Abstract. In this work, we propose modification for PSB update with a new extended Quasi – Newton condition for unconstrained optimization problem, so it’s called \( \alpha \) method. PSB update kind of rank two update, which solve the unconstrained optimization problem, but this update can’t guarantee the positive definite property of Hessian matrix. In this work, the guarantee of positive definite property for the Hessian matrix be confirmed by updating the vector \( s_k \), which represent the difference between the next gradient and the current gradient of the objective function which assume to be continuous twice and differentiable. Then we proved the existentialism of this update. Numerical results are reported where the comparison between the proposed method and the PSB update method under standard problems.

Keyword: PSB update, Quasi – Newton, Hessian matrix.

1. Introduction

Today’s Quasi – Newton methods are considered one of the most effective methods to solve non-linear unconstrained or bounded constrained optimization problem.

These methods are mostly used when the second derivative matrix of the target function is either un available or too expensive to compute. It is very similar to Newton's method, but avoid the need of computing Hessian matrices by repeating a symmetric matrix that, from repetition to repetition can be considered as an approximation for Hessian. Thus, it allow the curvature of the problem to be exploited in the numerical algorithm, despite the fact that only first derivative (gradients) and the values of the functions are required \[2\]. The Quasi-Newton methods are very useful and effective methods for solving the unconstrained minimization problem

\[
\min f(x) \; ; \; x \in \mathbb{R}^n \quad \cdots (1)
\]
Where \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is continuously twice and differentiable. Beginning from the point \( x_0 \) and a symmetric and positive definite matrix \( B_0 \), a Quasi-Newton method generates sequence \( \{x_k\} \) and \( \{B_k\} \) by the iteration of the form

\[
x_{k+1} = x_k - B_k^{-1}\nabla f(x_k), \quad k = 0, 1, 2 \ldots \quad \text{... (2)}
\]

where \( B_k \in \mathbb{R}^{n \times n} \) is satisfying the following formula of Quasi-Newton condition

\[
B_{k+1}s_k = y_k \quad \text{... (3)}
\]

where

\[
s_k = \lambda_k d_k, \quad y_k = g_{k+1} - g_k \quad \text{... (4)}
\]

where the \( \lambda_k \) is the step length and \( d_k \) is a search direction we got it by solving the equation

\[
d_k = -B_k^{-1}g_k \quad \text{... (5)}
\]
in which \( g_k = \nabla f(x_k) \) is the gradient of \( f(x) \) at \( x_k \) and \( B_k \) is an approximation to the Hessian matrix \( G_k = \nabla^2 f(x_k) \). The updating matrix \( B_k \) is wanted to satisfy the Quasi-Newton equation (3) with (4). So that \( B_{k+1} \) is a reasonable approximation to \( G_{k+1} \).

Some researchers have also presented some important researches see [4], [5], [6].

PSB update is important in both theoretical and practical research computing. But, the disadvantage is that the PSB update can't keep the Positive definiteness of updates is detrimental to their computing performance. Luckily, The disadvantage can be avoided if we use the trust region range with PSB updated [7].

2. \( \alpha \)-PSB method

PSB update it is officially known as the Powell-symmetric-Broyden update is important in theoretical research and practical computing. But, the disadvantages that PSB update can't retain the positive definiteness of updates hurts its performance in computing [7].

PSB update it can written as

\[
B_{k+1} = B_k + \frac{(y_k - B_k s_k)s_k^T + s_k(y_k - B_k s_k)^T}{s_k^Ts_k} - \frac{(y_k - B_k s_k)^Ts_k}{(s_k^Ts_k)^2} s_k s_k^T
\]

The PSB method tries to update the Hessian matrix by using the formula (2.1) which represent the solution of the Quasi-Newton condition (3) but the update is not preserve the positive definite property that means there is no guarantee to minimize the function at each iteration, so if the current Hessian matrix approximation is positive definite then, the next Hessian matrix approximation may be
not positive definite and hence this iteration must be deleted. The problem that the PSB update will be not positive is possibly in this part \((- \frac{(y_k - B_k s_k)^T s_k}{(s_k^T s_k)^2} s_k s_k^T)\) from (2.1).

By taking this part from equation (2.1) and let \(z \neq 0\), then
\[
\begin{aligned}
  z^T \left( - \frac{(y_k - B_k s_k)^T s_k}{(s_k^T s_k)^2} s_k s_k^T \right) Z &= - \frac{(y_k - B_k s_k)^T s_k}{(s_k^T s_k)^2} Z s_k s_k^T Z \\
  &= - \frac{(y_k - B_k s_k)^T s_k}{(s_k^T s_k)^2} (s_k^T Z) (s_k^T Z) = - \frac{(y_k - B_k s_k)^T s_k}{(s_k^T s_k)^2} \|s_k^T Z\|^2
\end{aligned}
\]

Since \(\|s_k^T Z\|^2\) is positive and \((s_k^T s_k)^2\) positive, then we have the last part \((y_k - B_k s_k)^T s_k\) must be negative.

Now we will processing this problem by multiplying the vector \(s_k\) by a constant number \(\alpha_k\) and determine the value of \(\alpha_k\) to guarantee the positive definite property because it is very important to guarantee the existence of the minimizer of the objective function, because the hessian matrix is symmetric \((\mathbf{f} \text{ continuous})\), so the symmetric property very important to guarantee the convergent of \(B_{k+1}\) to the original hessian matrix.

In this modification we guarantee the positive definite property for the Hessian matrix by updating the vector \(s_k\) by multiplication with \(\alpha_k\), so \(s_k\) will become \(\alpha_k s_k\).

Now we take this part \((y_k - B_k s_k)^T s_k\) of PSB update from (2.1) to guarantee this part is positive
\[
s_k = \alpha_k s_k \quad (2.2)
\]

Let \((y_k - B_k \alpha_k s_k)^T \alpha_k s_k < 0\) \quad (2.3)
\[
[ \alpha_k y_k^T s_k - \alpha_k^2 s_k^T B_k s_k < 0 ] + \alpha_k
\]
\[
y_k^T s_k - \alpha_k s_k^T B_k s_k < 0 \quad (2.5)
\]
\[
y_k^T s_k < \alpha_k s_k^T B_k s_k \quad (2.6)
\]

Then \(\alpha_k\) will become
\[
\alpha_k > \frac{y_k^T s_k}{s_k^T B_k s_k} \quad (2.7)
\]

The choose of \(\alpha_k\) with respect to (2.7) can be as follows
\[
\alpha_k = \frac{y_k^T s_k}{s_k^T B_k s_k} + \text{random}(\text{uniform}[1:100]) \in \mathbb{R} \quad (2.8)
\]
where the random uniform is a random number between [1:100] it is distributed in a uniform distribution and we add it to (2.7) by trial to guarantee the matrix is positive definite.

The $\alpha$ – PSB update consists of iteration of the form (2) where $d_k$ is the search direction of the form (5) and the Hessian approximation $B_k$ is update by the $\alpha$ – PSB method.

Now we replace every $s_k$ by $\alpha_k s_k$ in (2.1) and hence we get

$$B_{k+1} = B_k + \frac{(y_k - \alpha_k B_k s_k)s_k^T + s_k(y_k - \alpha_k B_k s_k)^T}{\alpha_k s_k^T s_k} - \frac{(y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$

(2.9)

### 2.1. Theorem

The Hessian matrix $B_{k+1}$ produced by $\alpha$ – PSB update

$$B_{k+1} = B_k + \frac{(y_k - \alpha_k B_k s_k)s_k^T + s_k(y_k - \alpha_k B_k s_k)^T}{\alpha_k s_k^T s_k} - \frac{(y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$

is symmetric.

Proof:

$$B_{k+1} = B_k + \frac{(y_k - \alpha_k B_k s_k)s_k^T + s_k(y_k - \alpha_k B_k s_k)^T}{\alpha_k s_k^T s_k} - \frac{(y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$

$$B_{k+1}^T = B_k^T + \frac{s_k^T (y_k - \alpha_k B_k s_k)^T + (y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k s_k^T s_k} - \frac{s_k^T (y_k - \alpha_k B_k s_k)^T}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$

$$B_{k+1}^T = B_k + \frac{s_k^T (y_k - \alpha_k B_k s_k)^T + (y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k s_k^T s_k} - \frac{s_k^T (y_k - \alpha_k B_k s_k)^T}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$

Since $B_k^T = B_k$, $s_k^T = s_k$ and $(y_k - \alpha_k B_k s_k)^T = (y_k - \alpha_k B_k s_k)$

We have

$$B_{k+1}^T = B_k + \frac{s_k^T (y_k - \alpha_k B_k s_k)^T + (y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k s_k^T s_k} - \frac{s_k^T (y_k - \alpha_k B_k s_k)}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$

$$B_{k+1}^T = B_k + \frac{s_k^T (y_k - \alpha_k B_k s_k)^T + (y_k - \alpha_k B_k s_k)^T s_k}{\alpha_k s_k^T s_k} - \frac{s_k^T (y_k - \alpha_k B_k s_k)}{\alpha_k (s_k^T s_k)^2} s_k s_k^T$$
There for $B_{k+1} = B_{k+1}^T$

Hence $B_{k+1}$ is symmetric.

2.2 Lemma: (Powel – Symmetric – Broyden update)

Let $A \in \mathbb{R}^{n \times n}$ be symmetric and $as, y \in \mathbb{R}^n$ with $s \neq 0$. Consider the set

$$\mathcal{B} = \{ B \in \mathbb{R}^{n \times n} : Bas = y, B = B^T \}$$  \hspace{1cm} (2.10)

Where $\alpha = \frac{ys^T}{s^T Bs} + \text{random}(\text{uniform} \ 1: \ 100) \in \mathbb{R}$

if $\alpha s^Ty \neq 0$ then there exist $B \in \mathcal{B}$ such that

$$\| A - B \|_F \leq \| A - C \|_F, \text{ for all } C \in \mathcal{B}$$

More ever $B$ has the following form

$$B = A + \frac{w s^T + w s^T}{\alpha s^T s} - (w^T s) \frac{s s^T}{(s^T s)^2} , \quad w = y - Aas$$

Then $B$ is a rank two perturbation of the matrix $A$.

Proof:

First of all notice that $\mathcal{B}$ is not empty, in fact

$$\frac{1}{\alpha s^T y} yy^T \in \mathcal{B} \ , \text{ substituting in (2.10) we get} \left[ \frac{1}{\alpha s^T y} yy^T \right] s = y$$

So that the problem is not empty. Next we reformulate the problem as a constrained minimum problem:

$$\arg \min_{A \in \mathbb{R}^{n \times n}} \frac{1}{2} \sum_{i,j=1}^{n} (A_{ij} - B_{ij})^2 \text{ subject to } Bas = y \text{ and } B = B^T$$

The solution is a stationary point of the lagrangian:

$$g(B, \lambda, M) = \frac{1}{2} \| A - B \|_F^2 + \lambda^T (Bas - y) + \sum_{i < j} (B_{ij} - B_{ji})$$

Taking the gradient we have

$$\frac{\partial}{\partial B_{ij}} g(B, \lambda, M) = A_{ij} - B_{ij} + \lambda_i \alpha s_j + M_{ij} = 0$$

Where

$$M_{ij} = \begin{cases} \mu_{ij} & \text{if } i < j \\ -\mu_{ij} & \text{if } i > j \\ 0 & \text{if } i = j \end{cases}$$
The previous equality can be written in matrix form as
\[ B = A + \lambda \alpha s^T + M. \]

Imposing symmetry for B
\[ A + \lambda \alpha s^T + M = A^T + \alpha s \lambda^T + M^T = A + \alpha s \lambda^T - M \]

Solving for M we have
\[ M = \frac{\alpha s \lambda^T - \lambda \alpha s^T}{2} \]

Substituting in B we have
\[ B = A + \frac{\alpha s \lambda^T - \lambda \alpha s^T}{2} \]

Imposing \( \alpha^2 s^T Bs = \alpha s^T y \)
\[ \alpha^2 s^T As + \frac{\alpha^3 s^T s \lambda^T s + \alpha^3 \lambda s \lambda^T s}{2} = s^T y \]

\[ \Rightarrow \lambda^T \alpha s = \frac{s^T w}{\alpha^2 s^T s} \]

where \( w = y - A\alpha \). Imposing \( Bas = y \)
\[ A\alpha s + \frac{\alpha^2 s \lambda^T s + \alpha^2 \lambda^T s}{2} = y \]

\[ \Rightarrow \lambda = \frac{2w}{\alpha^2 s^T s} - \frac{(s^T w)s}{\alpha^2 (s^T s)^2} \]

Next we compute the explicit form of B.

Substituting \( \lambda = \frac{2w}{\alpha^2 s^T s} - \frac{(s^T w)s}{\alpha^2 (s^T s)^2} \) in
\[ B = A + \frac{\alpha s \lambda^T - \lambda \alpha s}{2} \]

we obtain
\[ B = A + \frac{ws^T + sw^T}{\alpha s^T s} - (w^T s) \frac{ss^T}{\alpha (s^T s)^2} \]

\[ w = y - A\alpha s \]

The matrix B is a minimum, in fact
\[ \| B - A \|_F = \left\| \frac{ws^T + sw^T}{\alpha s^T s} - (w^T s) \frac{ss^T}{\alpha (s^T s)^2} \right\|_F \]

To bound this norm we need the following properties of Frobenius norm:
\[ \| M - N \|_F^2 = \| M \|_F^2 + \| N \|_F^2 - 2M * N \]

where \( M * N = \sum_{ij} M_{ij}N_{ij} \) setting
\[ M = \frac{ws^T + sw^T}{\alpha s^T s} \quad \text{and} \quad N = (w^T s) \frac{ss^T}{\alpha (s^T s)^2} \]
Now we compute \( \|M\|_F \), \( \|N\|_F \) and \( M \ast N \)

\[
M \ast N = \frac{w^T s}{\alpha^5 (s^T s)^3} \sum_{ij} (w_i \alpha s_j + w_j \alpha s_i) \alpha^2 s_i s_j
\]

\[
= \frac{w^T s}{\alpha^5 (s^T s)^3} \left[ \sum_i (w_i \alpha s_i) \sum_j \alpha^2 s_j^2 + \sum_j (w_j \alpha s_j) \sum_i \alpha^2 s_i^2 \right]
\]

\[
= \frac{w^T s}{\alpha^5 (s^T s)^3} \left[ (w^T \alpha s)(\alpha^2 s^T s) + (w^T \alpha s)(\alpha^2 s^T s) \right]
\]

\[
= \frac{2(w^T s)^2}{\alpha^2 (s^T s)^2}
\]

To bound \( \|N\|_F^2 \) and \( \|M\|_F^2 \) we need the following properties of Frobenius norm:

1. \( \|uv^T\|_F^2 = (u^T u)(v^T v) \);
2. \( \|uv^T + uv^T\|_F^2 = 2(u^T u)(v^T v) + 2(u^T v)^2 \);

Then we have

\[
\|N\|_F^2 = \frac{(w^T s)^2}{\alpha^6 (s^T s)^3} \|\alpha^2 ss^T\|_F^2 = \frac{(w^T s)^2}{\alpha^6 (s^T s)^4} (\alpha^2 s^T s)^2 = \frac{(w^T s)^2}{\alpha^2 (s^T s)^2}
\]

\[
\|M\|_F^2 = \frac{w s^T + s w^T}{\alpha s^T s} = \frac{2(w^T w)(s^T s) + 2(s^T w)^2}{\alpha^2 (s^T s)^2}
\]

Putting all together and using Cauchy-Schwartz inequality

\( \langle a^T b \rangle \leq \|a\|_F \|b\|_F \)

\[
\|M - N\|_F^2 = \frac{(w^T s)^2}{\alpha^2 (s^T s)^2} + \frac{2(w^T w)(s^T s) + 2(s^T w)^2}{\alpha^2 (s^T s)^2} - \frac{4(w^T s)^2}{\alpha^2 (s^T s)^2}
\]

\[
= \frac{2(w^T w)(s^T s) + (w^T s)^2}{\alpha^2 (s^T s)^2}
\]

\[
\leq \frac{w^T w}{s^T s} \frac{\|s\|^2}{\|w\|^2} \quad \quad [\text{used Cauchy – Schwartz}]
\]

\( w = y - A \alpha s \) and noticing that \( y = Cas \) for all \( C \in \mathcal{B} \). So

\[
\|w\| = \|y - A \alpha s\| = \|Cas - A \alpha s\| = \|(C - A) \alpha s\|
\]

To bound \( \|(C - A) \alpha s\| \) we need the following property of Frobenius norm:

\[
\|Mx\| \leq \|M\|_F \|x\| \;
\]
in fact

\[ \|Mx\|^2 = \sum_i \sum_j M_{ij} s_j \leq \sum_i \left( \sum_j M_{ij}^2 \right) \left( \sum_k \alpha^2 s_k^2 \right) \]

\[ = \|M\|^2 \|\alpha s\|^2 \]

Using this inequality

\[ \|M - N\|_F \leq \frac{\|w\|}{\|\alpha s\|} = \frac{\|(C - A)\alpha s\|}{\|\alpha s\|} \leq \frac{\|C - A\|_F \|\alpha s\|}{\|\alpha s\|} \]

i.e. we have \( \|A - B\|_F \leq \|C - A\|_F \) for all \( C \in \mathcal{B} \).

3. Algorithm of \( \alpha - \text{PSB} \) update

Step(1): Given an initial point \( x_0 \in \mathbb{R}^{n \times n} \) and an initial symmetric and positive definite matrix \( B_0 \in \mathbb{R}^{n \times n} , \quad \epsilon = 0.05 \), let \( k = 0 \).

Step(2): Compute \( g_k = \nabla f(x_k) \),

set \( d_k = -B_k^{-1} g_k \);

Step(3): Carry out a line search along \( d_k \) getting \( \lambda_k > 0 \);

Step(4): Set \( x_{k+1} = x_k + \lambda_k d_k \);

Step(5): Determine \( \alpha_k \) by equation (2.8);

Step(6): Determine \( B_{k+1} \) by (2.9), where \( s_k \) and \( y_k \) are defined by Q-N equation (3).

Step(7): If \( \|g_k\| < \epsilon \) then stop.

Step(8): \( k = k + 1 \), go to step (2).

4. Numerical results:

In this part, we devoted to numerical experiments.

We compare the performance of the modified PSB algorithm with the performance of the standard PSB algorithm, using the same starting points and convergence criteria and limits. We wrote the computer programs by MATLAP version (5.3). The reason for choosing them is that the problems appear to have been used in standard problems in the most of the literature, and these functions are a result of application in the branch of technology and industry. The performance of \( (\alpha - \text{PSB}) \) method has been tested and compared with standard PSB method using the following test functions. For the sake of uniformity in comparison. The results are presented in table (4-1).

The test function are chosen as follows

1- Extended Himmelblaa function [1]

\[ f(x) = \sum_{i=1}^{n/2} \left( x_{2i-1}^2 + x_{2i} - 11 \right)^2 + \left( x_{2i-1} + x_{2i} - 7 \right)^2. \]
2- Extended Rosen rock function [1]
\[ f(x) = \sum_{i=1}^{n/2} c \left( x_{2i} - x_{2i-1} \right)^2 + (1 - x_{2i-1})^2. \]
\[ c = 100 \]

3- Freudenstein and Roth function [3]
\[ f(x) = \{-13 + x_1 + [(5 - x_2)x_2 - 2|x_2|^2 + \{-29 + x_1 + [(x_2 + 1)x_2 - 14|x_2|^2\} \]

4- Trigonometric function, [7]
\[ 100(x_2 - \sin x_2)^2 + 0.25x_1^2 \]

5- Rosen brock function, [7]
\[ f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \]

6- Rosen brock’cliff function [8]
\[ f(x) = 10^{-4}(x_4 - 3)^2 - (x_1 - x_2) + e^{20(x_1 - x_2)} \]

7- Extended Rosen brock function, [7]
\[ f(x) = \sum_{i=1}^{n-1} \left[ 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \right] \]

8- Wood function, [7]
\[ f(x) = 100(x_1^2 - 2x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4)^2
\[ + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1) \]

9- Least square equation for two dimensions [1]
\[ f(x) = (2x_1 + 3x_2 - 5)^2 + (5x_1 - 2x_2 - 3)^2 \]

10- Cube function, [7]
\[ f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \]

The table (4-1) gives the comparison between the usual PSB update method and the modified \( \alpha - \text{PSB} \) for convex optimization, for our selected test functions. At the first, we give the number of iterations and function evaluation of each function from deferent starting point.

| Function | Starting points | Dim. | Feval. | Iter. | Feval. | Iter. |
|----------|-----------------|------|--------|-------|--------|-------|
| 1        | [-1; 5; -1; 5; -1; 5] | 6    | 118.2017 | 2     | 1.4767e-013 | 7     |
| 1        | [0.2; 0.2; 0.2; 0.2; 0.2; 0.2; 0.2] | 8    | 262.0737 | 2     | 1.4280e-011 | 12    |
| 2        | [0.5; 0.5; 0.5; 0.5] | 4    | 0.058   | 4     | 4.5249e-010 | 10    |
| 2        | [-1.0; -1.0; -1.0] | 6    | NaN     | 84    | 0.9755  | 73     |
| 3        | [-0.5; 0.5]      | 2    | 84.2573 | 24    | 23.7463 | 10     |
| 3        | [-1.2; -1.2]     | 4    | 0.0014  | 23    | 4.9882e-011 | 13    |
| 4        | [0.1; 5]         | 2    | 372.5254 | 17    | 1.2067  | 10     |
| 4        | [0.3; -4; 0.3; -4] | 4    | 19.7351 | 70    | 4.8097e-008 | 55    |
| 5        | [0.2; 0.2; 0.2]  | 4    | 0.0491  | 4     | 1.4462e-010 | 10    |
Now we discuss the results between $\alpha$ – PSB update method and PSB update. Note that the function number (1) at the starting point [-1; 5; -1; 5; -1; 5] from the table (4-1) that was resolved by $\alpha$ – PSB method it is better than the same function that was resolved by PSB update, because Feval for $\alpha$ – PSB update method is (1.4767e-013) but the Feval for PSB update is (118.2017). And we notice the preference in converging to the rest of the functions number (2),(3),(4),(5),(7),(8),(9) and (10) Also in the function number (2) with starting point [-1;0;-1;0;-1;0] and the function number (6) with starting point [-1;0] we notice, the function solved by $\alpha$ – PSB method but can't be solved by PSB method. PSB can't terminate at the minimum point, because the Hessian matrix is not positive definite at every iteration, that means there is some iterations do not finding the minimizer at this iteration, so the objective function can't have a descent direction.

5. Conclusion
In this paper, we proposed an update formula to modify PSB method with an extended Quasi-Newton condition (3) for unconstrained optimization problem.

We used a new equivalent method (2.9), to update the Hessian matrix. We develop a new formula of PSB update method which makes the system convergence to a minimum by generating a positive definite Hessian matrix ($B_{k+1}$) in each iteration. This new formula called modified PSB update ($\alpha$ – PSB) . This formula of ($\alpha$ – PSB) update method meets the modifying of Quasi-Newton condition over $s_k$ condition as the derivation of this formula relied on this vector and then is given suitable algorithm, in addition to resolving. Then we proved the symmetric property for ($\alpha$ – PSB) in theorem (2.1) and we proved the existentialism of this update in lemma (2.2). Then we achieved numerically convergence in Table (4.1) and we noticed that the function number (2) with starting point [-1;0;-1;0;-1;0] and the function number six with starting point [-1;0] are solved by ($\alpha$ – PSB) method but can't be solved by PSB method. And the rest of all the functions in Table (4.1) have a better convergence for the ($\alpha$ – PSB) method than the PSB method.
References

[1] Al-Bayati, A., (1991), "A new family of self-scaling variable metric Algorithms for unconstrained optimization", *Journal of Educ. and Sci., Iraq, Vol. 12, pp. 25-54.
[2] Byrd R., Nocedal J., and Yuan Y., (2000), "Global convergence of a class of quasi-Newton methods on convex problem", *SIAM, Vol. 12, Pages 1095-1115.
[3] Freudenstein F. and Roth B., (1963), "Numerical solution of system of nonlinear equations, *Journal of ACM, Vol. 10, No. 4, pp. 550-556.
[4] Mahmood, S. S., & Muhanah, N. S. (2019). Symmetric and Positive Definite Broyden Update for Unconstrained Optimization. *Baghdad Science Journal, 16(3).
[5] Mahmood, S. S., Mansour, A. I., & Abdrazak, B. T. (2011). $\alpha$ BFGS UPDATE FOR UNCONSTRAINED OPTIMIZATION. *Journal of college of Education, (1).
[6] Saad S. and Samira H. (2017), "On Modified DFP Update for Unconstrained Optimization", *American Journal of Applied Mathematics, Vol. 5, No. 1.
[7] Sun W. and Yuan Y., (2006), "Optimization Theory and Method: Nonlinear Programming", *Vol. 1 of Springer optimization and its Applications, Springer, New York, NY, USA.
[8] Todd M. J., (1984), "Quasi-Newton update in abstract spaces", *SIAM Review, 26" 367-377."