A new hybrid scaled search direction for unconstrained optimization

Runak M. Abdula [*] ***and Abbas Y. Al Bayati*** [**]
** College of Science-University of Sulaimani.*
***College of Computers science and Mathematics-University of Mosul*

Abstract

The best spectral CG-algorithm which is introduced by (Birgin &Martinez) and (Andrei. N) is modified in this paper by a hybrid search direction to overcome the lackness positive definiteness of the matrix defining the search direction.Two successive scalar parameters are introduced in this paper which are satisfy QN-like condition.These parameters are combined in such away to give a hybrid scaled search direction.The new proposed algorithm is still global convergent both theoretically and numerically.Computational results for (43) unconstrained test functions(Andri.N)show that the new algorithm substantially outperform the well- known(Andrei.N) scaled algorithm including the spectral (Birgin & Martinez) algorithm .

Introduction

Our problem is the following unconstrained optimization problem :

$$\min_{x \in R^n} \quad f(x) \qquad \ldots(1.1)$$

where a function $f : R^n \to R$ is smooth and it's gradient $g(x) = \nabla f(x)$ is available.Iterative methods are widely used for solving (1.1) and it's form is giving by

$$x_{k+1} = x_k + \alpha_k d_k \quad , \quad k = 0,1,... \qquad \ldots(1.2)$$

where $x_k \in R^n$ is the k-th approximation to the solution, $\alpha_k > 0$ is a step-size.and $d_k \in R^n$ is a search direction and satisfy the (Wolfe,1969,1971) conditions:

$$f(x_k + \alpha_k d_k) - f(x_k) \le \sigma_1 \alpha_k g_k^T d_k \qquad \ldots(1.3)$$

$$\nabla f(x_k + \alpha_k d_k)^T d_k \ge \sigma_2 g_k^T d_k \qquad \ldots(1.4)$$

where $0 < \sigma_1 \le \sigma_2 < 1$

There are many kinds of iterative method,the most effective iterative method for solving (1.1)are the Newton and Quasi-Newton methods because they have fast rate of convergence property.

However, they need matrices,this makes it difficult to apply these methods to a large scale problem,recently,the limited memory BFGS method is used to overcome this difficulty;variable –metric algorithm begin with an estimate x_1 to the minimiser $x_{\min}$ and a numerical estimate H_1 of the inverse Hessian matrix $G^{-1}(x)$ .A sequence of points x_k is then defined by :

$$x_{k+1} = x_k - \alpha_k H_k g_k$$

where α_k is a scalar chosen so as to reduce the value of f(x) at each iteration.The matrix H_k is updated by :

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \phi_k w_k w_k^T + \rho_k \frac{s_k s_k^T}{s_k^T y_k} \qquad \ldots(1.5)$$

with

$$\left.\begin{aligned} s_k &= x_{k+1} - x_k \text{ and } y_k = g_{k+1} - g_k \\ w_k &= (y_k^T H_k y_k) s_k - (s_k^T y_k) H_k y_k \end{aligned}\right\} \qquad \ldots(1.6)$$

where ϕ_k , ρ_k are scalars ;

The updating is perform so that :

$$H_{k+1} y_k = \rho_k s_k \qquad \ldots(1.7)$$

This condition is commonly satisfied with $\rho_k = \rho = 1, \forall k$ and is then called the Quasi-Newton (QN) condition; with this restriction of (1.2) we have so called (Broyden family).For a quadratic function, G^{-1} is constant and satisfies $s_k = G^{-1} y_k$ for any corresponding y_k and s_k ; Cleary the objective of such updating formula is that H_k tends (in some sense) to the inverse Hessian $G^{-1}(x_k)$. For a general function. It is well-known that if f is a quadratic and exact line search are carried out then after n iterations, $H_{k+1} = G^{-1}$.Perhaps,the strongest result concerning the convergence of the H-matrices towards G^{-1} for quadratic function is that of (Oren and Luenberger) .

Original Algorithm (Andrei, N.)

step(1): Let $x_0 \in R^n$, and the parameters $0 < \sigma_1 \le \sigma_2 < 1$. Compute $f(x_0)$ and $g_0 = \nabla f(x_0)$ .Set $d_0 = -g_0$ and $\alpha_0 = 1/\|g_0\|$. Set k=0.

Step(2): Compute α_k satisfy the wolfe conditions (1.3) and (1.4).Update the variables.Compute $f(x_{k+1})$, g_{k+1} and $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$.

Step(3): Test for the continuation of iterations. If this test is satisfied,then the iterations are stopped, else set k=k+1.

Step(4): Compute θ_k using a spectral $\theta_{k+1} = \frac{s_k^T s_k}{y_k^T s_k}$ or an anticipative

$$\theta_{k+1} = \frac{1}{\gamma_{k+1}}$$

Where γ_{k+1} is given by :

$$\gamma_{k+1} = \frac{2}{d_k^T d_k}\frac{1}{\alpha_k^2}[f(x_{k+1}) - f(x_k) - \alpha_k g_k^T d_k] \text{ or}$$

$$\gamma_{k+1} = \frac{2}{d_k^T d_k}\frac{1}{(\alpha_k - \eta_k)^2}[f(x_{k+1}) - f(x_k) - (\alpha_k - \eta_k) g_k^T d_k], \text{ where}$$

$$\eta_{k+1} = \frac{1}{g_k^T d_k}\frac{1}{\alpha_k^2}[f(x_k) - f(x_{k+1}) + \alpha_k g_k^T d_k + \delta], \text{ select a real } \delta > 0.$$

Step(5): Compute the search direction by :

$$d_{k+1} = -\theta_{k+1} g_{k+1} + \theta_{k+1}\left(\frac{g_{k+1}^T s_k}{y_k^T s_k}\right) y_k - \left[\left(1 + \theta_{k+1}\frac{y_k^T y_k}{y_k^T s_k}\right)\frac{g_{k+1}^T s_k}{y_k^T s_k} - \theta_{k+1}\frac{g_{k+1}^T y_k}{y_k^T s_k}\right] s_k \quad \ldots(2.1)$$

Step(6): Compute the initial guess of the step-length as :

$\alpha_k = \alpha_{k-1}\|d_{k-1}\|_2 / \|d_k\|_2$.

With this initialization compute α_k satisfying wolfe conditions (1.3) and (1.4).Update the variables $x_{k+1} = x_k + \alpha_k d_k$ .Compute $f(x_{k+1}), g_{k+1}$ and $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$.

Step(7): Store : $\theta = \theta_k$, $s = s_k$ and $y = y_k$.

Step(8): Test for the continuation of iterations.If this test is satisfied,then the iterations are stopped, else set k=k+1.

Step(9): Restart. If the powel restart criterion $\left|g_{k+1}^T g_k\right| \ge 0.2\|g_{k+1}\|^2$ or the angle restart criterion $d_k^T g_{k+1} > -10^{-3}\|d_k\|_2\|g_{k+1}\|_2$ are satisfied,then go to step(4); otherwise continue with step(10) .

Step(10): Compute:

$$v=\theta g_k-\theta\left(\frac{g_k^T s}{y^T s}\right)y\ +\left[\left(1+\theta\ \frac{y^T y}{y^T s}\right)\frac{g_k^T s}{y^T s}-\theta\ \frac{g_k^T y}{y^T s}\right]s$$

$$w=\theta y_k-\theta\left(\frac{y_{k-1}^T s}{y^T s}\right)y\ +\left[\left(1+\theta\ \frac{y^T y}{y^T s}\right)\frac{y_{k-1}^T s}{y^T s}-\theta\ \frac{y_{k-1}^T y}{y^T s}\right]s$$

and

$$d_k=-v+\frac{(g_k^T s_{k-1})w+(g_k^T w)s_{k-1}}{y_{k-1}^T s_{k-1}}-\left(1+\frac{y_{k-1}^T w}{y_{k-1}^T s_{k-1}}\right)\frac{g_k^T s_{k-1}}{y_{k-1}^T s_{k-1}}s_{k-1}$$

Step(11): Compute the initial guess of the step-length as :

$\alpha_k=\alpha_{k-1}\|d_{k-1}\|_2/\|d_k\|_2$.

With this initialization compute α_k satisfying wolfe conditions (1.3) and (1.4) . Update the variables $x_{k+1}=x_k+\alpha_k d_k$.Compute $f(x_{k+1})$, g_{k+1} and $s_k=x_{k+1}-x_k$, $y_k=g_{k+1}-g_k$.

Step(12): Test for the continuation of iterations.If this test is satisfied,then the iterations are stopped, else set k=k+1 and go to step(9).

The new hybrid parameters for the search direction

For solving unconstrained optimization problem (1.1) we can use an iterative process,initialized with x_0 and $d_0=-g_0$, $x_{k+1}=x_k+\alpha_k d_k$

$$d_{k+1}=-\theta_{k+1}g_{k+1}+\beta_k d_k \quad \ldots(3.1)$$

if $\theta=1$,then we get the classical conjugate gradient(CG) algorithms according to the value of β_k.On other hand if $\beta_k=0$ then we get another class of algorithms according to the selection θ_{k+1}.There are two possibilities for θ_{k+1}: a positive scalar or a positive definite matrix.If $\theta_{k+1}=1$we have steepest descent algorithm.If $\theta_{k+1}=G^{-1}$ or an approximation of it then we get the Newton or Quasi-Newton algorithm. Respectively , therefore we assume that $\theta_{k+1}\neq 0$ is selected in a Quasi-Newton manner and $\beta_k\neq 0$ then (2.1) represents a combination between (QN) and (CG). To determine θ_{k+1} consider the following procedure:Let $d_{k+1}=-H_{k+1}g_{k+1}$ $d_{k+1}=-H_{k+1}g_{k+1}$, where H_{k+1} is the inverse Hessian or an approximation of an inverse Hessian which satisfies Quasi-Newton condition.

and $d_{k+1} = -\theta_{k+1} g_{k+1} + \beta_k d_k$ let $-H_{k+1} g_{k+1} = -\theta_{k+1} g_{k+1} + \beta_k d_k$
multiply both sides by y_k ,we get $-y_K^t H_{k+1} g_{k+1} = -\theta_{k+1} y_k^T g_{k+1} + \alpha_k \beta_k y_k^T s_k$,
where ($H_{k+1} y_k = s_k$) $\rightarrow$ $-g_{k+1}^T s_k = -\theta_{k+1} y_k^T g_{k+1} + \alpha_k \beta_k y_k^T s_k$ or

$$\theta_{k+1} = \frac{g_{k+1}^T s_k}{y_k^T g_{k+1}} + \alpha_k \beta_k \frac{y_k^T s_k}{y_k^T g_{k+1}} \rightarrow \overline{\theta_{k+1}} = \frac{(g_{k+1} + \alpha_k \beta_k y_k)^T s_k}{y_k^T g_{k+1}} \qquad \ldots(3.2)$$

$$d_{k+1} = -\theta_{k+1} g_{k+1} + \beta_k d_k$$

For CG algorithm if $\beta_k = \dfrac{y_k^T g_{k+1}}{d_k^T y_k}$ then

$$d_{k+1} = -\theta_{k+1} g_{k+1} + \frac{g_{k+1}^T y_k}{s_k^T y_k} s_k$$

$$= -\theta_{k+1} g_{k+1} + \frac{g_{k+1}^T s_k}{s_k^T y_k} y_k$$

$\because d_{k+1}^T y_k = -s_k^T g_{k+1}$ (because ELS).

$\therefore -\theta_{k+1} g_{k+1}^T y_k + \dfrac{g_{k+1}^T s_k}{s_k^T y_k} y_k^T y_k = -s_k^T g_{k+1}$ or $\theta_{k+1} g_{k+1}^T y_k = (1 + \dfrac{y_k^T y_k}{s_k^T y_k}) s_k^T g_{k+1} \rightarrow$

$$\overset{*}{\theta}_{k+1} = \frac{(s_k^T y_k + y_k^T y_k) s_k^T g_{k+1}}{(s_k^T y_k)(y_k^T g_{k+1})} \qquad \ldots(3.3)$$

From the two new values of the parameters of the scaled parameters defined in (3.2) and (3.3), we are going to propose a new hybrid scaled parameter from the linear combination of the two parameters defined earlier I (3.2) and (3.3) as follows:

Outlines of the new proposed algorithm:

step(1): Let $x_0 \in R^n$; $d_0 = -g_0$; and k=0.

Step(2): Compute α_k satisfy the wolfe conditions (1.3) and (1.4).
Compute $f(x_{k+1})$, g_{k+1} , s_k and y_k.

Step(3): Test for the convergence . If $\|g_{k+1}\| < 1x10^{-5}$ stop , else continue.

Step(4): Compute the new scalar parameters using

$$\overline{\theta_{k+1}} = \frac{(g_{k+1} + \alpha_k \beta_k y_k) s_k}{y_k^T g_{k+1}} \quad \text{, from (3.2)}$$

$$\overset{*}{\theta}_{k+1} = \frac{(s_k^T y_k + y_k^T y_k) s_k^T g_{k+1}}{(s_k^T y_k)(y_k^T g_{k+1})} \quad \text{, from (3.3)}$$

set $\hat{\theta}_{k+1} = \lambda_{k+1}\bar{\theta}_{k+1} + (1-\lambda_{k+1})\theta^{*}_{k+1}$

Where λ_{k+1} is the optimal step size parameter computed from the line search procedure .

Step(5): Compute the new search direction by :

$$d_{k+1} = -\theta_{k+1} g_{k+1} + \beta_k d_k$$

Step(6): Compute α_k which satisfies (1.3) and (1.4) and defined by

$$\alpha_k = \alpha_{k-1}\|d_{k-1}\|_2 / \|d_k\|_2 \ .$$

Update the variables $x_{k+1} = x_k + \alpha_k d_k$. Compute $f(x_{k+1})$, g_{k+1} and $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$, $\hat{\theta}_{k+1}$

Step(7): Restart if $|g^T_{k+1} g_k| \geq 0.2\|g_{k+1}\|^2$ or $d^T_k g_{k+1} > -10^{-3}\|d_k\|_2 \|g_{k+1}\|_2$ are satisfied,then go to step(4) .

Step(8): Set k=k+1 and continue.

Some theoretical results:

Theorem(1):

Suppose that α_k in (1.2) satisfies the Wolfe conditions (1.3) and (1.4),then the direction d_{k+1} given by (2.1) is a descent direction.

Proof: since $d_0 = -g_0$, we have $g^T_0 d_0 = -\|g_0\|^2 \leq 0$, multiplying (2.1) by g^T_{k+1}, we have

$$g^T_{k+1} d_{k+1} = \frac{1}{(y^T_k s_k)^2}[-\theta_{k+1}\|g_{k+1}\|^2 (y^T_k s_k)^2 + 2\theta_{k+1}(g^T_{k+1} y_k)(g^T_{k+1} s_k)(y^T_k s_k) - (g^T_{k+1} s_k)^2 (y^T_k s_k) - \theta_{k+1}(y^T_k y_k)(g^T_{k+1} s_k)^2].$$

Applying the in equality $u^T v \leq \frac{1}{2}(\|u\|^2 + \|v\|^2)$ to the second term of the right hand side of the above equality,with $u = (s^T_k y_k) g_{k+1}$ and $v = (g^T_{k+1} s_k) y_k$ we get : $g^T_{k+1} d_{k+1} \leq \dfrac{(g^T_{k+1} s_k)^2}{y^T_k s_k}$

But, by Wolfe condition (1.4), $y^T_k s_k > 0$, therefore , $g^T_{k+1} d_{k+1} < 0$ for every k=0,1,… ,which completes the proof #

Theorem (2):

Assume that f is strongly convex .If at every step of the conjugate gradient (1.2) with d_{k+1} given by **(2.1) and** the step length α_k selected to

Satisfy the Wolfe conditions (1.3) and (1.4), then either $g_k = 0$ for some k, or $\lim_{k\to\infty} g_k = 0$.

Proof: Suppose $g_k \neq 0$ for all k. By strong convexity we have

$$y_k^T d_k = (g_{k+1} - g_k)^T d_k \geq \mu \alpha_k \|d_k\|^2. \qquad \ldots(3.4)$$

Since $g_k^T d_k < 0$

By theorem (1), therefore , the assumption $g_k \neq 0$ implies $d_k \neq 0$. Since $\alpha_k > 0$, from (3.4) it follows that $y_k^T d_k > 0$. But f is strongly convex , therefore f is bounded from below. Now, summing over k the first Wolfe condition (1.3) we have $\sum_{k=0}^{\infty} \alpha_k g_k^T d_k > -\infty$. Considering the lower bound for α_k given in $\alpha_k \geq \frac{1-\sigma_2 |g_k^T d_k|}{L\|d_k\|^2}$, $\sigma_2 < 1$ and having in view that d_k is a descent direction, it follow that

$$\sum_{k=0}^{\infty} \frac{|g_k^T d_k|^2}{\|d_k\|^2} < \infty \qquad \ldots(3.5)$$

from $g_{k+1}^T d_{k+1} \leq -\frac{(g_{k+1}^T s_k)^2}{y_k^T s_k}$,using the inequality of Cauchy and $y_k^T s_k \geq \mu \|s_k\|^2, \mu > 0$ we get $g_{k+1}^T d_{k+1} \leq -\frac{(g_{k+1}^T s_k)^2}{y_k^T s_k} \leq -\frac{\|g_{k+1}\|^2 \|s_k\|^2}{\mu \|s_k\|^2} = -\frac{\|g_{k+1}\|^2}{\mu}$.

Therefore ,from (3.5) it follows that :

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^2}{\|d_k\|^2} < \infty \qquad \ldots(3.6)$$

inserting in (3.6) the upper bound of d_k given by : $\|d_{k+1}\| \leq \left(\frac{2}{\mu} + \frac{2L}{\mu^2} + \frac{L^2}{\mu^3}\right)\|g_{k+1}\|$ or $\|d_{k+1}\| \leq \left(\frac{1}{m} + \frac{2L}{m\mu} + \frac{L^2}{m\mu^2}\right)\|g_{k+1}\|$, $m > 0$ we get

$$\sum_{k=0}^{\infty} \|g_k\|^2 < \infty$$

which completes the proof #

Numerical results

The comparative test involves(43) well-known standard test functions (given in the Appendix) with different dimensions. The results are given in the Table (1) is specifically quoting the number of function evaluations (NOF) and the number of gradient evaluations (NOG) . All programs are written in FORTRAN 90 language and for all cases the stopping criterion is taken to be $\|g_{k+1}\| < 1x10^{-5}$. The results are given in table (1): this table shows also that there are several test functions which are not working by the original algorithm. From table (2) we conclude that the new proposed algorithm has an improvement against the original algorithm in about (%25)NOI(number of iterations) and (%37) (NOF+NOG) according to our numerical results done in this work.

Table 1: Comparison between(New Algorithm and Original algorithm)

Test Function	**N**	**Original algorithm**		**New algorithm**	
		NOI	**NOF+NOG**	**NOI**	**NOF+NOG**
Extended Trigonometric	1000	69	98	47	73
	5000	30	54	29	56
	9000	38	66	36	61
Extended white & Holst	1000	32	55	30	55
	5000	32	60	29	55
	9000	32	58	33	56
Extended Beale	3000	14	24	11	21
	5000	11	20	10	19
	7000	11	20	10	18
	9000	11	20	9	17
Raydan2	1000	4	9	4	9
	7000	4	9	4	9
	9000	4	9		
Diagonal2	1000	246	372	235	371
	9000	895	1326	752	1180
Diagonal3	1000	OVERFLOW	OVERFLOW	3001	25419
Hager	1000	OVERFLOW	OVERFLOW	326	9148
Generalized Tridiagonal-1	1000	26	49	26	49
	7000	42	534	32	315
Extended Tridiagonal-1	1000	9	18	9	15
	5000	12	21	9	17
	9000	8	16	9	15
Diagonal4	1000	4	8	4	8
	5000	4	8	4	8
	9000	4	8	4	8
Extended PSC1	1000	24	176	21	36
	5000	39	554	24	43
	7000	42	812	26	47
	9000	35	384	28	144
Extended Powel	1000	41	97	45	96
	3000	51	102	44	93
	5000	49	107	44	99
	9000	48	97	45	105
Full Hessian FH1	1000	4	10	19	22
	5000	6	12	10	12
	7000	6	11	11	14
	9000	OVERFLOW	OVERFLOW	11	14
Full Hessian FH2	1000	OVERFLOW	OVERFLOW	550	1066
	5000	OVERFLOW	OVERFLOW	1445	2624
	9000	OVERFLOW	OVERFLOW	1869	3450
Extended Marators	1000	56	125	55	117
	3000	50	101	56	109
	9000	53	107	53	107
Total		2046	5557	1837	3479

Table 2: Percentage Performance of the new proposed algorithm against the original algorithm

Tools	Original algorithm	New algorithm
NOI	%100	%75
NOF+NOG	%100	%63

References

- Andrei. N.,(2005): A Scaled Nonlinear Conjugate Gradient Algorithm for Unconstrained Optimization, Research Institute for Informatics. Bucharest.
- Andrei. N.,(2004): Unconstrained Optimization-Test Function Research Institute for Informatics. Bucharest.
- Birgin. E and Martinez. M.,(2001): a Spectral Conjugate Gradient Method for Unconstrained Optimization Applied Math. And Optimization,Vol. 43,.
- Broyden, C.G., (1970): The Convergence of a Class of Double Rank Minimization Algorithms II. The New Algorithm", Journal of The Institute of Mathematics And it's Applications, Vol. 6,pp. 221-231.
- Oren, S.S. and Luenberger, D.G.,(1974): (SSVM) Algorithm, Part I, Criteria and Sufficient Conditions for scaling a class of Algorithms Management Science, Vol. 20,pp.845-862,.
- Wolfe,P.,(1969) :Convergence condition for search methods, SIAM Rev., Vol. 11, PP.226-235,.
- Wolfe,P.,(1971): Convergence condition for ascent methods II: Some Corrections,SIAM Rev.,Vol.13, PP.185-188,.

Appendix

All the test functions used in this paper are from general literature:

1.Extended Trigonometric Function

$$f(x)=\sum_{i=1}^{n/2}(-13+x_{2i-1}+((5-x_{2i})x_{2i}-2)x_{2i})^2+(-29+x_{2i-1}+((x_{2i}+1)-14)x_{2i})^2,$$

$$x_0=[0.5,-2,0.5,-2,...,0.5,-2]$$

2.Extended White & Holst Function

$$f(x)=\sum_{i=1}^{n}\left[\left[n-\sum_{j=1}^{n}\cos x_j\right]+i(1-\cos x_i)-\sin x_i\right]^2 \quad , \quad x_0=[0.2,0.2,...,0.2]$$

3.Extended Beale Function

$$f(x)=\sum_{i=1}^{n/2}(1.5-x_{2i-1}(1-x_{2i}))^2+(2.25-x_{2i-1}(1-x_{2i}^2))^2+(2.625-x_{2i-1}(1-x_{2i}^3))^2 \quad ,$$

$$x_0=[1,0.8,...,1,0.8]$$

4. Raydan2 Function

$$f(x)=\sum_{i=1}^{n}(\exp(x_i)-x_i) \quad , \quad x_0=[1,1,...,1]$$

5. Diagonal2 Function

$$f(x)=\sum_{i=1}^{n}(\exp(x_i)-\frac{x_i}{i}) \quad , \quad x_0=[1/1,1/2,...,1/n]$$

6. Diagonal3 Function

$$f(x)=\sum_{i=1}^{n}(\exp(x_i)-i\sin(x_i) \quad , \quad x_0=[1,1,...,1]$$

7.Hager Function

$$f(x)=\sum_{i=1}^{n}(\exp(x_i)-\sqrt{i}x_i) \quad , \quad x_0=[1,1,...,1]$$

8. Generalized Tridiagonal -1 Function

$$f(x)=\sum_{i=1}^{n-1}(x_i+x_{i+1}-3)^2+(x_i-x_{i+1}+1)^4 \quad , \quad x_0=[2,2,...,2]$$

9. Extended Tridiagonal -1 Function

$$f(x)=\sum_{i=1}^{n/2}(x_{i-2}+2x_i-3)^2+(x_{2i-1}-x_{2i}+1)^4 \quad , \qquad x_0=[2,2,...,2]$$

10. Diagonal4 Function

$$f(x)=\sum_{i=1}^{n/2}\frac{1}{2}(x_{2i-1}^2+cx_{2i}^2) \quad , \qquad x_0=[1,1,...,1]$$

11. Generalized PC1 Function

$$f(x)=\sum_{i=1}^{n-1}(x_{2i-1}^2+x_{2i}^2+x_i x_{i+1})^2+\sin^2(x_{2i-1})+\cos^2(x_{2i}) \quad , \qquad x_0=[3,0.1,...,3,0.1]$$

12. Extended Powell Function

$$f(x)=\sum_{i=1}^{n/4}(x_{4i-3}+10x_{4i-2})^2+5(x_{4i-1}-x_{4i})^2+(x_{4i-2}-2x_{4i-1})^4+10(x_{4i-3}-x_{4i})^4 \quad ,$$

$$x_0=[3,-1,0,1,...,3,-1,0,1]$$

13. Full Hessian FH1 Function

$$f(x)=(x_1-3)^2+\sum_{i=2}^{n}(x_1-3-2(x_1+x_2+...+x_i)^2)^2 \quad , \quad x_0=[0.1,0.1,...,0.1]$$

14. Full Hessian FH2 Function

$$f(x)=(x_1-5)^2+\sum_{i=2}^{n}\ (x_1+x_2+...+x_{i-1})^2 \quad , \qquad x_0=[0.1,0.1,...,0.1]$$

15. Extended Maratos Function

$$f(x)=\sum_{i=1}^{n/2}x_{2i-1}+c(x_{2i-1}^2+x_{2i}^2-1)^2 \quad , \qquad x_0=[1.1,0.1,,...,1.1,0.1]$$

اتجاه بحث هجيني جديد في الامثلية غير المقيدة

روناك محمد عبدالله*   و   عباس يونس الياس*
***كلية العلوم – جامعة السليمانية**
***كلية علوم الحاسبات والرياضيات – جامعة الموصل**

الخلاصة

الخوارزمية المثلى للتدرج المترافق الطيفية المستخدمة من قبل (Birgin & Martinez) و (Andrei) تم تطويرها في هذا البحث للتغلب على ظاهرة كون المصفوفة المستخدمة تكون احياناً غير موجبة التعريف. تم التعرف على معلمتين جديدتين تحققان ما يسمى بـــ الشرط الشبيه بشرط نيوتن بشكل متداخل و هجيني لتكون اتجاه بحث جديد. الخوارزمية الجديدة لازالت تحقق التقارب الشامل نظرياً وعملياً . النتائج الحسابية لــــ (43) دالة غير مقيدة(Andrei) اثبت تفوق الخوارزمية المقترحة على خوارزمية (Andrei) التي تحتوي ضمنياً على الخوارزمية الطيفية لــــ (Birgin & Martinez).