Modified shifted variable metric algorithms for solving unconstrained minimization problems

Prof Dr. Abbas Y. AL-Bayati **Basim A. Hassan**
Department of mathematics,
College of Computers Sciences and Mathematics
University of Mosul

Received 26 / 02 / 2009 **Accepted** 05 / 05 / 2009

الملخص

في هذا البحث تم اقتراح خوارزميات متطورة لخوارزمية المتري المتغير المرفوع عندما تستخدم المصفوفة $H_k = \xi_k I + A_k$, عندما $k \geq 1$, $\xi_k \succ 0$ والمصفوفة A_k هي مصفوفة متماثلة وموجبة التعريف وتستخدم شرط جديد لشرط نيوتن معرفاً $A_{k+1}y_k^* = \rho_k^* \tilde{v}_k$ عندما $\tilde{v}_k = v_k - \xi_k y_k^*$, $y_k^* = y_k + m_k v_k$, $\rho_k^* = y_k^{*T} A_k y_k^* / v_k^T y_k^*$ وتم التحقق من اشتقاق الخوارزميات المطورة.

النتائج العددية اثبت كفاءة الخوارزميات المطورة بالاعتماد عن المقارنة مع خوارزميات BFGS القياسية و BFGS المرفوعة.

Abstract

In this paper, we propose a modified version of the shifted VM algorithms where the matrices H_k have the form $H_k = \xi_k I + A_k$, $k \geq 1$ where $\xi_k \succ 0$ and A_k are symmetric positive semi definite matrices, usually $A_1 = 0$, A_{k+1} is obtained from A_k to satisfy the shifted modified quasi-Newton condition ,we consider our new QN-condition the form $A_{k+1}y_k^* = \rho_k^* \tilde{v}_k$, $\tilde{v}_k = v_k - \xi_k y_k^*$, $y_k^* = y_k + m_k v_k$ where $\rho_k^* = y_k^{*T} A_k y_k^* / v_k^T y_k^*$, and we derive of the modified algorithms.

Experimental results indicate that the new proposed algorithms were efficient than the both standard BFGS & the shifted BFGS-algorithms.

1. Introduction :

Consider the unconstrained optimization problem :

$$\min\left\{f(x)\ \middle|\ x\in R^n\right\} \qquad \text{..........(1)}$$

where f is a continuously differentiable function of n variables. Quasi-Newton methods for solving (1) often need the new search direction d_{k+1} at each iteration defined by:

$$d_k = -H_k g_k \qquad \text{..........(2)}$$

where $g_{k+1} = \nabla f(x_{k+1})$ is the gradient of f evaluated at the current iterate x_{k+1}. **[8]** One then computes the next iterate by

$$x_{k+1} = x_k + \alpha_k d_k \qquad \text{..........(3)}$$

where the step size α_k satisfies the Wolfe – Powell (WP) conditions

$$f(x_k + \alpha_k d_k) \le f(x_k) + \delta_1 \alpha_k d_k^T g_k \qquad \text{..........(4)}$$

$$g(x_k + \alpha_k d_k)^T d_k \ge \delta_2 d_k^T g_k \qquad \text{..........(5)}$$

where $\delta_1 \prec 1/2$ and $\delta_1 \prec \delta_2 \prec 1$, and H_{k+1} is an approximation to $\{\nabla^2 f(x_k)\}^{-1}$. The matrix H_{k+1} satisfies standard the quasi-Newton condition

$$H_{k+1} y_k = v_k \qquad \text{..........(6)}$$

where $y_k = g_{k+1} - g_k$, $v_k = x_{k+1} - x_k$, and $B_k = H_k^{-1}$ see **[2]**. The standard Broyden-Fletcher-Goldfarb-Shanno (BFGS) update can be separated into two components , $H^{(1)}$ *and* $H^{(2)}$ so that:

$$H_{BFSG} = H^{(1)} + H^{(2)} \qquad \text{..........(7)}$$

where

$$H^{(1)} = H_k - \frac{H_k y_k v_k^T + v_k y_k^T H_k}{y_k^T H_k y_k} + \frac{v_k v_k^T}{v_k^T y_k}\left[\frac{y_k^T H_k y_k}{v_k^T y_k}\right] \quad , \quad H^{(2)} = \frac{v_k v_k^T}{v_k^T y_k}$$

In **[4]** the BFGS update which can be written in the product form:

$$H_{k+1} = (I - \vartheta_k v_k y_k^T) H_k (I - \vartheta_k y_k v_k^T) + \vartheta_k v_k v_k^T \qquad \text{..........(8)}$$

where

$$\vartheta_k = \frac{1}{v_k^T y_k} \qquad \text{..........(9)}$$

AL- Bayati's **[1]** self-scaling VM-update, can be expressed in the product form as follows

$$H_{k+1} = (I - \vartheta_k v_k y_k^T) H_k (I - \vartheta_k y_k v_k^T) + \rho_k \vartheta_k v_k v_k^T \qquad \text{..........(10)}$$

where

$$\vartheta_k = \frac{1}{v_k^T y_k} \quad \& \quad \rho_k = \frac{y_k^T H_k y_k}{v_k^T y_k} \qquad \text{..........(11)}$$

In **[5]** Li & Fukushima proposed a new Quasi –Newton equation defined by:

$$H_{k+1} y_k^* = v_k \qquad ....(12)$$

where

$$y_k^* = y_k + m_k v_k \qquad ....(13)$$

where m_k is a positive constant. They choose m_k to be a very small number, for example $m \le 10^{-6}$.

2. Amodified Shifted variable metric algorithms :

Variable metric algorithms, use symmetric positive defined matrices H_k or $B_k = H_k^{-1}$, $k \ge 1$,usually $H_1 = I$ and H_{k+1} is obtained from H_k by a rank–two VM update to satisfy the quasi–Newton condition $H_{k+1} y_k^* = \rho_k^* v_k$, where $y_k^* = y_k + m_k v_k$, ρ^* is any scalar.

In shifted BFGS algorithm, matrices H_k have the form:

$$H_k = \xi_k I + A_k \quad , k \ge 1 \qquad ....(14)$$

where $\xi_k \succ 0$ and A_k are symmetric positive semi definite matrices, usually $A_1 = 0$, A_{k+1} and is obtained from A_k by a rank-two VM update to satisfy the shifted modified quasi-Newton condition, we consider it usually in the form

$$A_{k+1} y_k^* = \rho_k^* \tilde{v}_k \;,\; \tilde{v}_k = v_k - \xi_k y_k^* \;,\; y_k^* = y_k + m_k v_k \qquad ....(15)$$

where $\rho_k^* \succ 0$ is a non quadratic correction parameter.**[6]**

Relation (14) & (15) imply that matrix H_{k+1} satisfies the quasi-Newton condition $H_{k+1} y_k^* = \rho_k^* v_k$. To simplify the notation we often omit index k and replace index $k+1$ by symbols +. In the subsequent analysis we use the following notation:

$$a = y^{*T} H y^* \;;\; \bar{a} = y^{*T} A y^* \;;\; \hat{a} = y^{*T} y^* \;;\; b = v^T y^*$$
$$\bar{b} = v^T B A y^* \;;\; \tilde{b} = \tilde{v}_k^{\,T} y^* \;;\; \bar{c} = v^T BABv. \qquad .....(16)$$

Involving the scaling and the non quadratic correction and using the same argumentation as in standard VM methods, we can write the new modified-update for $\tilde{b} \succ 0$ (which implies $\tilde{v} \ne 0$, $y^* \ne 0$) in the form

$$A_+^{\mathrm{modified}\,(1)} = A + \left(\rho + \frac{\bar{a}}{\tilde{b}}\right) \frac{\tilde{v}\tilde{v}^T}{\tilde{b}} - \frac{\tilde{v} y^{*T} A + A y^* \tilde{v}^T}{\tilde{b}} \quad where \;\; \rho = \frac{\bar{a}^*}{\tilde{b}^*} \qquad ....(17)$$

The shifted modified AL-Bayati-update which can be written in the product form:

$$A_+^{\text{modified (1)}} = (I - \vartheta^* \tilde{v} y^{*T}) A (I - \vartheta^* y^* \tilde{v}^T) + \rho^* \vartheta^* \tilde{v}\tilde{v}^T \qquad \text{......(18 } a)$$

where

$$\vartheta^* = \frac{1}{\tilde{v}^T y^*} \quad , \quad \rho^* = \vartheta^* y^{*T} A y^* \qquad \text{..........(18 } b)$$

We replace y^* by y. Then update (18) will be replaced by the following modified (2) algorithm

$$A_+^{\text{modified (2)}} = (I - \vartheta \tilde{v} y^T) A (I - \vartheta y \tilde{v}^T) + \rho \vartheta \tilde{v} \tilde{v}^T \qquad \text{......(19 } a)$$

where

$$\vartheta = \frac{1}{\tilde{v}^T y} \quad , \quad \rho = \vartheta y^T A y \qquad \text{..........(19 } b)$$

In **[6]** the parameter ξ_+ should satisfy inequality $0 \prec \xi_+ \prec b/\hat{a}$. Therefore, it is advantageous introduce relative shift parameter $\theta = \xi_k \frac{\hat{a}}{b} \in (0,1)$ and by (15) we can write

$$\xi_+ = \theta \frac{b}{\hat{a}} \quad , \quad \tilde{b} = \tilde{v}^T y = b - \xi_+ a = b(1-\theta) \qquad \text{..........(20)}$$

$$\mu = \min\left\{ \max\left(\sqrt{1 - \bar{a}/a} \Big/ \left(1 + \sqrt{1 - b^2 / \hat{a}\|s\|^2}\right), 0.2 \right), 0.8 \right\} \qquad \text{..........(21)}$$

3. Derivetion of the modified Shifted algorithms :

Standard VM methods can be obtained by solving a certain variational problem-we find an update of VM matrix see[6]. Using the product form of the update, we can extend this approach to VM- methods. First we give the following theorem, where the shifted modified quasi-Newton condition $U_+ U_+^T y^* = A_+ y^* = \rho^* \tilde{v}$ is equivalently replaced by

$$U_+^T y^* = z, \quad U_+ Z = \rho^* \tilde{v}, \quad z^T z = \rho^* \tilde{b} \qquad \text{..........(22)}$$

where U_k, $k \geq 1$,is a rectangular metric and $\rho_k^* = y_k^{*T} A_k y_k^* / v_k^T y_k^*$.

Theorem (3.1) : Let T be a symmetric positive definite matrix, $z \in R^n$ and denote Y the set of $N * m$ matrices. Then the unique solution to

$$\min\{\phi(U_+) : U_+ \in Y\} \;\; S.t.: (22), \phi(U_+) = y^{*T} T y^* \left\| T^{-1/2} (U_+ - U) \right\|_F^2 \qquad \text{..........(23)}$$

(Frobenius matrix norm) is

$$U_+ = U - \frac{Ty^*}{y^{*T}Ty^*}y^{*T}U + \left(\rho^*\tilde{v} - Uz + \frac{y^{*T}Uz}{y^{*T}Ty^*}Ty^*\right)\frac{z^T}{z^Tz} \qquad \text{..........(24)}$$

and for this solution the value of $\phi(U_+)$ is

$$\phi(U_+) = \left|U^Ty^* - z\right|^2 + \frac{y^{*T}Ty^*}{z^Tz}\upsilon^TT^{-1}\upsilon,\ \upsilon = \rho^*\tilde{v} - Uz + \frac{y^{*T}Uz}{y^{*T}Ty^*}Ty^* \qquad \text{....(25)}$$

Proof : Setting $U_+ = (u_1^+, \ldots\ldots\ldots\ldots, u_m^+)$, define Lagrange function $L = L(U_+, e_1, e_2)$ as

$$L = \frac{1}{2}\phi(U_+) + e_1^T(U_+^Ty^* - z) + e_2^T(U_+z - \rho^*\tilde{v}) \qquad \text{..........(26)}$$

$$= -e_1^Tz - \rho^*e_2^T\tilde{v} + \sum_{k=1}^{m}\left[\frac{y^{*T}Ty^*}{2}(u_k^+ - u_k)^{-1}T^{-1}(u_k^+ - u_k) + e_1^Ty^Tu_k^+ + ze_2^Tu_k^+\right] \qquad \text{....(27)}$$

a local minimizer U_+ satisfied the equation $\partial L / \partial u_k^+ = 0, k = 1, \ldots\ldots\ldots, m$, which gives $y^{*T}Ty^*T^{-1}(u_k^+ - u_k) + e_1y + ze_2 = 0, k = 1, \ldots\ldots, m$, and yielding

$$U_+ = U - \frac{Ty^*}{y^{*T}Ty^*}e_1^T - \frac{Te_2}{y^{*T}Ty^*}z^T. \qquad \text{..........(28)}$$

Using the first condition in (22), we have $e_1 = U^Ty^* - (1 + y^{*T}Te_2 / y^{*T}Ty^*)z$. Substituting this e_1 to (28), we obtain $U_+ = U - Ty^*y^{*T}U / y^{*T}Ty^* + \bar{e}z^T$ with some vector $\bar{e}$. The second condition in (22) yields

$$\bar{e} = \frac{1}{z^Tz}\left(\rho^*\tilde{v} - Uz - \frac{z^Tz - y^{*T}Uz}{y^{*T}Ty^*}Ty^*\right) = \frac{\upsilon}{z^Tz} \qquad \text{..........(29)}$$

and by using (24) it follows. Matrix U_+ obtained in this way minimizes ϕ in view of convexity of Frobenius norm.. Furthemore, we get

$$\bar{e} - \frac{Ty^*}{y^{*T}Ty^*} = \frac{1}{z^Tz}\left(\rho^*\tilde{v} - Uz - \frac{z^Tz - y^{*T}Uz}{y^{*T}Ty^*}Ty^*\right) = \frac{\upsilon}{z^Tz} \qquad \text{..........(30)}$$

by (22) & (25), thus by (24) and $\upsilon^Ty = 0$

$$\frac{\phi(U_+)}{y^{*T}Ty^*} = \left\|T^{-1/2}\left(\frac{Ty^*}{y^{*T}Ty^*}y^{*T}U - \bar{e}z^T\right)\right\|_F^2$$

$$= \left\|T^{-1/2}\left(\frac{Ty^*}{y^{*T}Ty^*}(U^Ty^* - z)^T - \frac{\upsilon}{z^Tz}z^T\right)\right\|_F^2$$

$$= Tr\left(\frac{(U^Ty^* - z)(U^Ty^* - z)^T}{y^{*T}Ty^*} + \frac{\upsilon^TT^{-1}\upsilon}{(z^Tz)^2}zz^T\right)$$

$$\frac{\phi(U_+)}{y^{*T}Ty^*} = \frac{\left|U^Ty^* - z\right|^2}{y^{*T}Ty^*} + \frac{\upsilon^TT^{-1}\upsilon}{z^Tz}.$$

The advantage of variationally- derived update formula (24) consists in possibility of parameters choice. To use Theorem (3.1) for the shifted AL-Bayati update, we replace y^* by y. Then update (24) will be replaced by

$$U_+ = U - \frac{Ty}{y^T Ty} y^T U + \left(\rho \tilde{v} - Uz + \frac{y^T Uz}{y^T Ty} Ty \right) \frac{z^T}{z^T z} \qquad ....(31)$$

Thus the general form of the modified algorithm (24) can be rewritten, using (22):

$$U_+ = \frac{\tilde{v} z^T}{\tilde{b}} + \left(I - \frac{Ty^* y^{*T}}{y^{*T} Ty^*} \right) U \left(I - \frac{zz^T}{z^T z} \right). \qquad ....(32)$$

Since $z^T (I - zz^T / z^T z) = 0$ and $(I - zz^T / z^T z)^2 = I - zz^T / z^T z$, this yields

$$A_+ = \rho^* \frac{\tilde{v} \tilde{v}^T}{\tilde{b}} + \left(I - \frac{Ty^* y^{*T}}{y^{*T} Ty^*} \right) U \left(I - \frac{zz^T}{z^T z} \right) U^T \left(I - \frac{y^* y^{*T} T}{y^{*T} Ty^*} \right) \qquad ....(33)$$

by $A_+ = U_+ U_+^T$. Usually $Ty^* = \tilde{v}$ & $\rho_k^* = y_k^{*T} A_k y_k^* / v_k^T y_k^*$. This choice gives the modified algorithms if $z = 0$.

$$A_+ = \rho^* \frac{\tilde{v} \tilde{v}^T}{\tilde{b}} + \left(1 - \frac{\tilde{v} y^{*T}}{\tilde{b}} \right) A \left(1 - \frac{y^* \tilde{v}^T}{\tilde{b}} \right). \qquad ....(34)$$

Theorem (3.2) :

Let A be positive semi definite, and $\xi_+ \hat{a} \prec b$. Then matrix A_+ given by (18) is positive semi definite.

Proof: Since $\tilde{b} = \tilde{v}^T$, $y = b - \xi_+ \hat{a} \succ 0$,It is evident form (18) that $A_+ = Z^T Z + L$ where $L = \vartheta_k \tilde{v}_k \tilde{v}_k^T$ and $Z = A_k^{1/2} (I - \vartheta_k \tilde{v}_k y_k^T)$. Since both $Z^T Z$ & L are positive semi definite, so is A_+, where $\vartheta_k = 1/\tilde{b}$.

3.2 Outlines of the modified (1) and (2) Shifted Algorithms :

The outline of the algorithm as follows:

Step 0 : Choose an initial point $x_1 \in R^n, \varepsilon = 10^{-6}$, set $k = 1$.
Step 1 : If $\|g_k\| \le \varepsilon$, stop .
Step 2 : Compute the search direction $d_k = -A_k g_k$.
Step 3 : Find α_k by (WP) step size rule (4) and (5) .
Step 4 : Generate a new iterated point by $x_{k+1} = x_k + \alpha_k d_k$ and calculate the updating formula , where A_k as given in (18) or (19).
Step 5 : Set $k = k + 1$ and go to Step 12 .

4. Numerical Results :

In this section, we have been reported the numerical results for the new formulae (18) & (19). We tested, using the collection of problems for general sparse and separable unconstrained optimization test problems. From **[7]**, we have used the dimension of the problem (N), N=10,100,1000,10000,100000. For each problem, we choose the initial matrix $A_{k+1} = \rho_k \frac{v_k v_k^T}{v_k^T y_k}$. These algorithms use the cubic fitting technique which satisfy Wolfe-Powell (WP) condition as a line search technique in which $\delta_1 = 0.0001 \quad \delta_2 = 0.2$.

We will test the following VM-algorithms.

1. The standard BFGS algorithm .
2. Shifted BFGS algorithm .
3. Modified algorithm (1) defined by (18).
4. Modified algorithm (2) defined by (19).

Table (4-1) shows the computational results, where the columns have the following meanings:

Problem : the name of the test problem .
NOI : number of iterations .
NOF : number of function evaluations .
F : value of the objective function at the point x.
G : gradient of the objective function at the point x.

From Table **(4-1)**, we observed that the average performances of the formula (18) & formula (19) are better than the standard BFGS and for unconstrained minimization problems.

Table (4-1)
Comparison results of all algorithm as a total of (22) test functions.

	Standard BFGS			**Shifted BFGS**		
N	**NOI**	**NOF**	**TIME**	**NOI**	**NOF**	**TIME**
10	40	290	0:00:00.02	45	297	0:00:00.02
100	96	358	0:00:00.03	59	334	0:00:00.03
1000	39	287	0:00:00.22	46	298	0:00:00.22
10000	42	293	0:00:02.33	46	299	0:00:02.33
100000	40	300	0:00:21.89	51	306	0:00:21.04
Total 5	**257**	**1528**	**0:00:24.49**	**247**	**1534**	**0:00:23.64**
	Modified (1)			**Modified (2)**		
N	**NOI**	**NOF**	**TIME**	**NOI**	**NOF**	**TIME**
10	41	291	0:00:00.02	41	291	0:00:00.02
100	54	340	0:00:00.03	49	338	0:00:00.03
1000	43	297	0:00:00.20	43	297	0:00:00.20
10000	39	299	0:00:02.33	39	299	0:00:02.34
100000	47	298	0:00:20.52	47	298	0:00:20.56
Total 5	**224**	**1525**	**0:00:23.10**	**219**	**1523**	**0:00:23.15**

The details of these results are fully described in the subsequent tables.

Table (4-2)

Algorithm	**Standard BFGS N=10**			
PROBLEM	NOI	NOF	F	G
1	3	18	38.2409094	0.914E+02
2	1	12	36943.1000	0.228E+05
3	2	15	1451.64654	0.928E+03
4	2	15	135.480501	0.281E+03
5	2	15	11.5052711	0.171E+02
6	4	19	0.209316523	0.227E+01
7	2	15	15.4996348	0.145E+02
8	1	12	53.8716349	0.307E+03
9	1	3	19.8146467	0.219E+02
10	1	12	-74.3037218	0.444E+02
11	2	15	536.191313	0.632E+03
12	5	22	126.378863	0.939E+02
13	2	15	0.490077684	0.934E+00
14	1	12	0.788519101E-03	0.299E-01
15	1	12	2.69650325	0.596E+00
16	1	12	32.8566120	0.865E+01
17	4	15	0.692944009E-01	0.562E+00
18	1	12	0.124445052	0.304E+00
19	1	12	61.8160567	0.525E+01
20	1	12	0.892519217	0.121E+01
21	1	3	2.77353532	0.347E+00
22	1	12	16.4797549	0.107E+03
	40	290		

TIME= 0:00:00.02

Table (4-3)

Algorithm	**Standard BFGS N=100**			
PROBLEM	NOI	NOF	F	G
1	3	18	571.266458	0.224E+03
2	1	12	176353.100	0.228E+05
3	2	15	13874.1436	0.621E+03
4	2	15	2401.63238	0.280E+03
5	2	15	29.8378515	0.166E+02
6	5	22	0.567193976	0.324E+01
7	2	15	72.4430234	0.134E+02
8	2	15	2106.68093	0.613E+02
9	56	57	-98.8560279	0.562E-06
10	2	24	-12.0651404	0.209E+02
11	2	15	4565.11518	0.570E+03
12	4	19	2206.20062	0.411E+03
13	2	15	0.499920183	0.995E+00
14	1	12	0.123292512E-05	0.392E-03
15	1	12	2.70244809	0.650E-01
16	1	12	2404.77898	0.844E+02
17	3	14	0.306588390	0.988E+00
18	1	12	0.133219936	0.388E-01
19	1	12	71.2531594	0.339E+02
20	1	12	10.1072709	0.712E+01
21	1	3	2.77217424	0.394E-01
22	1	12	17.9775744	0.237E+02
	96	358		

TIME= 0:00:00.03

Table (4-4)

Algorithm	**Standard BFGS N=1000**			
PROBLEM	NOI	NOF	F	G
1	3	18	5815.83263	0.333E+03
2	1	12	1570453.10	0.228E+05
3	2	15	142969.809	0.615E+03
4	2	15	25062.4645	0.280E+03
5	2	15	198.422616	0.165E+02
6	5	20	7.20555307	0.123E+02
7	2	15	656.436702	0.132E+02
8	1	12	811605.408	0.417E+03
9	1	3	600.149240	0.220E+00
10	1	12	-0.677458374	0.409E+02
11	2	15	44342.5823	0.564E+03
12	4	19	21245.1326	0.367E+03
13	2	15	0.499999209	0.999E+00
14	1	12	0.129382924E-08	0.399E-05
15	1	12	2.70251875	0.656E-02
16	1	12	230919.325	0.842E+03
17	3	14	0.346551919	0.104E+01
18	1	12	0.133332170	0.399E-02
19	1	12	146.474353	0.334E+03
20	1	12	101.466723	0.680E+02
21	1	3	2.77215807	0.399E-02
22	1	12	17.9997664	0.258E+01
	39	287		

TIME= 0:00:00.22

Table (4-5)

Algorithm	**Standard BFGS N=10000**			
PROBLEM	NOI	NOF	F	G
1	3	18	58036.4398	0.347E+03
2	1	12	15511453.1	0.228E+05
3	2	15	1433914.01	0.614E+03
4	2	15	251670.744	0.280E+03
5	2	15	1887.11490	0.165E+02
6	7	24	0.783070379	0.281E+01
7	2	15	6503.50807	0.131E+02
8	1	12	81093465.3	0.418E+03
9	1	3	6000.01493	0.220E-01
10	3	25	-0.813967853	0.313E+02
11	2	15	442092.563	0.564E+03
12	4	19	211682.548	0.362E+03
13	2	15	0.499999992	0.100E+01
14	0	1	0.130012999E	0.400E-07
15	1	12	2.70251947	0.657E-03
16	1	12	22995597.1	0.841E+04
17	3	14	0.350786442	0.105E+01
18	1	12	0.133333322	0.400E-03
19	1	12	896.496783	0.333E+04
20	1	12	1014.98070	0.677E+03
21	1	3	2.77215791	0.400E-03
22	1	12	17.9999977	0.260E+00
	42	293		

TIME= 0:00:02.33

Table (4-6)

Algorithm	**Standard BFGS N=100000**			
PROBLEM	NOI	NOF	F	G
1	3	18	580207.987	0.349E+03
2	1	12	154921453.	0.228E+05
3	2	15	14343354.9	0.614E+03
4	2	15	2517753.54	0.280E+03
5	2	15	18774.4436	0.165E+02
6	5	20	47.9223683	0.405E+02
7	2	15	64975.0381	0.131E+02
8	1	12	0.810843707E+10	0.418E+03
9	1	3	60000.0015	0.220E-02
10	3	36	-0.427345858	0.373E+02
11	2	15	4419590.11	0.564E+03
12	4	19	2116061.34	0.362E+03
13	2	15	0.500000000	0.100E+01
14	0	1	0.130076223E-14	0.400E-09
15	1	12	2.70251948	0.657E-04
16	1	12	0.229859560E+10	0.841E+05
17	3	14	0.351212385	0.105E+01
18	1	12	0.133333333	0.400E-04
19	1	12	8396.49903	0.333E+05
20	1	12	10150.1124	0.677E+04
21	1	3	2.77215791	0.400E-04
22	1	12	18.0000000	0.260E-01
	40	300		

TIME= 0:00:22.13

Table (4-7)

Algorithm	**Shift BFGS N=10**			
PROBLEM	NOI	NOF	F	G
1	5	20	13.5448572	0.261E+02
2	1	12	36943.1000	0.228E+05
3	2	15	1451.64654	0.928E+03
4	2	15	135.480501	0.281E+03
5	3	16	3.65107386	0.876E+01
6	6	23	0.817799823E-03	0.678E-01
7	2	15	15.4996348	0.145E+02
8	1	12	53.8716349	0.307E+03
9	1	3	19.8146467	0.219E+02
10	1	12	-74.3037218	0.444E+02
11	3	16	384.639070	0.293E+03
12	4	19	192.459370	0.401E+03
13	2	15	0.490077684	0.934E+00
14	1	12	0.788519101E-03	0.299E-01
15	1	12	2.69650325	0.596E+00
16	1	12	32.8566120	0.865E+01
17	4	17	-0.382799017E-01	0.163E-01
18	1	12	0.124445052	0.304E+00
19	1	12	61.8160567	0.525E+01
20	1	12	0.892519217	0.121E+01
21	1	3	2.77353532	0.347E+00
22	1	12	16.4797549	0.107E+03
	45	297		

TIME= 0:00:00.02

Table (4-8)

Algorithm	**Shift BFGS N=100**			
PROBLEM	NOI	NOF	F	G
1	4	19	380.881506	0.245E+03
2	1	12	176353.100	0.228E+05
3	2	15	13874.1436	0.621E+03
4	2	15	2401.63238	0.280E+03
5	2	15	29.8378515	0.166E+02
6	5	20	0.284469196	0.135E+01
7	2	15	72.4430234	0.134E+02
8	2	15	2106.68093	0.613E+02
9	14	28	-98.8560279	0.282E-04
10	2	24	-12.0651404	0.209E+02
11	6	19	82.7339625	0.145E+03
12	4	19	2104.08891	0.369E+03
13	2	15	0.499920183	0.995E+00
14	1	12	0.123292512E-05	0.392E-03
15	1	12	2.70244809	0.650E-01
16	1	12	2404.77898	0.844E+02
17	3	16	0.260799692	0.789E+00
18	1	12	0.133219936	0.388E-01
19	1	12	71.2531594	0.339E+02
20	1	12	10.1072709	0.712E+01
21	1	3	2.77217424	0.394E-01
22	1	12	17.9775744	0.237E+02
	59	334		

TIME= 0:00:00.03

Table (4-9)

Algorithm	**Shift BFGS N=1000**			
PROBLEM	NOI	NOF	F	G
1	4	19	3872.53795	0.364E+03
2	1	12	1570453.10	0.228E+05
3	2	15	142969.809	0.615E+03
4	2	15	25062.4645	0.280E+03
5	2	15	198.422616	0.165E+02
6	7	24	0.327658716E-01	0.576E+00
7	2	15	656.436702	0.132E+02
8	1	12	811605.408	0.417E+03
9	1	3	600.149240	0.220E+00
10	1	12	-0.677458374	0.409E+02
11	6	19	336.226588	0.135E+03
12	4	19	21148.6523	0.363E+03
13	2	15	0.499999209	0.999E+00
14	1	12	0.129382924E-08	0.399E-05
15	1	12	2.70251875	0.656E-02
16	1	12	230919.325	0.842E+03
17	3	16	0.341274143	0.831E+00
18	1	12	0.133332170	0.399E-02
19	1	12	146.474353	0.334E+03
20	1	12	101.466723	0.680E+02
21	1	3	2.77215807	0.399E-02
22	1	12	17.9997664	0.258E+01