Tikrit Journal of Pure Science
ISSN: 1813 – 1662 (Print) --- E-ISSN: 2415 – 1726 (Online)
Journal Homepage: http://tjps.tu.edu.iq/index.php/j

Anew Conjugate Gradient Algorithm Based on The (Dai-Liao) Conjugate Gradient Method

SHAHER QAHTAN HUSSEIN[1] , GHASSAN EZZULDDIN ARIF[1] , YOKSAL ABDLL SATTAR[2]
[1] Department of Mathematics , College of Education for pure Science , University of Tikrit , Tikrit , Iraq
[2] Department of Mathematics , College of Science , University of Kirkuk , Kirkuk , Iraq
DOI: http://dx.doi.org/10.25130/tjps.25.2020.019

ARTICLE INFO.
Article history:
-Received: 1 / 9 / 2019
-Accepted: 25 / 11 / 2019
-Available online: / / 2019
Keywords: Search direction, Descent, Convergence, Conjugate gradient methods, Optimization problem, Line search, Exact line search, Inexact line search, Algorithm.
Corresponding Author:
Name:
SHAHER QAHTAN HUSSEIN
E-mail:
shahir.aljbory@gmail,com
Tel:

ABSTRACT

In this paper we can derive a new search direction of conjugating gradient method associated with (Dai-Liao method) the new algorithm becomes converged by assuming some hypothesis. We are also able to prove the Descent property for the new method, numerical results showed for the proposed method is effective comparing with the (FR, HS and DY) methods.

1. Introduction

Conjugated gradient methods represent an important class of optimization algorithms that are not constrained by strong local and global convergence characteristics and simple memory requirements. For 50 years now, researchers continue to express their particular interest in convergence performance and the ease of representation of algorithms in computer programs in a consistent manner. These methods are efficient in solving issues of large dimensions in unrestricted optimization **[1].**

Attention is given to the methods of conjugate gradient for two reasons.

1-These methods are among the oldest and best known techniques for solving equation systems Linear large dimensions are called linear conjugate vector methods

2-These methods can be adapted to solve non-linear optimization issues

The conditions mentioned above apply to the conjugate vector algorithms to solve the system of linear equations and are called the method of linear conjugate directions However, our study is limited to finding the lower end of nonlinear functions, and therefore we look at the types of conjugate direction algorithms (CG) to find the minimum end of functions in a brief way.

which made them popular for engineers and mathematicians are engaged in solving large-scale problems in the following form:

$\min f(x)$, $x \in R^n$ ….(1.1)

Where $f: R^n \to R$ is smooth nonlinear function and its gradient is available . The iterative formula of CG .method is given by $S_k = x_{k+1} - x_k$, $k=1,2,\ldots$

In which α_k is step – length to be computed by line search procedure and d_k is the search direction defend by

$d_1 = -g_1$, $d_{k+1} = -g_{k+1} + \beta_k s_k$, $k = 1,2,..$(1.2)

Where $g_k = \nabla f(x_k)$ and β_k is parameter called the contumacy condition, The step – length α_k is chosen to satisfy certain line search condition .

For general nonlinear function ,different choices of β_k lead to different conjugate gradient methods ,will-known formulas for β_k are called the Fletcher-

Reeves(FR) [**13**], Hestenes-Stiefel(HS)**[7],** Polak-Ribier(PR)[14], are given by :

$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}$, $\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$, $\beta_k^{PR} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}$ …. (1.3)

Where $y_k = g_{k+1} - g_k$ *and* $\|\bullet\|$ denote the Euclidian norm .

The line search in conjugate gradient algorithms is often based on standard wolf condition :

$f(x_k + \alpha_k d_k) \le f(x_k) + c_1 \alpha_k g_k^T d_k$ … (1.4)

$|g(x_k + \alpha_k d_k)^T| \le c_2 |g_k^T d_k|$ …. (1.5)

Where d_k is a descent direction and $0 < c_1 \le c_2 < 1$.

However for some conjugate gradient algorithms.**[11 and 12**]

2.1 method of Liao-Dai (Dai and Liao's Method 2001)

The search direction d_k for many unconstrained optimization methods, including Quasi Newton (QN), and method BFGS (memory less) and BFGS (Limited Memory), can be written as**[2]**

$d_{k+1} = -H_{k+1} g_{k+1}$ … (2.1)

wher H_k is positive definite matrix of type achieves the QN equation:

$H_{k+1} y_k = s_k$ …. (2.2)

It $s_k = \alpha_k d_k$ represents a step. Using (2.1) and (2.2) we get

$d_{k+1}^T y_k = -(H_{k+1} g_{k+1})^T y_k = -g_{k+1}^T (H_{k+1} y_k) = -g_{k+1}^T s_k$ … (2.3)

Since in this case $g_{k+1}^T s_k = 0$ with (ELS) then the previous relationship leads to the fulfillment of the conjugation condition. However, applied numerical algorithms usually adopt inexact line search (ILS) instead of an Exact line search (ELS). For this reason, it seems more appropriate to replace the conjugation condition

$d_{k+1}^T y_k = 0$ … (2.4)

With the following condition:-

$d_{k+1}^T y_k = -t g_{k+1}^T s_k$ …. (2.5)

$t \ge 0$ is a numerical quantity to ensure that the search direction d_k in (2.5) fulfills the conjugation condition

(2.5) by multiplying (2.5) by y_k and by using (2.5) Produce

$\beta_{k+1}^{DL} = \frac{g_{k+1}^T (y_k - t s_k)}{d_k^T y_k}$ … (2.6)

it is clear that

$\beta_{k+1}^{DL} = \beta_{k+1}^{HS} - t \frac{g_{k+1}^T s_k}{d_k^T y_k}$ … (2.7)

Both(Dai and Liao) proposed an amendment to version(2.7) from the point of view of global convergence of general functions, which limits the first term to non-negative values

$\beta_{k+1}^{DL} = Max\left\{\frac{g_{k+1}^T y_k}{d_k^T y_k}, 0\right\} - t \frac{g_{k+1}^T s_k}{d_k^T y_k}$ .. (2.8)

The length of the step α_k can be obtained by using any form of the search line. The two strong Wolfe conditions (1.4) *and* (1.5)

i.e.(4) and (5), are commonly used in conjugated gradient methods.

Both(Dai and Liao) have demonstrated the overall convergence of this method, and for further explanation see the source **[2].**

2.2 Method of Yabe-Takano (Yabe and Takano's Method 2003) [3]

The Quasi- Newton method is defined as another way to solve the problem of unconstrained optimization. This method generates a series of vectors $\{x_k\}$ and the matrix $\{B_k\}$ Using iteration (2.9)

$x_{k+1} = x_k + \alpha_k d_k$ …. (2.9)

and an update formula for the matrix B_k , when d_k representing the direction of the search and obtained by solving the linear system of equations $B_k d_k = -g_k$, B_k represents an approximation of the invers Hessian matrix $\nabla^2 f(x_k)$, the matrix B_{k+1} usually requires Secant Condition**[3]**

$\beta_{k+1} s_k = y_k$ … (2.10)

That is obtained from Tyler's expansion of the gradient vector, as $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$

Zhang, Deng and Chen (1999) and Zhang and Xu (2001) expanded the secant requirement (2.10) and proposed the following modified secant condition :-

$\beta_{k+1} s_k = \hat{y}_k$ … (2.11)

$\hat{y}_k = y_k + \frac{\theta_k}{s_k^T u_k} u_k$ …. (2.12)

$\theta_k = 6(f(x_k) - f(x_{k+1})) + 3(g_k + g_{k+1})^T s_k$ … (2.13)

$u_k \in R^n$ represents any vector that achieves $s_k^T u_k \neq 0$. Taking H_{k+1} an inverse Hessian approximation, the modified cut-off condition as follows

$H_{k+1} \hat{y}_k = s_k$ … (2.14)

$\hat{y} = y_k + \frac{\theta_k}{s_k^T u_k} u_k$ …. (2.15)

Yabe and Takano deriving a new concomitant condition according to Dai and Liao (Dai and Liao, 2001)**.** For this purpose, the modified secant condition (2.14) is used instead of the normal secant condition (2.13)

Let z_k be known as follows:-

$z_k = y_k + \rho\left(\frac{\theta_k}{s_k^T u_k} u_k\right)$ …. (2.16)

$\theta_k = 6(f_k - f_{k+1}) + 3(g_k + g_{k+1})^T s_k$

where ρ represents a constant non-negative quantity and $u_k \in R^n$ represents any vector that achieves $s_k^T u_k \neq 0$.

and z_k modified secant condition became as follows

$H_{k+1} z_k = s_k$ …. (2.17)

In the case $\rho = 0$ *and* $\rho = 1$, this condition corresponds to the normal secant condition (2.11) and the modified secant condition (2.13), respectively. Of (2.10) and (2.17) followed

$d_{k+1}^T z_k = -(H_{k+1} g_{k+1})^T z_k = -g_{k+1}^T (H_{k+1} z_k) = -g_{k+1}^T s_k$ …. (2.18)

By taking this relationship into the hypothesis, substitute the conjugation condition (2.17) with the new condition

$d_{k+1}^{T}z_k = -tg_{k+1}^{T}s_k$ ... (2.19)

If $t \geq 0$ represents a numerical quantity. To ensure that the direction of the search d_k fulfills this requirement by compensating (2.5) in (2.19) as follows

$-g_{k+1}^{T}z_k + \beta_{k+1}d_k^{T}z_k = -tg_{k+1}^{T}s_k$ ... (2.20)

To get β_{k+1} new as follows

$\beta_{k+1}^{YT} = \frac{g_{k+1}^{T}(z_k - ts_k)}{d_k^{T}z_k}$ ... (2.21)

According to Dai and Liao, Yabe and Takano modified the formula (2.21) as follows $\beta_{k+1}^{YT+} = Max\left\{\frac{g_{k+1}^{T}z_k}{d_k^{T}z_k}, 0\right\} - t\frac{g_{k+1}^{T}s_k}{d_k^{T}z_k}$ ... (2.22)

The length of the step α_k can be obtained by using any form of the search line. Both Yabe and Takano proved that the gradient method associated with formula (2.22) has an overall convergence, see the source **[3]**

2.3 Derive the new conjugated gradient formula

The main idea is to get the evolution of the method by deriving a new formula for conjugated gradient methods using the conjugated gradient method of Dai and Liao. The search directions given for Dai and Liao are as follows

$d_{k+1} = -g_{k+1} + \beta_k s_k$ ... (2.22)

Multiply the equation above by $\bar{y}_k^{T}$

$d_{k+1}^{T}\bar{y}_k = -g_{k+1}^{T}\bar{y}_k + \beta_k s_k^{T}\bar{y}_k = 0$ ... (2.23)

And using the conjugation condition $d_{k+1}^{T}\bar{y}_k = 0$

After simplification we get

$\beta_k s_k^{T}\bar{y}_k = g_{k+1}^{T}\bar{y}_k$ ... (2.24)

$\beta_k = \frac{g_{k+1}^{T}\bar{y}_k}{s_k^{T}\bar{y}_k}$ ... (2.25)

Substituting the value of $\bar{y}_k = \tau\theta_k y_k$, in equation (2.26) we get

That's where the value $\bar{y}_k$ came to impose

$\beta_k = \frac{g_{k+1}^{T}(\tau\theta_k y_k)}{s_k^{T}(\tau\theta_k y_k)}$, $\theta_1 = \frac{s_k^{T}s_k}{s_k^{T}y_k}$, $\theta_2 = \frac{s_k^{T}y_k}{y_k^{T}y_k}$

Substituting the $\theta_1$and θ_2 in the coefficient above, we get the new formula for the coefficient as follows

$\beta_k^{SYG1} = \frac{g_{k+1}^{T}(\tau\theta_1 y_k)}{s_k^{T}(\tau\theta_1 y_k)}$, $\tau = 0.0001$ ... (2.26)

$\beta_k^{SYG2} = \frac{g_{k+1}^{T}(\tau\theta_2 y_k)}{s_k^{T}(\tau\theta_2 y_k)}$, $\tau = 0.000001$ .... (2.27)

Thus, we get the final version of the search direction as follows

$d_{k+1} = -g_{k+1} + \beta_k^{SYG1}s_k$ .... (2.28)

$d_{k+1} = -g_{k+1} + \beta_k^{SYG2}s_k$ ... (2.29)

2.4- A New Algorithm

Step 1: Initialization Select $x_1 \in R^n$, $\varepsilon > 0$, set $g_1 = \nabla f_1$, $d_1 = -g_1$ *and* $k = 1$

Step 2: If $\|g_k\|_\infty \leq \varepsilon$ *then stop* x_k *is the optimal point , otherwise go to step*3

Step 3: Calculate the length of step $\alpha_k > 0$ satisfy the wolf condition (1.4) and (1.5)

Step 4: Calculate $x_{k+1} = x_k + \alpha_k d_k$ and calculate f_{k+1} and g_{k+1} ,

Step 5: Calculate the search direction $d_{k+1} = -g_{k+1} + \beta_k^{SYGn}s_k$, n = 1,2

and computeβ_k^{SYG1} , β_k^{SYG2} by use eq (2.26)and (2.27) and $\theta_1 = \frac{s_k^{T}s_k}{s_k^{T}y_k}$ *and* $\theta_2 = \frac{s_k^{T}y_k}{y_k^{T}y_k}$

Step 6: If the rate of convergence $|g_{k+1}g_k| > 0.2\|g_{k+1}\|^2$ satisfied then set

$d_{k+1} = -g_{k+1}$

Step 7: Calculate an initial value $\alpha_{k+1} = \alpha_k\left(\frac{\|d_k\|}{\|d_{k+1}\|}\right)$

Step 8: set k=k+1 and go to Step2

2.5 Some theoretical properties of the new algorithm

This section contains proof of some important properties of the proposed algorithm such as the regression property of this algorithm as well as the conjugation property

2.5.1 The Descent property of the new formula

We will mention the proof of the sufficient Descent Property of the proposed new formula for the conjugated gradient algorithm and that the sufficient gradient of the conjugated gradient algorithm is expressed as follows:

$g_{k+1}^{T}d_{k+1} \leq -c\|g_{k+1}\|^2$ ... (2.30)

On the other hand, the (Lipschize) condition $y_k \leq Ls_k$ achieves the following divergence

$s_k^{T}y_k \leq L\|s_k\|^2$ ... (2.31)

Theory (2.1) (New)

Let d_k the search direction for each $(k \geq 0)$ is generated by the formula (2.28) and (2.29). Suppose that the step size α_k meets the Wolfe standard condition (SDWC) (1.4) and (1.5) then d_k achieves sufficient descent property (2.31)

proof:-

Proof of mathematical induction

1- When k=1 then $d_1 = -g_1 \longrightarrow < 0$ $g_1^{T}d_1 < 0$

2- Suppose the relation (2.28) is true for all k.

3- We demonstrate the validity of the relationship (2.28) when k=k+1 By multiplying the ends of the relationship (2.28) by g_{k+1}^{T} we get

$d_{k+1}^{T}g_{k+1} = -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^{T}(\tau\theta_1 y_k)}{s_k^{T}(\tau\theta_1 y_k)}\right)s_k^{T}g_{k+1}$ ....(**)

Substituting for $\theta_1 = \frac{s_k^{T}s_k}{s_k^{T}y_k}$ in the equation above we get

$d_{k+1}^{T}g_{k+1} = -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^{T}\tau\left(\frac{s_k^{T}s_k}{s_k^{T}y_k}\right)y_k}{s_k^{T}\tau\left(\frac{s_k^{T}s_k}{s_k^{T}y_k}\right)y_k}\right)s_k^{T}g_{k+1}$ .. (2.32)

Using $y_k^{T}y_k \leq Ly_k^{T}s_k$ we get the following

$d_{k+1}^{T}g_{k+1} \leq -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^{T}\tau\left(\frac{s_k^{T}s_k}{Ly_k^{T}y_k}\right)y_k}{s_k^{T}\tau\left(\frac{s_k^{T}s_k}{Ly_k^{T}y_k}\right)y_k}\right)s_k^{T}g_{k+1}$ ... (2.33)

for substitution $\bar{y}_k = \tau\left(\frac{s_k^{T}s_k}{Ly_k^{T}y_k}\right)y_k$ In the equation above we get the following formula.

$d_{k+1}^{T}g_{k+1} \leq -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^{T}\bar{y}_k}{s_k^{T}\bar{y}_k}\right)s_k^{T}g_{k+1}$ ... (2.34)

Where $g_{k+1}^{T}\bar{y}_k \leq 0$ and $s_k^{T}\bar{y}_k > 0$, $s_k^{T}g_{k+1} > 0$ we get the following

$d_{k+1}^{T} g_{k+1} \leq 0$ ... (2.35)

Thus, the Descent property of the first proposed formula is demonstrated now we demonstrate the Descent property of θ_2 on the following equation

$d_{k+1}^{T} g_{k+1} =$

$-\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^{T}(\tau\theta_2 y_k)}{s_k^{T}(\tau\theta_2 y_k)}\right) s_k^{T} g_{k+1}$ .. (2.36)

Substituting for $\theta_2 = \frac{s_k^T y_k}{y_k^T y_k}$ in the equation above we get

$d_{k+1}^{T} g_{k+1} =$

$-\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^T \tau\left(\frac{s_k^T y_k}{y_k^T y_k}\right) y_k}{s_k^T \tau\left(\frac{s_k^T y_k}{y_k^T y_k}\right) y_k}\right) s_k^{T} g_{k+1}$ ... (2.37)

According to $y_k^{T} y_k \leq M s_k^T y_k$ we substitute in the equation above we get the following formulas

$d_{k+1}^{T} g_{k+1} \leq$

$-\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^T \tau\left(\frac{M y_k^T y_k}{y_k^T y_k}\right) y_k}{s_k^T \tau\left(\frac{M y_k^T y_k}{y_k^T y_k}\right) y_k}\right) s_k^{T} g_{k+1}$ ....(2.38)

Compensation for $\bar{y}_k = \tau\left(\frac{M y_k^T y_k}{y_k^T y_k}\right) y_k$ In the equation above we get the following formula

$d_{k+1}^{T} g_{k+1} \leq -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^T \bar{y}_k}{s_k^T \bar{y}_k}\right) s_k^{T} g_{k+1}$ ....(2.39)

Since $g_{k+1}^T \bar{y}_k \leq 0$, and $s_k^T \bar{y}_k > 0$, $s_k^T g_{k+1} > 0$ we get the ending formula as follows

$g_{k+1}^T \bar{y}_k \leq 0$ ....(2.40)

2.5.2 Analysis of the convergence property of the proposed algorithm

In this section we will demonstrate that the(CG) method with the direction of the research d_k converges absolutely to analyze the overall convergence in many iterative methods we need the following hypothesis

2.5.2Assumption (A1) [5]

We will impose the following hypotheses on the target function

1- Level Set $S = \{x \in R^n : f(x) \leq f(x_0)\}$ Closed and bonded at the primary point. That is, there is a constant $B > 0$ so that $\|x\| \leq B$, $\forall x \in S$

2- f is continuous and derivative in some of the vicinity of the level S and its gradients is Lipchitz continues , there exist $L > 0$ such that $\|g(x) - g(y)\| \leq L\|x - y\|$ $\forall x, y \in N$

3- f is uniformly convex function , then there exist a constant $\Gamma > 0$ such that

$(\nabla f(x) - \nabla f(y))^T (x - y) \geq \Gamma\|x - y\|^2$, *for any* $x, y \in S$

Or equivalently

$y_k^T s_k \geq \Gamma\|s_k\|^2$ *and* $\Gamma\|s_k\|^2 \leq y_k^T s_k \leq L\|s_k\|^2$

On the other hand , under assumption (A1) , it is clear that there exist positive constant B such that.

$\|x\| \leq B$ $\forall x \in S$ ... (2.41)

$\|\nabla f(x)\| \leq \bar{\varphi}$ $\forall x \in S$ ....(2.42)

Lemma (2.3) [5,6 and 7]

Suppose that assumption (2.3) and equation (2.41) hold true, and the sequence $\{x_k\}$ is generated from $x_{k+1} = x_k + \alpha_k d_k$. And d_k is Descent search direction and α_k the length of the step is obtained by Wolfe Line search .

If

$\sum_{k>1} \frac{1}{\|d_{k+1}\|^2} = \infty$ ... (2.43)

Then we get

$\lim_{k\to\infty}(\inf\|g_k\|) = 0$ ... (2.44)

More details can be found in**[2 , 8 and 15]**

Convex function uniformly, so by taking (2.5) we can prove the following theories

Theorem (2.5)

Suppose that assumption (2.3) and the sequence $\{x_k\}$ and the descent condition hold . conceder a conjugate gradient method in the form

$d_{k+1} = -g_{k+1} + \beta_k^{SYGn} s_k$, $n = 1,2$

Where α_k is computed from line search condition (1.4) and (1.5) , if the objective function is uniformly on a set S. then

$\lim_{k\to\infty}(\inf\|g_k\|) = 0$

Proof

Firstly , we need substituting our β_{k+1}^{SYG1} in the direction d_{k+1} there for we obtain :

$d_{k+1} = -g_{k+1} + \beta_k^{SYG1} s_k$ ....(2.45)

After simplify above equation we get

$\|d_{k+1}\|^2 = \left\|-g_{k+1} + \left(\frac{g_{k+1}^T \bar{y}_k}{s_k^T \bar{y}_k}\right) s_k\right\|^2$ ....(2.46)

$\|d_{k+1}\|^2 \leq \|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2 \|\bar{y}_k\|^2 \|s_k\|^2}{\|s_k\|^2 \|\bar{y}_k\|^2}$ ....(2.47)

$\|d_{k+1}\|^2 \leq \|g_{k+1}\|^2 \left(1 + \frac{\|\bar{y}_k\|^2 \|s_k\|^2}{\|s_k\|^2 \|\bar{y}_k\|^2}\right)$ ....(2.48)

Suppose that N= $\left(1 + \frac{\|\bar{y}_k\|^2 \|s_k\|^2}{\|s_k\|^2 \|\bar{y}_k\|^2}\right)$

$\|d_{k+1}\|^2 \leq N\bar{\varphi}^2$ ... (2.49)

$\|d_{k+1}\|^2 \leq \frac{1}{\bar{\varphi}^2}(N)^2$ ... (2.50)

$\|d_{k+1}\|^2 \leq \frac{1}{\bar{\varphi}^2} N$ ... (2.51)

$\sum_{k\geq 1} \frac{1}{\|d_{k+1}\|^2} \geq \frac{1}{N}\bar{\varphi}^2 \sum 1 = \infty$ ....(2.52)

And by using assumption (2.3) then

$\lim_{k\to\infty}(\inf\|g_k\|) = 0$

Now We will demonstrate the convergence property for $\theta_2 = \frac{s_K^T s_K}{y_k^T y_k}$

$\|d_{k+1}\|^2 = \left\|-g_{k+1} + \left(\frac{g_{k+1}^T \bar{y}_k}{y_k^T \bar{y}_k}\right) s_k\right\|^2$ ... (2.53)

$\|d_{k+1}\|^2 \leq \|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2 \|\bar{y}_k\|^2 \|s_k\|^2}{y_k^T \|\bar{y}_k\|^2}$ ... (2.54)

$\|d_{k+1}\|^2 \leq \|g_{k+1}\|^2 \left(1 + \frac{\|\bar{y}_k\|^2 \|s_k\|^2}{y_k^T \|\bar{y}_k\|^2}\right)$ ... (2.55)

Suppose that F=$\left(1 + \frac{\|\bar{y}_k\|^2 \|s_k\|^2}{y_k^T \|\bar{y}_k\|^2}\right)$

$\|d_{k+1}\|^2 \leq \bar{\varphi}^2 F$ ....(2.56)

$\|d_{k+1}\|^2 \leq \frac{1}{\bar{\varphi}^2} F$ ... (2.57)

$\|d_{k+1}\|^2 \leq \frac{1}{\bar{\varphi}^2} F$ ....(2.58)

$\sum_{k\geq 1} \frac{1}{\|d_{k+1}\|^2} \geq \frac{1}{F}\bar{\varphi}^2 \sum 1 = \infty$ ... (2.59)

And by using assumption (2.3) then

$\lim_{k\to\infty}(\inf\|g_k\|) = 0$ ... (2.60)

3.6 Numerical Results

In this section we will discuss the numerical results of the proposed new algorithm obtained from the use of the new formula for the $\beta_{K+1}^{SYG2}$and β_{k+1}^{SYG1} conjugation coefficients as well as the Wolfe (1.4) and (1.5) conditional set of test functions in the unconstrained optimization taken.(Andrei, 2008) **[9]** Always in the calculation of unconstrained optimization algorithms, we need the practical side because it is complementary to the theoretical side. To understand the power of the algorithm, we need to do it in practical terms and test various non-linear unconstrained problems, To evaluate the performance of these two proposed algorithms, 20 test functions have been selected which are included in this letter and are described in the Appendix..The functions are selected for dimensions n=100,…,1000, and by comparing the performance of these two new proposed algorithms with the DY, HS, FR algorithms, the measure used to stop the iterations of the algorithms is $\|g_k\|^2 \leq 10^{-6}$ The program is written in(FORTRAN 77) and translated using Visual (Fortran 6.6) in double-precision using a (standard-capacity Pentium-4 calculator), the algorithms in this thesis use the (in Exact line search) strategy (ILS). The test functions usually start with the standard starting point and the numerical results summary are recorded in Figures (2.1),(2.2) and 2.3) and by (Matlab R2009b) the program, The algorithm evaluation scale is compared based on the method of (Dolan and more) [3] to compare the efficiency of the proposed algorithms with the DY, HS, FR algorithms defined as the$p = 750$ set n_p of test functions and $S = 5$ the number of algorithms used. Let $l_{p,s}$ the number of times the value of the objective function be found by the algorithm S to solve a problem p.

$$r_{p,s} = \frac{l_{p,s}}{l_p^*} \quad \ldots.(3.61)$$

Where $l_p^* = \min\{l_{p,s} : s \in S\}$ Obviously $r_{p,s} \geq 1$ for all values p, s. If the algorithm fails to solve problems, the ratio $r_{p,s}$ is equal to the large number M .Efficiency of the algorithm S defines the counseling distribution of the efficiency rate $r_{p,s}$.

$$P_s(\tau) = \frac{size\{p \in P : r_{p,s} \leq \tau\}}{n_p} \ldots (3.62)$$

Obviously $p_s(1)$ represents the percentage of successful algorithm S preference Efficiency can also be used to analyze iterations, number of gradient values and processor time, In addition to get clear observations in the following chart horizontal coordinates and exponential scale **[8].**

Nots:-

1- Use conditional wollfe (1.4) &(1.5) to choose α_k

2- Choose $\tau = 0.0001$ $for\ \beta_k^{SYG1}$ and $\tau = 0.000001$ $for\ \beta_k^{SYG2}$

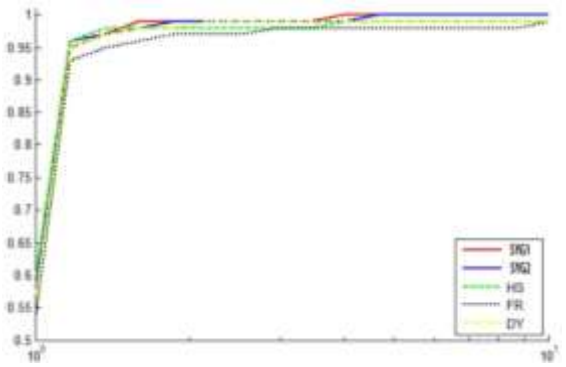

Figure (3-1) Comparison of algorithms in the number of iterations

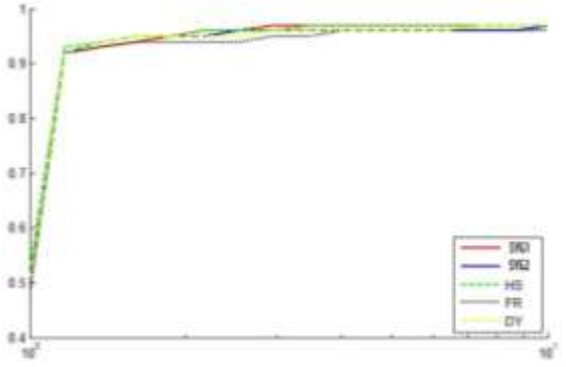

Figure (3-2) Comparison of algorithms in the number of times the function is calculated

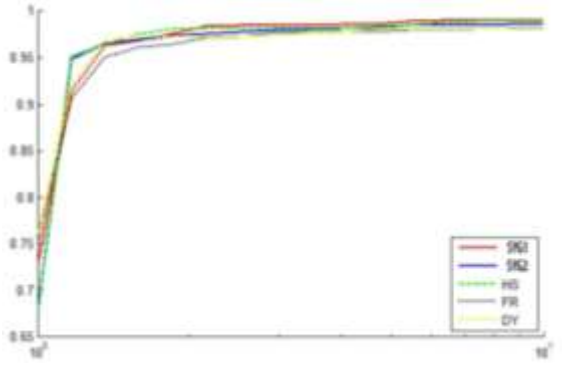

Figure (3-3) Comparison of algorithms in time

Conclusions

In this paper we propose two new Conjugate Gradient Methods based on(Dai – Liao) method. We study the characteristics of these two formulas from the scientific point of view and proved the properties of the descent and convergence by using some hypotheses. We also study the characteristics of the matrices and compared their performance with the methods of (HS, DY and FR) and gave us good results**.**

References

[1] Andrei, N., (2008b), "A Hybrid Conjugate Gradient Algorithm for Unconstrained Optimization as a Convex Combination of Hestenes Stiefel and Dai-Yuan". Studies in Informatics and Control, 17 (1), 55-70.

[2] Dai, Y. H. and Liao, L. Z., (2001), "New Conjugacy Conditions and Related Nonlinear Conjugate Gradient Methods". Applied Mathematics and Optimization, Springer-Verlag, New York, USA, 43, 87-101

[3] Yabe H. and Sakaiwa, N. (2003), A New Nonlinear Conjugate Gradient Method for Unconstrained Optimization, Technical Report, Department of Mathematical Information Science, Tokyo University of Science, March,

[4] Zoutendijk, G., (1970), "Nonlinear Programming, Computational Methods". Integer and Nonlinear Programming (J. bbadie ed.). North-Holland, 37-86.

[5] Sarin, R., Singh, B. K., & Rajan, S. (2014). "Study on Approximate Gradient Projection (AGP) Property in Nonlinear Programming". International Journal of Mathematics and Computer Applications Research (IJMCAR), 4(1), 19-30.

[6] Dai Y. H and Yuan Y, (1999). A nonlinear conjugate gradient method with a strong global convergence property, SIAM J.optimization, pp. 177-182

[7] Hestenes MR, Stiefel E (1952) Methods of conjugate gradients for solving linear systems. J Res Nat Bur Stand Sec B48:409–436..

[8] Dolan. E. D and Mor´e. J. J, "Benchmarking optimization software with performance profiles", Math. Programming, 91(2002), pp. 201-213

[9] Andrei N. (2008), "An Unconstrained Optimization test function collection". Adv. Model. Optimization. 10. pp.147-161

[10] Zhang, J., Deng, N. and Chen, L.(1999), "New Qusi-Newton equation and related method for unconstrained optimization ",J. Optim.Theory and Appl.(102)

[11] Namik, K. K. A. A. J. (2017). "Three-terms conjugate gradient algorithm based on the Dai-Liao and the Powell symmetric methods"

[12] Lylani, YA, (2016), "Developing new three-Term Conjugate Gradient Algorithms with Applications in Multilayer Networks, pp.1-10

[13] Fletcher R. and C.M. Reeves, (1964). "Function Minimization by Conjugate Gradients. Computer Journal", 7, PP. 149-154

[14] Polak E. and G. Ribiére, (1969). "Note Sur la Convergence de Directions Conjuguée. Revue Francaise Information", Recherche.Operationnelle, (16), pp.35-43

[15] Khudhur, H M, (2015), "Numerical and Analytical study of some Descent Algorithms to solve Unconstrained Optimization, pp. 33-43

خوارزمية جديدة في طريقة التدرج المترافق استناداً الى طريقة Dao-Liao للتدرج المترافق

شاهر قحطان حسين[1] ، غسان عزالدين عارف[1] ، يوكسل عبدالستار صادق[2]

[1]قسم الرياضيات ، كلية التربية للعلوم الصرفة ، جامعة تكريت ، تكريت ، العراق

[2]قسم الرياضيات ، كلية العلوم ، جامعة كركوك ، كركوك ، العراق

الملخص

في هذا البحث، تم اقتراح اتجاه بحث جديد، استنادا الى طريقة Dai-Liao للتدرج المترافق. لتصبح الخوارزمية المقترحة متقاربة مع بعض الفرضيات، وقد تم أيضًا إثبات خاصية الانحدار للطريقة الجديدة المقترحة، وأظهرت النتائج العددية أن الطريقة المقترحة فعالة مقارنة مع طرق (FR، HS و DY).