A New Formulation for Quasi-Newton Methods

ISSAM A.R. MOGHRABI
Department of M.I.S.
Gulf University for Science and Technology
Abdullah Mubarak area, West Mishref
KUWAIT
moughrabi.i@gust.edu.kw http://www.gust.edu.kw

Abstract: - We develop a framework for the construction of multi-step quasi-Newton methods which utilize values of the objective function. The model developed here is constructed via interpolants of the m+1 most recent iterates / gradient evaluations, and possesses double free parameters which introduce an additional degree of flexibility. This permits the interpolating polynomials to exploit function-values which are readily available at each iteration. A new algorithm is derived for which function values are incorporated in the update of the inverse Hessian approximation at each iteration, in an attempt to accelerate convergence. The idea of incorporating function values is not new within the context of quasi-Newton methods but the presentation made in this paper presents a new approach for such algorithms. It has been shown in several earlier works that Including function values data in the update of the Hessian approximation numerically improves the convergence of Secant-like methods. The numerical performance of the new method is assessed with promising results.

Key-Words: - Unconstrained optimization, quasi-Newton methods, multi-step methods, function value algorithms, nonlinear programming, Newton method

1 Introduction

The "Newton Equation" ([3]), which may be regarded as a generalization of the "Secant Equation" (Dennis and Schnabel [4]), is usually employed in the construction of quasi-Newton methods for optimization.

Let $f(\underline{x})$ be the objective function, where $\underline{x}\, \varepsilon\, R^n$, and let $\underline{g}$ and G denote the gradient and Hessian of f, respectively. Let $X = \{\underline{x}(\tau)\}$ denote a differentiable path in R^n, where $\tau\, \varepsilon\, R$. Then,upon applying the Chain Rule to $\underline{g}(\underline{x}(\tau))$ in order to determine its derivative with respect to τ, we obtain

$$G(\underline{x}(\tau))\underline{x}'(\tau) = \underline{g}'(\underline{x}(\tau)). \qquad (1)$$

In particular, if we choose for the path X to pass through the most recent iterate $\underline{x}_{i+1}$ (so that $\underline{x}(\tau_m) = \underline{x}_{i+1}$, say), then equation (1) provides a condition (termed the "Newton Equation" in [1,3]) which the Hessian $G(\underline{x}_{i+1})$ must satisfy:

$$G(\underline{x}_{i+1})\underline{x}'(\tau_m) = \underline{g}'\left(\underline{x}(\tau_m)\right). \qquad (2)$$

Therefore, if B_{i+1} denotes an approximation to $G(\underline{x}_{i+1})$, if

$$\underline{r_i} \stackrel{\text{def}}{=} \underline{x}'(\tau_m) \qquad (3)$$

and $\underline{w_i}$ denotes an approximation to $\underline{g}'(\underline{x}(\tau_m))$, it is reasonable (by equation (2)) to require that B_{i+1} should satisfy the relation

$$B_{i+1}\underline{r_i} = \underline{w_i}. \qquad (4)$$

(The derivation, in particular, of the Secant equation from the Newton Equation is described in [4].) The relation in (4) defines the basis of the the Multi-step methods derived in [1, 2]. In [9], it was proposed that X should be the vector polynomial which interpolates the $m+1$ most recent iterates $\{\underline{x}_{i-m+k+1}\}_{k=0}^{m}$ and that $\underline{w_i}$ should be obtained by constructing and differentiating the corresponding vector polynomial ($\hat{\underline{g}}(\tau)$, say) which interpolates the known gradient values $\{\underline{g}(\underline{x}_{i-m+k+1})\}_{k=0}^{m}$. Thus, the following explicit expressions for $\underline{r_i}$ and $\underline{w_i}$ may be derived [8,9,10,11]:

$$\underline{r_i} = \underline{x}'(\tau_m)$$
$$= \textstyle\sum_{j=0}^{m-1} \underline{s}_{i-j}\{\sum_{k=m-j}^{m} \mathcal{L}'_k(\tau_m)\}; \qquad (5)$$

$$\underline{w_i} = \underline{\hat{g}}'(\tau_m)$$

$$= \sum_{j=0}^{m-1} \underline{y}_{i-j}\{\sum_{k=m-j}^{m} \mathcal{L}_k'(\tau_m)\} \quad (6)$$

$$\approx \underline{g}'(\underline{x}(\tau_m)).$$

(In the equations above,

$$\underline{s}_i \overset{\Delta}{=} \underline{x}_{i+1} - \underline{x}_i, \quad (7)$$

$$\underline{y}_i \overset{\Delta}{=} \underline{g}(\underline{x}_{i+1}) - \underline{g}(\underline{x}_i) \quad (8)$$

and $\mathcal{L}_j(\tau)$ is the j^{th} Lagrange polynomial of degree m corresponding to the set of values $\{\tau_k\}_{k=0}^{m}$, so that $\mathcal{L}_j(\tau_j) = 1$ and $\mathcal{L}_j(\tau_i) = 0$ for $i \neq j$. The scalars $\{\tau_k\}_{k=0}^{m}$ are the values of τ associated with the iterates $\{\underline{x}_{i-m+k+1}\}_{k=0}^{m}$ on the path $X = \{\underline{x}(\tau)\}$:

$$\underline{x}(\tau_k) = \underline{x}_{i-m+k+1}, \text{ for } k = 0,1,\ldots,m.) \quad (9)$$

We now stipulate, arbitrarily, that the set $\{\tau_j\}_{j=0}^{2}$ has been chosen such that

$$\tau_1 = 0, \quad (10)$$

and then write

$$\tau_1 - \tau_0 = -\tau_0 \overset{\text{def}}{=} \rho_{i-1} > 0;$$

$$\tau_2 - \tau_2 = \tau_2 \overset{\text{def}}{=} \rho_i > 0, \quad (11)$$

where, for example, the quantities ρ_{i-1} and ρ_i could be defined (as they are in method **A1** [1], where other ways of defining ρ_{i-1} and ρ_i are also discussed) by

$$\rho_{i-1} = \| \underline{s}_{i-1} \|_2; \ \rho_i = \| \underline{s}_i \|_2. \quad (12)$$

In this paper, we will investigate a class of parameterized models for the path X. The free parameters in such models can be viewed as providing a means by which more information can be utilized in updating the Hessian approximation (or its inverse), as in the methods derived in [8]. We describe, in the next sections, the non-linear model and then the particular technique (which essentially involves making use of the function-values at our disposal from the m most recent iterations) that will be used in determining the free parameters. We finally present the numerical test results conducted on the method and evaluate those results.

2 The Nonlinear Model

We investigate here a model that embodies two free parameters (namely, ϑ_1 and ϑ_2), which allow us to specify that the model satisfies, simultaneously, more than one property. For example, we may require that the parameters are determined such that

$$\phi(\tau_2, \vartheta_1, \vartheta_2) = f_{i+1} \quad (13)$$

and

$$\phi(\tau_1, \vartheta_1, \vartheta_2) = f_i. \quad (14)$$

hold simustaneously, for

$$\emptyset(\tau, \vartheta_1, \vartheta_2) \overset{\Delta}{=} f\big(x(\tau, \vartheta_1, \vartheta_2)\big).$$

Our model is defined by

$$\psi_i(\tau, \vartheta_1, \vartheta_2) \overset{\Delta}{=} t^i / \propto (\tau, \vartheta_1, \vartheta_2) \quad (15)$$

for $i = 0,1,2$, (for a 2-step method) and where

$$\propto (\tau, \vartheta_1, \vartheta_2) \overset{\Delta}{=} 1 + \vartheta_1\tau + \vartheta_2\tau^2.$$

Alternatively, if we express this polynomial in its Lagrangian form, we obtain.

$$u(\tau, \vartheta_1, \vartheta_2) \overset{\Delta}{=} \frac{1}{\propto(\tau, \vartheta_1, \vartheta_2)}$$

$$\{\frac{\tau(\tau+\rho_{i-1})}{\rho_i(\rho_{i-1}+\rho_i)}[1+\vartheta_1\rho_i+\vartheta_2\rho_i^2]u_{i+1}$$

$$-\frac{\tau+\rho_{i-1}}{\rho_{i-1}\rho_i}(\tau-\rho_i)u_i$$

$$+\frac{\tau(\tau-\rho_i)}{\rho_{i-1}(\rho_{i-1}+\rho_i)}[1-\vartheta_1\rho_{i-1}+\vartheta_2\rho_{i-1}^2]u_{i-1}\}$$

$$\overset{\Delta}{=} q(\tau, \vartheta_1, \vartheta_2)/\propto (\tau, \vartheta_1, \vartheta_2). \quad (16)$$

Now from (16), we obtain

$$u'(\tau, \vartheta_1, \vartheta_2) \overset{\Delta}{=} \frac{1}{\propto (\tau, \vartheta_1, \vartheta_2)}$$

$$[g'(\tau, \vartheta_1, \vartheta_2) \propto (\tau, \vartheta_1, \vartheta_2) - q(\tau, \vartheta_1, \vartheta_2)(\vartheta_1 + 2\vartheta_2\tau)]$$

$$= [g'(\tau,\vartheta_1,\vartheta_2) - (\vartheta_1 + 2\vartheta_2\tau)u(\tau,\vartheta_1,\vartheta_2)]/\propto (\tau,\vartheta_1,\vartheta_2). \quad (17)$$

From (17), it follows that $u'(\tau,\theta_1,\theta_2)$ at the three points, $\tau_0 = -\rho_{i-1}, \tau_1 = 0$ and $\tau_2 = \rho_i$ (see (11) and (12)), is given, respectively, by the following expressions

$$u'(-\rho_{i-1},\vartheta_1,\vartheta_2) = \frac{q'(-\rho_{i-1},\vartheta_1,\vartheta_2) - (\vartheta_1 - 2\vartheta_2\rho_{i-1})u_{i-1}}{1 - \vartheta_1\rho_{i-1} + \vartheta_2\rho_{i-1}^2}, \quad (18)$$

$$u'(0,\vartheta_1,\vartheta_2) = q'(0,\vartheta_1,\vartheta_2) - \vartheta_1 u_i, \quad (19)$$

and

$$u'(\rho_i,\vartheta_1,\vartheta_2) = \frac{q'(\rho_i,\vartheta_1,\vartheta_2) - (\vartheta_1 + 2\vartheta_2\rho_i)u_{i+1}}{1 + \vartheta_1\rho_i + \vartheta_2\rho_i^2}. \quad (20)$$

Using (16), we derive

$$q'(\tau,\vartheta_1,\vartheta_2) = \left\{\frac{2\tau + \rho_{i-1}}{\rho_{i-1}}(1 + \vartheta_1\rho_i + \vartheta_2\rho_i^2)u_{i+1} - \frac{(2\tau + \rho_{i-1} - \rho_i)\mu}{\rho_{i-1}\rho_i}u_i\right\}$$

$$+ \frac{2\tau - \rho_i}{\rho_{i-1}}[1 - \vartheta_1\rho_{i-1} + \vartheta_2\rho_{i-1}^2]u_{i-1}\}\mu^{-1}, \quad (21)$$

from which we obtain the following quantities (for $\mu \stackrel{\Delta}{=} \rho_i + \rho_{i-1}$, $\delta \stackrel{\Delta}{=} -\rho_i/\rho_{i-1}$ and $\Delta u_j \stackrel{\Delta}{=} u_{j+1} - u_j$):

$$q'(0,\vartheta_1,\vartheta_2) = \mu^{-1}\{-\delta^{-1}(1 + \vartheta_1\rho_i + \vartheta_2\rho_i^2)u_{i+1} + (\delta^{-1} - \delta)u_i) + [1 - \vartheta_1\rho_{i-1} + \vartheta_2\rho_{i-1}^2]\delta u_{i-1}$$

$$= \mu^{-1}\{-\delta^{-1}\Delta u_i - \delta\Delta u_{i-1} + \vartheta_2\rho_{i-1}\rho_i(\Delta u_{i-1} + \Delta u_i) + \vartheta_1\rho_{i-1}u_{i+1} + \vartheta_1\rho_i u_{i-1}\}, \quad (22)$$

and

$$q'(0,\vartheta_1,\vartheta_2) = \mu^{-1}\{(2 - \delta^{-1})(1 + \vartheta_1\rho_i + \vartheta_2\rho_i^2)u_{i+1} + (\delta - 2 + \delta^{-1})u_i$$

$$-\delta(1 - \vartheta_1\rho_{i-1} + \vartheta_2\rho_{i-1}^2)u_{i-1}$$

$$= \mu^{-1}\{2\Delta u_i - \delta^{-1}\Delta u_i + \delta\Delta u_{i-1} - \vartheta_1\rho_{i-1}u_{i-1} - \vartheta_1\rho_i u_{i+1} + \vartheta_2\rho_{i-1}\rho_i[\Delta u_{i-1} + \Delta u_i] + 2\vartheta_1\rho_i u_{i+1} + 2\vartheta_2\rho_i^2 u_{i+1} + \vartheta_1\rho_{i-1}u_{i+1} - \vartheta_1\rho_i u_{i-1}\}, \quad (23)$$

and

$$q'(-\rho_{i-1},\vartheta_1,\vartheta_2) = \mu^{-1}\{\delta^{-1}(1 + \vartheta_1\rho_i + \vartheta_2\rho_i^2)u_{i+1} + (\delta - 2 + \delta^{-1})u_i + (\delta - 2)(1 - \vartheta_1\rho_{i-1} + \vartheta_2\rho_{i-1}^2)u_{i-1}$$

$$= \mu^{-1}\{\delta^{-1}\Delta u_i + (2 - \delta)\Delta u_{i-1} - \vartheta_1\rho_{i-1}[\Delta u_{i-1} + \Delta u_i] - \vartheta_2\rho_{i-1}\rho_i u_{i+1} + \vartheta_1\rho_{i-1}u_{i-1} - 2\vartheta_2\rho_{i-1}^2 u_{i-1} + \vartheta_1\rho_i u_{i-1} - \vartheta_2\rho_{i-1}\rho_i u_{i-1}. \quad (24)$$

We are now able to determine the quantities q' in (18), (19) and (20) as follows

$$q'(0,\vartheta_1,\vartheta_2) - \vartheta_1 u_i = \mu^{-1}\{-\delta^{-1}\Delta u_i - \delta\Delta u_{i-1} + \vartheta_2\rho_{i-1}\rho_i[\Delta u_{i-1} + \Delta u_i] + \vartheta_1(\rho_{i-1}\Delta u_i - \rho_i\Delta u_{i-1})\} \quad (25)$$

and

$$q'(\rho_i,\vartheta_1,\vartheta_2) - (\vartheta_1 + 2\vartheta_2\rho_i)u_{i+1} = \mu^{-1}\{(2 - \delta^{-1})\Delta u_i + \delta\Delta u_{i-1}\vartheta_2\rho_{i-1}\rho_i[\Delta u_{i-1} + \Delta u_i] + \vartheta_1\rho_i[\Delta u_{i-1} + \Delta u_i]\} \quad (26)$$

and

$$q'(-\rho_{i-1},\vartheta_1,\vartheta_2) - (\vartheta_1 - 2\vartheta_2\rho_{i-1})u_{i-1} = \mu^{-1}\{\delta^{-1}\Delta u_i + (2 - \delta)\Delta u_{i-1} - \vartheta_1\rho_{i-1}[\Delta u_{i-1} + \Delta u_i] - \vartheta_2\rho_{i-1}\rho_i[\Delta u_{i-1} + \Delta u_i]\}. \quad (27)$$

3 A New Function Value Algorithm

Algorithm Df1

For this algorithm, the free parameters θ_1 and θ_2 are determined via requiring the following relations

$$\phi'(0,\theta_1,\theta_2)[\tau_2-\tau_0]=f(x_{i+1})-f(x_{i-1}) \quad (28)$$

(for $\phi'(0,\theta_1,\theta_2)\stackrel{\Delta}{=}x'(0,\theta_1,\theta_2)^T g_i$)

and

$$x'(\tau_0,\theta_1,\theta_2)^T g_{i-1}=\omega'(\tau_0) \quad (29)$$

to hold simultanously and where $\omega(\tau)$ is the quadratic polynomial which interpolates the three most recent function values, f_{i-1} , f_i and f_{i+1}. That is,

$$\omega(\tau)=\alpha\tau^2+\beta\tau+\gamma.$$

The coefficients α, β, and γ are given by

$$\gamma=f_i,$$

$$\beta=((\delta^{-1}-\delta)f_i-\delta^{-1}f_{i+1}+\delta f_{i-1})/\mu,$$

and

$$\alpha=(f_{i+1}+(\delta-1)f_i-\delta f_{i-1})/\mu\tau_2.$$

Now from (28), we obtain (using (25) and (28))

$$-\delta^{-1}\sigma_{ii}-\delta\sigma_{i-1,i}+\vartheta_2\rho_i\rho_{i-1}(\sigma_{ii}+\sigma_{i-1,i})+\vartheta_1(\rho_{i-1}\sigma_{ii}-\rho_i\sigma_{i-1,i})=f_{i+1}-f_{i-1}. \quad (30)$$

Also using (29) we obtain

$$\vartheta_1\{\rho_{i-1}(\sigma_{i,i-1}+\sigma_{i-1,i-1}-\mu\omega'(-\rho_{i-1}))\}+\vartheta_2\{\rho_{i-1}\mu\omega'(-\rho_{i-1})+\rho_i\rho_{i-1}(\sigma_{i,i-1}+\sigma_{i-1,i-1})\}$$

$$=(2-\delta)\sigma_{i-1,i-1}+\delta^{-1}\sigma_{i,i-1}-\mu\omega'(-\rho_{i-1}) \quad (31)$$

If we define the quantities

$$\eta\stackrel{\Delta}{=}\rho_{i-1}\sigma_{i,i}-\rho_i\sigma_{i-1,i},$$

$$\zeta\stackrel{\Delta}{=}\rho_{i-1}[\sigma_{i,i-1}+\sigma_{i-1,i-1}-\mu\omega'(-\rho_{i-1})]$$

$$\nu\stackrel{\Delta}{=}\rho_i\rho_{i-1}(\sigma_{ii}+\sigma_{i-1,i})$$

$$\varepsilon\stackrel{\Delta}{=}\rho_{i-1}\mu\omega'(-\rho_{i-1})+\rho_i\rho_{i-1}(\sigma_{i,i-1}+\sigma_{i-1,i-1})$$

$$\pi\stackrel{\Delta}{=}f_{i+1}-f_{i-1}+\delta^{-1}\sigma_{ii}+\delta\sigma_{i-1,i}$$

and

$$\lambda\stackrel{\Delta}{=}(2-\delta)\sigma_{i-1,i-1}+\delta^{-1}\sigma_{i,i-1}-\mu\omega'(-\rho_{i-1})$$

and solve, simultaneously equations (28) and (29) for the unknowns θ_1 and θ_2, we obtain the following expressions for the two parameters

$$\vartheta_1=(\varepsilon\pi-\lambda\upsilon)/(\eta\varepsilon-\zeta\upsilon) \quad (32)$$

and

$$\vartheta_2=(\eta\lambda-\zeta\pi)/(\varepsilon\eta-\lambda\zeta), \quad (33)$$

for a denominator that must be safeguarded numerically against vanishing. For this algorithm, we update the Hessian approximation to satisfy

$$B_{i+1}\{(2-\delta^{-1}-\vartheta_2\rho_i\rho_{i-1}+\vartheta_1\rho_i)s_i+(\vartheta_1\rho_i-\vartheta_2\rho_i\rho_{i-1}+\delta)s_{i-1}\}$$

$$=(2-\delta^{-1}-\vartheta_2\rho_i\rho_{i-1}+\vartheta_1\rho_i)s_i+(\vartheta_1\rho_i-\vartheta_2\rho_i\rho_{i-1}+\delta)s_{i-1}. \quad (34)$$

4 Numerical Experiments

Ten standard test functions were used, each with two starting-points, giving a total of twenty problems. (Full details of the test functions and starting points may be found in [12].) The functions involved in the numerical experiments are tested on varying dimensions, wherever applicable, thus giving a total of 720 problems.

In all the methods considered here, the new point $\underline{x}_{i+1}$ was computed from $\underline{x}_i$ via a line-search algorithm which accepted the predicted point if the two standard stability conditions given below were satisfied and which, otherwise, used step-doubling and safeguarded cubic interpolation, as appropriate [5,6,7,16]. To be acceptable, $\underline{x}_{i+1}$ was required to satisfy the following conditions (see [13,14,15]):-

$$f(\underline{x}_{i+1})\le f(\underline{x}_i)+10^{-4}\underline{s}_i^T\,\underline{g}(\underline{x}_i)$$

and

$$\underline{s}_i^T \underline{g}(\underline{x}_{i+1}) \geq 0.9\{\underline{s}_i^T \underline{g}(\underline{x}_i)\}.$$

It is easy to show (by analogy with standard theory for the BFGS method [1,2,3]) that a necessary and sufficient condition for preserving positive-definiteness in the successive matrices $\{H_i\}$ is that $\underline{r}_i^T \underline{w}_i > 0$. In practice, we have imposed (in the implementations) the following requirement:-

$$\underline{r}_i^T \underline{w}_i > 10^{-4} \parallel \underline{r}_i \parallel_2 \parallel \underline{w}_i \parallel_2,$$

in order to ensure that $\underline{r}_i^T \underline{w}_i$ is "sufficiently" positive and thus avoid possible numerical instability in computing H_{i+1}. If this condition on $\underline{r}_i^T \underline{w}_i$ was not satisfied, the algorithm reverted to the choice $\theta_1 = \theta_2 = 0$.

The results presented in Table 1 show, for each problem, the number of function/gradient evaluations required to solve the problem is given followed (in brackets) by the number of iterations. The experiments were carried out on the new method, the standard BFGS algorithm and the method A1 that performed best in [9,10]. On the basis of these results, the new method gives the best numerical performance of the methods included in our tests

Table 1. Comparison of **Df1** with **BFGS and A1**.

Problem	Df1	BFGS	A1
Watson (a)	443(429)	542(530)	443(429)
(b)	1411(983)	1653(1293)	1487(1101)
Rosenbrock (a)	475(444)	631(612)	485(465)
(b)	672(634)	825(806)	679(652)
Ext. Powell (a)	201(124)	159(122)	174(129)
(b)	124(89)	162(156)	108(97)
Penalty fn. (a)	765(342)	807(386)	765(344)
(b)	577(450)	637(507)	569(463)
Trigonom-etric (a)	390(330)	634(596)	549(485)
(b)	733(668)	2390(2331)	1973(1877)
Broyden (a)	1479(860)	2300(1786)	1912(1092)
(b)	1208(1050)	2136(2001)	1772(1607)
Wolfe (a)	287(241)	280(226)	266(225)
(b)	2043(975)	1967(965)	1936(917)
Tridiagonal (a)	4918(1939)	5099(2272)	4696(1892)
(b)	3075(1974)	2822(1969)	2765(1819)
Powell (a)	780(667)	1034(987)	735(680)
(b)	1735(1382)	1779(1559)	1623(1379)
Sphere (a)	155(136)	132(126)	130(124)
(b)	532(391)	1505(1262)	1050(718)
TOTALS	22003(14108)	7494(20492)	24117(16495)
RATIOS	80.0%(68.8%)	100%(100%)	7.7%(80.5%)

5 Conclusion

A new model for the interpolating curve used in constructing multi-step quasi-Newton methods has been introduced. The model includes two free parameters and it has been shown how those parameters may be determined by using available values of the objective function to produce numerical estimates of the derivative of the function at the latest three iterates. It has always been argued that much of the data computed at each iteration is renounced without making use of it. One particular algorithm is derived here that shows a significant improvement, in numerical terms, over the standard (single-step) BFGS method and an earlier successful multi-step method. The idea presented here is new in terms of the incorporation of free parameters that provide a tool to utilize any desired available data in the update of the Hessian (or its inverse) approximation to improve the numerical performance of Secant-like methods.

References:

[1] Al-Baali, M. "New property and global convergence of the Fletcher–Reeves method with inexact line searches", *IMA J. Numer. Anal.*, 5, Feb. 1985, pp. 122–124.

[2] Broyden, C.G. "The convergence of a class of double-rank minimization algorithms - Part 2: The new algorithm", *J. Inst. Math. Applic.,* 6, June 1970, pp. 222-231.

[3] Byrd, R.H., Schnabel, R.B., and Shultz, G.A., "Parallel quasi-Newton methods for unconstrained optimization", *Math. Programing*, 42, April 1988, pp. 273-306.

[4] Dennis, J.E. and Schnabel, R.B., "Least change Secant updates for quasi-Newton methods", *SIAM Review,* 21, Feb. 1979, pp. 443-459.

[5] Fletcher, R., *Practical Methods of Optimization* (second edition), Wiley, New York, 1987.

[6] Fletcher, R., "A new approach to variable metric algorithms", *Comput. J.* , 13, April 1970, pp. 317-322.

[7] Fletcher, R. and Reeves, C., "Function minimization by conjugate gradients", *Computer J.,* 7, Jan. 1964, pp. 149–154.

[8] Ford, J.A. and Moghrabi I.A.R., "Using function-values in multi-step quasi-Newton methods", *J. Comput. Appl. Math.*, 66, March 1996, pp. 201-211.

[9] Ford, J.A. and Moghrabi I.A.R., "Multi-step quasi-Newton methods for optimization", J. *Comput. Appl. Math.*, 50, Jan. 1994, pp. 305-323.

[10] Ford, J.A. and Moghrabi I.A.R., "Alternative parameter choices for multi-step quasi-Newton methods", *Optimization Methods and Software*, 2, June 1993, pp. 357-370.

[11] Moghrabi, I.A.R., "Implicit extra-update multi-step quasi-newton methods", Int. J. Operational Research, 28, August 2017, pp. 69-81.

[12] Moré, J.J., Garbow, B.S., Hillstrom, K.E., "Testing unconstrained optimization software", *ACM Trans. Math. Softw.*, **7**, April 1981, pp. 17–41.

[13] Shanno, D.F. and Phua, K.H., "Matrix conditioning and nonlinear optimization", *Math. Programming* ,14, May 1978, pp. 149-160.

[14] Shanno, D.F., "On the convergence of a new conjugate gradient algorithm", *SIAM J. Numer. Anal.*, 15, Jan. 1978, pp. 1247–1257.

[15] Wolfe, P., "Convergence conditions for ascent methods II: Some corrections", *SIAM Rev.,* 13, June 1971, pp. 185-188