Transactions of the Nigerian Association of Mathematical Physics
Vol. 6 **(Jan., 2018), pp. 81– 86**

In honour of Prof. Ekhaguere at 70

A robust Broyden-like method for systems of nonlinear equations

I. A. Osinuga[a*] **and S. O. Yusuff**[b]

[a,b]*Department of Mathematics, Federal University of Agriculture, P.M.B. 2240, Abeokuta, Ogun State, Nigeria*

Abstract. In this paper, we propose and analyze new predictor-corrector method based on the quasi-newton approach for solving a nonlinear system of equations using the trapezoidal and midpoint quadrature formulas. A simple and flexible iterative method is proposed to determine the real roots of a system of nonlinear equations. The proposed method is the weight combination of Midpoint and Trapezoidal (MT). Numerical results of the method show that the proposed algorithm is robust.

Keywords: Broyden method, quadrature formula, predictor-corrector, nonlinear systems.

1. Introduction

Solving equations is very important in applied mathematics, since most of the mathematical models relating to scientific problems involve finding some values which are the solutions of equations. Those equations usually cannot be solved analytically, but require a numerical scheme to approximate the solutions [4]. Iterative algorithms for solving a system of nonlinear algebraic equations date back to the seminal work of Isaac Newton. We consider the standard problem of identifying the solution of a system of nonlinear equations is

$$F(x) = 0 \tag{1}$$

where the function $F : R^n \to R^n$ is a nonlinear differentiable function and n is large. The value of x is then called a solution or root of this equation and may be just one of many equations.The disadvantages of the Newton method arise from the need to calculate and invert the Jacobian matrix $J(x)$ at each iteration. Quasi-Newton methods were introduced because of the weaknesses of the Newton method. The most successful and simplest quasi-Newton method for solving nonlinear systems of equations is the Broyden method [14]. Broyden method was introduced due to its being a powerful alternative to Newton method, and hence it reduces the amount of calculation at each iteration, but the number of iterations needed to converge to a solution has increased, which inversely reduced the convergence order from quadratic to superlinear. Broyden method is given by

$$x_{k+1} = x_k - B_k^{-1} F(x_k) \tag{2}$$

where $x := \{x_1, \cdots, x_n\}$, $F := F_1, \cdots, F_n$ and B_k is an $n \times n$ matrix which is an approximation of the Newton Jacobian $J = F'(x_k) = \frac{\partial F_i}{\partial x_j}$, such that the quasi-Newton equation

$$B_{k+1}(x_{k+1} - x_k) = F(x_{k+1}) - F(x_k)$$

is satisfied for each k. Recently, several iterative methods have been developed to solve nonlinear equations and the system of nonlinear equations. These methods have been improved using Taylor series, quadrature formula [2, 3, 5, 19, 20], homotopy perturbation method and decomposition techniques [13, 16, 12, 6]. Quadrature-based Broyden-like methods have also been proposed [17, 14] and

*Corresponding author. Email: osinuga08@gmail.com

the references therein. A Broyden-like method using the trapezoidal rule was proposed by [17] in order to solve a system of nonlinear equations and reduce the number of iterations of the Classical Broyden method. Also [14] used the weighted combination of the midpoint and simpson quadrature formulas to propose a Broyden-like. Our aim in this work is to present alternative methods that will reduce the number of iterations required by the classical Broyden method to converge to a solution preserving its local order of convergence. The proposed methods is two-step in nature, where the first step is the classical Broyden method and the second step is our quadrature-based Broyden method. In Section 2, the derivation process of MT is given, Numerical results were discussed in Section 3 and section 4 is for the conclusion.

2. Derivation process of the proposed method

Let $F : D \subset R^n \to R^n$ be a sufficiently differentiable function in the convex set $D \subset R^n$ and let x^* be a zero of the nonlinear system of equations (1). For any $x, y \in D$,

$$F(y) = F(x) + F'(x)(y-x) + \frac{1}{2!}F''(x)(y-x)^2 + \cdots + \frac{1}{(r-1)!}F^{r-1}(y-x)^{r-1}$$
$$+ \int_0^1 \frac{(1-t)^{r-1}}{r!}F^r(x - t(y-x))(y-x)^r dt$$

is verified from Taylor's series. Then for $r = 1$, we have

$$F(y) = F(x) + \int_0^1 F'(x - t(y-x))(y-x)dt$$

from which we obtain, for kth iteration x_k:

$$F(y) = F(x_k) + \int_0^1 F'(x_k - t(y-x_k))(y-x_k)dt. \tag{3}$$

An estimation of Equation (3) with the weight combination of the Midpoint and Trapezoidal quadratures for $y = \bar{x}$ gives:

$$0 \approx F(x_k) + \frac{1}{4}\left(J_F(x_k) + 2J_F\left(\frac{x_k + \bar{x}}{2}\right) + J_F(\bar{x})\right)(\bar{x} - x_k)dt \tag{4}$$

So a new approximation x_{k+1} of $\bar{x}$ is:

$$x_{k+1} = x_k - 4\left[J_F(x_k) + 2J_F\left(\frac{x_k + x_{k+1}}{2}\right) + J_F(x_{k+1})\right] \tag{5}$$

Now, if we replace $J_F(x_k)$, $J_F(x_{k+1})$ and $J_F\left(\frac{x_k+x_{k+1}}{2}\right)$ by $B(x_k)$, $B(x_{k+1})$ and $B\left(\frac{x_k+x_{k+1}}{2}\right)$ respectively and use the same procedure as in [2, 1, 19], we have

$$x_{k+1} = x_k - 4\left[B(x_k) + 2B\left(\frac{x_k + x_{k+1}}{2}\right) + B(x_{k+1})\right]^{-1} F(x_k)$$

which is an implicit equation because we have x_{k+1} on both sides. In order to avoid the implicit nature of this equation, we use the $(k+1)$th iteration of the Broydens method $m_k = x_k - B_k^{-1}F(x_k)$ in the right hand side. Thus we have

$$x_{k+1} = x_k - 4\left[B(x_k) + 2B\left(\frac{x_k + m_k}{2}\right) + B(m_k)\right]^{-1} F(x_k)$$

which gives

$$x_{k+1} = x_k - 4[B(x_k) + 2B(z_k) + B(m_k)]^{-1}F(x_k) \tag{6}$$

for $z_k = \left(\frac{x_k+m_k}{2}\right)$. Suppose we set $B_k = 4[B(x_k) + 2B(z_k) + B(m_k)]$, then we have

$$x_{k+1} = x_k - 4B_k^{-1}F(x_k). \tag{7}$$

Hence we have the following two-step method using initial matrix $B_0 = I$ and an initial guess x_0.

For a given x_0 using initial matrix $B_0 = I$, compute the approximates solution x_{k+1} by the iterative schemes

$$\begin{aligned} m_k &= x_k - B_k^{-1}F(x_k) \\ x_{k+1} &= x_k - 4[B(x_k) + 2B(z_k) + B(m_k)]^{-1}F(x_k) \end{aligned} \tag{8}$$

for $z_k = \left(\frac{x_k+m_k}{2}\right)$, $k = 0, 1, \cdots$

3. Numerical Results and Discussion

In order to evaluate the performance of the proposed method, we apply the method to solve five (5) benchmark problems using eight (8) dimensions of 5-1065 variables. A comparison of the numerical test results of our new method is made with those of the following three well-known methods:

- Classical Broyden Method [9]
- Trapezoidal Broyden Method [17]
- Midpoint-Simpson Broyden Method [14]

The comparison was done on the number of iterations and the CPU time in seconds. The computational experiments were carried out using MATLAB 2012b with a double precission arithmetic. The program is designed to terminate whenever the number of iterations reaches 500. We used a stopping criteria $\|F(x_k)\| \leq 10^{-12}$ for the computer programs if no x_k satisfies. A failure is reported (denoted by '-') in the tabulated result.

List of Tested Problems

Problem 1 [1]
$F_i(x) = x_i x_{i+1} - 1$,
$F_n(x) = x_n x_1 - 1$.
$i = 1, 2, \cdots, n-1$ and $x_0 = (0.8, 0.8, \cdots, 0.8)^T$.

Problem 2 [7]
$F_i(x) = x_i x_{i+1} - 1$,
$F_n(x) = x_n x_1 - 1$.
$i = 1, 2, \cdots, n-1$ and $x_0 = (2, 2, \cdots, 2)^T$.

Problem 3 [19]
$F_i(x) = x_i^2 - \cos(x_i - 1)$,
$i = 1, 2, \cdots, n$ and $x_0 = (2, 2, \cdots, 2)^T$.

Problem 4 [7]
$F_i(x) = x_i^2 - 1$,
$i = 1, 2, \cdots, n$ and $x_0 = (0.5, 0.5, \cdots, 0.5)^T$.

Problem 5 [18]
$F_i(x) = \exp(x_i^2 - 1) - \cos(1 - x_i^2)$,
$i = 1, 2, \cdots, n$ and $x_0 = (0.5, 0.5, \cdots, 0.5)^T$.

Problem 6 [18]
$F_i(x) = \exp(x_i) - 1$,
$i = 1, 2, \cdots, n$ and $x_0 = (0.5, 0.5, \cdots, 0.5)^T$.

Table 1. Numerical Results for Systems (1) - (6)

Prob	n	CB		TB		MSB		MT	
		NI	CPU	NI	CPU	NI	CPU	NI	CPU
1	5	6	0.051	4	0.048	3	0.039	4	0.045
	15	6	0.065	4	0.084	3	0.062	4	0.078
	35	5	0.155	5	0.164	3	0.103	4	0.137
	65	5	0.215	5	0.389	3	0.247	4	0.329
	165	5	0.515	5	0.574	4	0.437	4	0.511
	365	6	1.521	5	1.424	4	1.648	4	1.101
	665	6	2.794	5	2.469	4	2.351	4	2.118
	1065	6	5.131	5	4.463	4	4.225	4	3.717
2	5	-	-	-	-	5	0.058	5	0.059
	15	-	-	-	-	5	0.063	5	0.093
	35	-	-	-	-	5	0.131	5	0.161
	65	-	-	-	-	5	0.241	5	0.338
	165	-	-	-	-	5	0.510	6	0.634
	365	-	-	-	-	5	2.895	6	1.218
	665	-	-	-	-	5	2.663	6	2.862
	1065	-	-	-	-	5	5.648	6	5.174
3	5	12	0.061	22	0.179	6	0.071	7	0.105
	15	12	0.163	24	0.385	6	0.088	7	0.139
	35	12	0.296	-	-	6	0.175	7	0.414
	65	12	1.386	24	1.873	6	0.380	7	0.422
	165	12	1.703	-	-	6	0.739	7	1.123
	365	14	6.819	-	-	7	3.069	7	3.290
	665	14	13.169	-	-	7	4.311	7	7.393
	1065	15	16.180	-	-	7	7.489	7	6.322
4	5	6	0.053	5	0.044	4	0.078	4	0.040
	15	6	0.069	5	0.134	4	0.056	4	0.075
	35	6	0.143	5	0.155	4	0.179	4	0.163
	65	6	1.063	5	0.333	4	0.629	4	0.252
	165	6	0.895	5	0.613	4	1.019	4	0.758
	365	6	3.135	5	1.784	4	2.824	4	1.556
	665	6	4.059	5	3.419	4	3.441	4	2.681
	1065	6	7.439	5	6.685	4	6.497	4	4.961
5	5	-	-	5	0.075	-	-	7	0.054
	15	-	-	5	0.089	-	-	7	0.207
	35	-	-	5	0.280	-	-	7	0.382
	65	-	-	5	0.453	-	-	7	0.682
	165	-	-	5	0.982	-	-	7	1.340
	365	-	-	5	3.596	-	-	7	5.396
	665	-	-	5	6.012	-	-	7	7.350
	1065	-	-	5	8.088	-	-	7	10.247
6	5	6	0.042	5	0.067	4	0.035	4	0.047
	15	6	0.072	5	0.099	4	0.073	4	0.074
	35	6	0.138	5	0.162	4	0.108	4	0.142
	65	6	0.388	5	0.356	4	0.857	4	0.437
	165	6	0.767	5	0.669	4	0.519	4	0.514
	365	6	2.554	5	3.133	4	3.615	4	2.034
	665	6	6.492	5	3.839	4	4.833	4	4.276
	1065	6	8.312	5	6.393	4	5.578	4	5.774

Table 1 shows that our new method is comparable with the other methods. The numerical results in Table 1 clearly shows that the method have better results compared to CB and the TB in terms of number of iteration and in terms of the CPU time, it is competitive with all the compared methods. It is clear from the table that the other methods failed to solve some problems while our method solve 100% of the test problems which confirm the superiority(in terms of robustness) of our method over others, although the MSB is the most efficient of all the methods. We use the performance profile proposed by [11] in Figure 1 and it shows the results obtained are encouraging and competitive with the MSB method in terms of number of iterations. To better compare the numerical performance of the four methods, we used the comparison indices proposed by [15]. Table 2 shows that in terms of robustness, MT is superior to CB, TB, and MSB. The robustness of our method confirms the stability of our algorithm and thus less prone to error. The robustness of the solvers are: CB (67%), TB (67%), MSB (83%), MT (100%) of successes. Hence, these observations grant further authentication of the advantages of the proposed method to CB, TB and MSB for

Table 2. Robustness Indices

	CB	TB	MSB	MT
R	0.6667	0.6667	0.8333	1.0000

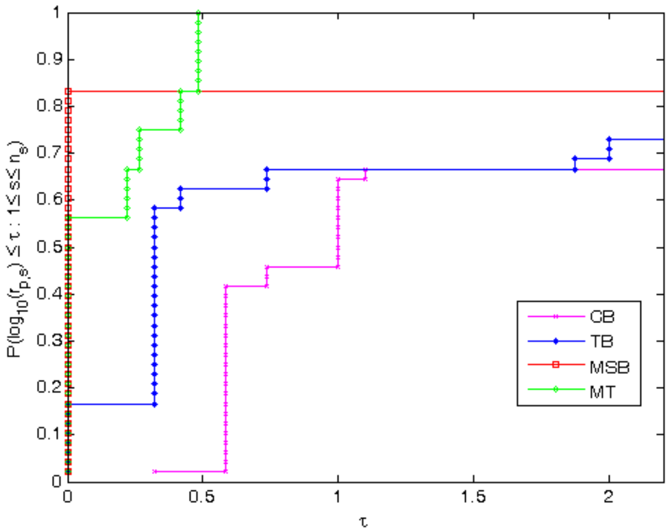

Figure 1. Performance profile for the compared methods in terms of the number of iterations

solving large scale systems of nonlinear equations.

4. Conclusion

We have shown that one can use the approach of approximating Jacobian matrices via the Midpoint and Trapezoidal quadrature formulas and obtained effective updates maintaining the local order of convergence of the classical Broyden method. Numerical experiments show a strong indication that the proposed Broyden-like method exhibit enhanced performance with respect to number of iterations in most of the tested problems. Using the comparison indices proposed by [15], it shows that the method is highly robust. However, we intend to extend the method in future to problems with much higher dimensions.

References

[1] A. Cordero, Juan R Torregrosa and M. P. Vassileva: *Pseudocomposition: a technique to design predictor-corrector methods for systems of nonlinear equations*, Appl Math Comput; 218: 11496-11504 (2012).

[2] A. Cordero and J. R. Torregrosa: *Variants of Newton's method for functions of several variables*, Appl Math Comput; 183: 199-208 (2006).

[3] A. Cordero and J. R. Torregrosa: *Variants of Newton's method using fifth-order quadrature formulas*, Appl. Math. Comput. 190, 686-698 (2007).

[4] A. Dhamacharoen *An efficient hybrid method for solving systems of nonlinear equations*, J Comput Appl Math; 263: 59-68 (2014).

[5] A. R. Soheili, S. A. Ahmadian, and J. Naghipoor: *A Family of Predictor-Corrector Methods Based on Weight Combination of Quadratures for Solving Nonlinear Equations*, International journal of nonlinear science, 6, 29-33 (2008).

[6] A. R. Vahidi, S. H. Javadi, and S. M. Khorasani: *Solving System of Nonlinear Equations by restarted Adomains method*, Journal of applied mathematical and computation, 6, 509-516 (2012).

[7] B. C. Shin, M. T. Darvishi, and C. H. Kim: *A comparison of the Newton-Krylov method with high order Newton-like methods to solve nonlinear systems*, Applied Mathematics and Computation, vol. 217, no. 7, pp. 31903198 (2010).

[8] B. Van De Rotten and S. V. Lunel: *A Limited Memory Broyden Method to Solve High-Dimensional Systems of Nonlinear Equations*, Technical Report MI 2003-2006. Leiden, the Netherlands: University of Leiden (2003).

[9] C. G. Broyden: *A class of methods for solving nonlinear simultaneous equations*, Math Comput; 19: 577-593 (1965).

[10] C. T. Kelly : *Iterative Methods for Linear and Nonlinear Equations*, Philadelphia, PA, USA: SIAM (1995).

[11] E. D. Dolan and J. J. More: *Benchmarking optimization software with performance profiles*, Math Program Ser A;91: 201-213 (2002).

[12] F. Awawdeh: *On new iterative method for solving systems of nonlinear equations*, Numer. Algor., 54: 395 (2010). http://dx.doi.org/10.1007/s11075-009-9342-8

[13] H. Jafari and V. D. Gejji : *Revised Adomian decomposition method for solving a system of nonlinear equations*, Journal of applied mathematical and computation, 175, 1-7 (2006).

[14] H. Mohammad and M. Y. Waziri: *On Broyden-like update via some quadratures for solving nonlinear systems of equations*, Turkish Journal of Mathematics, 39: 335-345 (2015).

[15] I. Bogle and J. Perkins: *A new sparsity preserving quasi-Newton update for solving nonlinear equations*. SIAM Journal on Scientific and Statistical Computing. 11(4): p. 621-630 (1990).

[16] J. H. He: *Homotopy perturbation technique*, Computer methods in applied mechanics and engineering, 178(3-4), 257-262 (1999). http://dx.doi.org/10.1016/S0045-7825(99)00018-3

[17] M. Mamat, K. Muhammad and M. Y. Waziri: *Trapezoidal Broydens Method for Solving Systems of Nonli-near Equations*, Applied Mathematical Sciences, 8, 251-260 (2014).

[18] M. T. Darvishi and B.Shin: *High-Order Newton-Krylov Methods to Solve Systems of Nonlinear Equations*, J. KSIAM, 15, 19-30 (2011).

[19] M. A. Hafiz and S. M. Bahgat: *An efficient two-step iterative method for solving system of nonlinear equations*, J Math Res; 4: 28-34 (2012).

[20] O. Oghovese and Emunefe O. John: *New Three-Steps Iterative Method for Solving Nonlinear Equations*, IOSR Journal of Mathematics (IOSR-JM) e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 10, Issue 5 Ver. IV, PP 44-47 (2014). www.iosrjournals.org

[21] S. Weerakoon and T.G.I. Fernando, *A variant of Newtons method with accelerated third-order convergence*, Applied Mathematics Letters 13 (8), 8793 (2000).