x_k) + F(x_k + \mu r_k^m d_k)\|d_k \|^2 \]  

Step 4: If \( z_k \in \Omega \) and \( \|F(z_k)\| \leq \epsilon \) stop, else compute \( x_{k+1} \) from ‘equation (16)’.

Step 5: Determine \( \eta_{k+1} \) using ‘equation (15)’.

Step 6: Set \( k=k+1 \), and go to 2.

2.2. New Modified Double Search Direction Algorithm (MD-RA)

Step 0: Given \( x_0 \in \Omega \eta_0, r, \sigma, \mu, \tau \in (0,1), \alpha > 0, \epsilon > 0 \), set \( k=0 \).

Step 1: Compute \( F_k = F(x_k) \) and test if \( \|F_k\| \leq \epsilon \) yes, then stop; otherwise, continue to the next step.

Step 2: Compute search direction \( d_k \) equation (14).

Step 3: Set \( z_k \) from (13) and compute step length \( \alpha_k \) using this line-search from equation (18).

Step 4: If \( z_k \in \Omega \) and \( \|F(z_k)\| \leq \epsilon \) stop, else compute \( x_{k+1} \) from equation (16).

Step 5: Determine \( \eta_{k+1} \) using equation (15).

Step 6: Set \( k=k+1 \), and go to 2.

3. Convergence Analysis.

In the previous section, we proposed two new algorithms (MS-RA and MD-RA) depending on the parameter \( \eta_{k+1} \) in equation (15). Now in this section, we will present several theorems to prove the global convergence property of the two new proposed algorithms, but before that, we must give the basic assumptions some amount of attention which is

3.1. Assumption A

- Suppose there is a set level defined by:

\[ \Omega = \{x|\|F(x)\| \leq \|F(x_0)\|\} \]

- There is \( x^* \) that belongs to \( \mathbb{R}^n \), where \( F(x^*) = 0 \) is true.
Let the $F$ function be continuously and differentiable in some neighborhood $N$, i.e. $N$ of $x^*$ contained in $\Omega$.

On $N$, i.e., there is a positive definite Jacobian of $F$ restricted, i.e. there are positive constants $M > m > 0$ such that:

$$
\|F'(x)\| \leq M, \ \forall \ x \in N.
$$

(19)

and

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

(20)

Assumption A means that the special solution of equation (1) in $N$ stands for $x^*$. Since $F'(x_k)$ is approximated by $\eta_k I$ along the search directions $s_k$.

### 3.2. Assumption B

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

$$
\|(F'(x_k) - \eta_k I) d_k\| \leq \epsilon \|F(x_k)\|
$$

(21)

where $\epsilon \in (0,1)$ is a small quantity for more information on equation (21) see [25].

### 3.3. Theorem (Descent Direction)

Suppose Assumption B holds and the new algorithms (MS-RA) and (MD-RA) produces $\{x_k\}$. Then, $d_k$ in equation (12) and equation (14) in the direction of the descent of $f(x_k)$ at $x_k$, i.e.

$$
\nabla f(x_k)^T d_k < 0
$$

(22)

**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 (MS-RA) algorithm, we will need a search direction from the equation (12), as:

$$
\nabla f(x_k)^T d_k = F(x_k)^T F'(x_k) d_k
$$

$$
= F(x_k)^T [(F'(x_k) - \eta_k I) d_k - F(x_k)]
$$

$$
\nabla f(x_k)^T d_k = F(x_k)^T (F'(x_k) - \eta_k I) d_k - \|F(x_k)\|^2
$$

(23)

by Cauchy-Schwarz inequality, we have:

$$
\nabla f(x_k)^T d_k \leq \|F(x_k)\| \|(F'(x_k) - \eta_k I) d_k\| - \|F(x_k)\|^2
$$

(24)

If equation (21) satisfy then,

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

(25)

Hence for $\epsilon \in (0,1)$, this proves of part 1 is true.

Part 2: when dealing with the (MD-RA) algorithm, we will need a search direction from equation (12) and equation (14), as:

$$
\nabla f(x_k)^T d_k = F(x_k)^T F'(x_k) d_k
$$

$$
= F(x_k)^T [(F'(x_k) - \eta_k I) d_k - (1 + \alpha_k) F(x_k)]
$$

$$
\nabla f(x_k)^T d_k = F(x_k)^T (F'(x_k) - \eta_k I) d_k - (1 + \alpha_k) \|F(x_k)\|^2
$$

(26)

by Cauchy-Schwarz inequality, we have:

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

If equation (21) satisfy then,

$$
\nabla f(x_k)^T d_k \leq \epsilon \|F(x_k)\|^2 - (1 + \alpha_k) \|F(x_k)\|^2 \leq -(1 - \epsilon + \alpha_k) \|F(x_k)\|^2
$$
Hence this proves part 2. This means that the newly proposed algorithms (MS-RA) and (MD-RA) have descent search directions.

3.4. Remark

From the Theorem (descent direction), we can deduce that the norm functions \( f(x_k) \) 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 \( x_k \in \Omega \).

3.5. Lemma (Bounded \( \eta_{k+1} \))

Suppose Assumption A holds and \( \{x_k\} \) is algorithm-generated (MS-RA) and \( \{x_k\} \) (MD-RA). So, a constant \( M > m > 0 \) exists, such that for all \( k \):

\[
\frac{\|s_k\|^2 - \|y_k\|^2}{e^\|F(x_{k+1})\|^2} \leq \frac{(1-M^2)}{B}
\]

**Proof:**

We get From Assumption A:

\[
y_k^T s_k \geq m \|s_k\|^2
\]

from [26], we have:

\[
M^2 y_k^T s_k \geq m \|y_k\|^2
\]

then,

\[
[m \|s_k\|^2 - m \|y_k\|^2] \leq [y_k^T s_k - M^2 y_k^T s_k]
\]

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

from the above Remark we have:

\[
m \|s_k\| \leq \|F_{k+1}\| \leq \|F_{k+1} - F_k\| \leq M \|s_k\|
\]

(30)

Let \( e^{m^2\|s_k\|^2} \cong C \) and \( \|s_k\|^2 \cong D \). Hence,

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

The inequality (27) is true. Using (27), \( \eta_{k+1} \) is generated by the update of formula (15) and we can deduce that \( \eta_{k+1} \) inherit the positive definiteness of \( \eta_k \).

3.6. Lemma (Bounded \( d_k \))

Suppose the assumptions A and B are retained and the algorithm \( \{x_k\} \) is generated (MS-RA) and \( \{x_k\} \) (MD-RA). Then a \( b>0 \) constant exists such that \( \forall k > 0 \),

\[
\|d_k\| \leq b_i
\]

(31)

where \( i=1,2 \).

**Proof:**

In this lemma, we will present two elements, each of which relies on the search direction resulting from an algorithm, while in:

Part 1: We have equations (12,15), and Assumption A:

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

(32)

and using the result of Lemma (Bounded \( \eta_{k+1} \)).
\[ \| d_k \| \leq e^{\frac{[1-M^2]}{c}}\| F(x_k)\| \] (33)
\[ \| d_k \| \leq [B_1\| F(x_0)\|] \] (34)
\[ \| d_k \| \leq b_1 \] (35)

where \( B_1 > 0 \) and \( b_1 = B_1\| F(x_0)\| \).

Part 2: From equations (14,15), and Assumption A we have:
\[ \| d_k \| = \left\| -(1 + \alpha_k) e^{-\frac{(\| s_k \|^2 - \| y_k \|^2)}{c}} F(x_k) \right\| \] (36)
and using the same procedure used in the first part,
\[ \| d_k \| \leq (1 + \alpha_k) e^{\frac{[1-M^2]}{c}}\| F(x_k)\| \] (37)
\[ \| d_k \| \leq \left[ ([\| F(x_0)\| + \alpha_k\| F(x_k)\|)]e^{-\frac{[1-M^2]}{c}} \right] \] (38)
\[ \| d_k \| \leq \left[ ([\| F(x_0)\| + B_2)e^{-\frac{[1-M^2]}{c}} \right] \] (39)
\[ \| d_k \| \leq b_2 \] (40)

where \( B > 0 \) and \( b_2 = ([\| F(x_0)\|] + B_2)e^{-\frac{[1-M^2]}{c}} \).

The theorem below deals with the global property of convergence. To show that there is an accumulation point of \( x_k \) under a few appropriate conditions, which is a solution to the problem equation (1).

### 3.7. Theorem (Global Convergence).

Supposing Assumption B holds, the algorithm \( \{x_k\} \) (MS-RA) is generated and the algorithm (MS-RA) is generated (MD-RA). Assume for more \( \forall k > 0 \),
\[ \alpha_k \geq \zeta \frac{|F(x_k)^T d_k|}{\|d_k\|^2} \] (41)
where \( \zeta \) is some positive constant. Then
\[ \lim_{K \rightarrow \infty} \| F(x_k) \| = 0 \] (42)

**Proof:**
We have from equation (31), and (Descent Direction Theorem)
\[ \lim_{K \rightarrow \infty} \| s_k \| = 0 \] (43)
and the bounded of \( \| d_k \| \), we have:
\[ \lim_{K \rightarrow \infty} \alpha_k \| d_k \|^2 = 0 \] (44)

From equation (41) and (44) it follows that:
\[ \lim_{K \rightarrow \infty} |F(x_k)^T d_k| = 0 \] (45)

We will take two parts according to the two new algorithms at this point of the proof, as in:

Part 1: According to the (MS-RA) algorithm and from equation (12), we have:
\[ F(x_k)^T d_k = -\eta_k^{-1}\| F(x_k)\|^2 \]
\[-\eta_k F(x_k)^T d_k = \|F(x_k)\|^2\]
\[
|\eta_k|F(x_k)^T d_k = \|F(x_k)\|^2
\]

and as in the theorem, we imposed:
\[
|\eta_k| \frac{1}{c} \alpha_k \|d_k\|^2 \geq \|F(x_k)\|^2
\] (46)

While
\[
|\eta_k| = \left| \frac{\|y_{k-1}\|^2 - \|y_{k-1}\|^2}{e^{\|F(x_k)\|^2}} \right| \leq e^{\frac{1 - M_k^2}{c}} \leq \rho
\]

so, from the equation (46), then
\[
0 \leftarrow \frac{\rho}{c} \alpha_k \|d_k\|^2 \geq \|F(x_k)\|^2 \geq 0
\] (47)

Consequently, the equation (42) is valid and the proof for Part 1 is complete.

Part 2: According to the (MD-RA) algorithm and using equation (14), we have:
\[
F(x_k)^T d_k = -\eta_k^{-1}(1 + \alpha_k)\|F(x_k)\|^2
\] (48)
\[
-\eta_k F(x_k)^T d_k = (1 + \alpha_k)\|F(x_k)\|^2
\]
\[
\|F(x_k)\|^2 = \|-\eta_k F(x_k)^T d_k\| - \|\alpha_k F(x_k)\|^2
\] (49)
\[
\|F(x_k)\|^2 \leq |\eta_k| |F(x_k)^T d_k|
\] (50)

while \(|\eta_k| \leq \delta\) as in part 1, so from the equation (50), then
\[
0 \leftarrow \delta \|F(x_k)^T d_k\| \geq \|F(x_k)\|^2 \geq 0
\] (51)

The equation (42) is then valid and the evidence for Part 2 is complete.

4. Numerical Performance.
Our computational results for comparisons between the two recent proposed algorithms (MS-RA) and (MD-RA) and the two standard algorithms (HWY) [26] and (SLW) [24] that are devoid of the derivative for solving such nonlinear test problems will be presented in this section. In our implementation of all three algorithms in a laptop calculator, we used the Matlab R2018b program with its Corei5 requirements. As for the instruments used in the two algorithms, such that:

| Parameters | Value of new algorithms | Value of old algorithms |
|------------|-------------------------|-------------------------|
| \(\eta_0\)  | 0.3                     | 0.5                     |
| \(\mu\)    | 0.5                     | 0.5, 0.9                |
| \(r\)      | 0.9                     | 0.9                     |
| \(\sigma\) | 0.02                    | 0.01                    |
| \(w_1 = w_2\) | \(10^{-4}\)            |                         |
| \(\tau\)  | 0.1                     | 0.1                     |
| \(\vartheta\) | 1.7                    | 1.7                     |

\[
\|F(x_0)\| < 10^{-8}
\]

The results of several non-derivative functions are found by the program through several initial points indicated in the table below:

| Variables | Starting point |
|-----------|----------------|
|          |                |
These algorithms we implemented within dimensions \( n \) (1000, 2000, 5000, 7000, 12000). All such algorithms are recognized by their performance in:

| Variable | Starting point |
|----------|----------------|
| \( x_1 \) | \((1,1,1,\ldots,1)^T\) |
| \( x_2 \) | \((0,2,0,2,0,2,\ldots,0.2)^T\) |
| \( x_3 \) | \((20,20,20,\ldots,20)^T\) |
| \( x_4 \) | \((\text{rand}, \text{rand}, \text{rand}, \ldots, \text{rand})^T\) |

Table 3: Tools of comparison

| Iter | The number of iterations |
|------|--------------------------|
| Eval-F | The number of functions evaluations |
| Time | TimeCPU in the second |

The test problems \( F(x) = (f_1, f_2, f_3, \ldots, f_n)^T \) where \( x = (x_1, x_2, x_3, \ldots, x_n)^T \), for \( i = 1,2,\ldots,n \) and \( \Omega = R^d_n \) are from [27,28] and listed as follows:

- **Problem 1:** \( F_i(x) = x_i - \sin x_i \)
- **Problem 2:** \( F_i(x) = e^{x_i} - 1 \)
- **Problem 3:** \( F_i(x) = \sqrt{e} (x_i - 1), i = 2,3,\ldots,n-1 \) : \( F_n(x) = \frac{1}{4n} \sum_{j=1}^{n} x_j^2 - 1/4, c = 1 * 10^{-5} \).
- **Problem 4:** \( F_i(x) = \ln(|x_i| + 1) - \frac{x_i}{n} \)
- **Problem 5:** \( F_i(x) = \min(\min(|x_i|, x_i^2), \max(|x_i|, x_i^3)) \)
- **Problem 6:** \( F_i(x) = x_i - e^{\frac{\cos(x_i + x_j)}{n+1}} \)
  \( F_i(x) = x_i - e^{\frac{\cos(x_i + x_j)}{n+1}}, f o r i = 2,3,\ldots,n - 1 \)
  \( F_n(x) = x_n - e^{\frac{\cos(x_n + x_n)}{n+1}} \)
- **Problem 7:** \( F_i(x) = \frac{i}{n} e^{x_i} - 1 \)
- **Problem 8:** \( F_i(x) = e^{x_i} - 1; F_i(x) = e^{x_i} - x_{i-1} - 1 \)
- **Problem 9:** \( F_i(x) = \sum_{i=1}^{n} |x_i|^i \)
- **Problem 10:** \( F_i(x) = \sum_{i=1}^{n} |x_i| \)
- **Problem 11:** \( F_i(x) = \max |x_i| \)
- **Problem 12:** \( F_i(x) = \sum_{i=1}^{n} |x_i| e^{-\sum_{i=1}^{n} \sin(x_i^2)} \)
- **Problem 13:** \( F_i(x) = \sum_{i=1}^{n} |x_i| e^{i+1} \)

Figures (1, 2, and 3) are used to equate (HWY) and (SLW) with (MS-RA) and (MD-RA) algorithms when flipping the search path from 1 to 2 using the Dolan and Mor'e style [29]. Two initial points, the first 1, 2, 3, and 4, were compared. The figures are around the initial point 1 since they work better in such a way that:
Figure 1. Iterations efficiency for the (MS-RA) with ((a) HWY and (b) SLW) algorithms.

Figure 2. Function evaluation efficiency for the (MS-RA) with ((a) HWY and (b) SLW) algorithms.

Figure 3. Time efficiency for the (MS-RA) with ((a) HWY and (b) SLW) algorithms.
Figure 4. Iteration efficiency for the (MD-RA) with (a) HWY and (b) SLW algorithms.

Figure 5. Function evaluation efficiency for the (MD-RA) with (a) HWY and (b) SLW algorithms.

Figure 6. Time efficiency for the (MD-RA) with (a) HWY and (b) SLW algorithms.
The first part describes the output of the first new algorithm (MS-RA) with a repeated point with one search direction equation (12), where we show the figures (1), (2), and (3) which we compared with the output of the (HWY and SLW) algorithm also with one search direction, and the comparison was rendered in three ways (the number of iterations, the number of times the function evaluations is calculated, time spent in implementation).

As for the second part, represented by the figures (4), (5), and (6), the second new algorithm was used, which replaced one search direction equation (12) with two search directions in the previous iterative point (14), and the contrast was made with the first new algorithm in the same normalization steps.

5. Conclusions.
The results in the six figures within the numerical performance section show how efficiently the two new algorithms (MS-RA) and (MD-RA) are when we compare their performance with the previous standard algorithms (HWY and SLW). The direction of research in the two new algorithms gave us faster results with fewer iterations when we used the first point, no matter how much we increased the dimensions of the variables. Also, the two algorithms converge to reach the optimum point for solving nonlinear functions.

6. References.
[1] Ahmed H I Al-Kawaz R Z and Al-Bayati A Y 2019 Spectral Three-Term Constrained Conjugate Gradient Algorithm for Function Minimizations Journal of Applied Mathematics 2019
[2] Hamed E T Al-Kawaz R Z and Al-Bayati A Y 2020 New Investigation for the Liu-Story Scaled Conjugate Gradient Method for Nonlinear Optimization Journal of Mathematics 2020
[3] Al-Kawaz R Z Al-Bayati A Y and Jameel M S 2020 Interaction between un updated FR-CG algorithms with an optimal Cuckoo algorithm Indonesian Journal of Electrical Engineering and Computer Science 19 1497-1504
[4] Al-Arbo A A and Al-Kawaz R Z 2020 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 19 1582-1589
[5] Al-Arbo A A and Al-Kawaz R Z 2021 A fast spectral conjugate gradient method for solving nonlinear optimization problems Indonesia Journal of Electrical Engineering and Computer Science 21 429-439
[6] Al-Kawaz R Z and Al-Bayati A Y 2021 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. 21 505-515
[7] Abubakar A B Kumam P and Mohammad H 2020 A note on the spectral gradient projection method for nonlinear monotone equations with applications Computational and Applied Mathematics 39 1-35
[8] Halilu A S and Waziri M Y 2018 An improved derivative-free method via a double direction approach for solving systems of nonlinear equations Journal of the Ramanujan Mathematical Society 33 75-89
[9] Abdullahi H Halilu A S and Waziri M Y 2018 A Modified Conjugate Gradient Method via a Double Direction Approach for solving large-scale Symmetric Nonlinear Systems Journal of Numerical Mathematics and Stochastics 1 32-44
[10] Halilu A S Waziri M Y and Musa Y B 2020 Inexact Double Step Length Method For Solving Systems Of Nonlinear Equations Statistics, Optimization & Information Computing 8 165-174
[11] Đuranović-Miličić N 2008 A multi-step curve search algorithm in nonlinear optimization Yugoslav Journal of Operations Research 18 47-52
[12] Petrović M J and Stanimirović P S 2014 Accelerated double direction method for solving unconstrained optimization problems Mathematical Problems in Engineering 2014
Acknowledgments.
The research is supported by the College of Computer Sciences and Mathematics, University of Mosul, and College of Basic Education, University of Telafer, Mosul, Republic of Iraq. The authors declare that there are no conflicts of interest regarding this work.
## Authors’ Background

| Your Name          | Title*          | Research Field                        |
|--------------------|-----------------|---------------------------------------|
| Rana Z. Al-Kawaz   | Ph.D. Candidate | Numerical Optimization                |
| Abbas Y. Al-Bayati | Full Professor  | Computer Oriented Numerical Optimization |