

Fourth Order Methods for Finding Multiple as well as Distinct Zeros of Non-linear Equations*

Nazir Ahmad Mir[†]

*Department of Mathematics, COMSATS Institute of Information Technology,
Plot No. 30, Sector H-8/1, Islamabad 44000, Pakistan*

and

Farooq Ahmad[‡], Muhammad Raza[§], Tahira Nawaz[¶]

*Centre for Advanced Studies in Pure and Applied Mathematics,
Bahauddin Zakariya University, Multan, Pakistan*

Received February 17, 2009, Accepted May 19, 2009.

Abstract

In this paper, we point out and analyse six two-step iterative methods for finding multiple as well as distinct zeros of non-linear equations. We prove that the methods for multiple zeros as well as for distinct zeros have fourth order convergence. The methods calculate the multiple as well as distinct zeros with high accuracy. The numerical tests show their better performance in case of algebraic as well as non-algebraic equations.

Keywords and Phrases: *Two-step iterative methods, Predictor-corrector methods, Convergence-Order, Multiple zeros, Distinct zeros, Non-linear equations.*

*2000 *Mathematics Subject Classification.* Primary 34A34, 65H05.

[†]Corresponding author. E-mail: nazirahmad.mir@gmail.com

[‡]E-mail: farooqgujar@gmail.com

[§]E-mail: mreza06r@gmail.com

[¶]E-mail: tahira.nawaz@gmail.com

1. Introduction

Finding the roots of a single variable non-linear equations efficiently, is a very interesting and old problem in numerical analysis and has many applications in engineering and other applied sciences.

We consider equation of the form

$$f(x) = 0, \quad (1.1)$$

where $f(x)$ is single variable non-linear function. Many one-step iterative methods for solving non-linear equations for distinct zeros have been developed using various techniques, see [1 – 7]. Recently, Ujevic [10] has suggested two-step method for solving non-linear equations for distinct zeros.

Motivated by the research going on in this direction, we present and analyze six two-step iterative methods for finding multiple as well as distinct zeros of non-linear equations. We prove that the methods for multiple zeros as well as the methods for distinct zeros have fourth order convergence. The methods calculate the multiple as well as the distinct zeros with high accuracy. The numerical tests show their better performance in case of algebraic as well as non-algebraic equations. First, we generalize the method due to Mamta et al. [7] to the case of multiple zero finding of non-linear equations and combine it with modified J. Chen and W. Li method for multiple zero finding of non-linear equations [2, 9] and classical Newton's method for multiple zeros of non-linear equations. Our results can be considered as generalization, improvement and refinement of the previously known results in the literature.

2. Two-step Iterative Methods for Multiple and Distinct Zeros of Non-linear Equations

We suggest the following generalization to Mamta et al. method to the case of multiple zero finding of non-linear equations:

Consider the Mamta et al. method [7] :

$$x_{n+1} = x_n - h(x_n),$$

where

$$h(x_n) = \frac{f(x_n)}{f'(x_n) + pf(x_n)}, \quad (2.1)$$

p is chosen here as positive or negative sign so as to make the denominator largest in magnitude. We observe that

$$\frac{f(x_n)}{f'(x_n)} = \frac{h(x_n)}{1 - ph(x_n)}.$$

The modified Mamta et al. method for multiple zeros can be written as

$$x_{n+1} = x_n - \alpha \frac{h(x_n)}{1 - ph(x_n)}, \quad (2.2)$$

where α is the multiplicity of the root and $h(x_n)$ is provided by (2.1). The relation (2.2) can further be written as:

$$\begin{aligned} x_{n+1} &= x_n \left(1 - \alpha \frac{h(x_n)}{x_n(1 - ph(x_n))} \right) \\ &\simeq x_n \exp\left(-\alpha \frac{h(x_n)}{x_n(1 - ph(x_n))}\right). \end{aligned} \quad (2.3)$$

Combining the method (2.3) with modified Chen Li method for multiple zeros [9], namely

$$x_{n+1} \simeq x_n \exp\left(-\alpha \frac{f(x_n)}{x_n f'(x_n)}\right), \quad (2.4)$$

we have therefore, the following two-step method for finding multiple zeros of non-linear equations:

$$\begin{aligned} z_n &= x_n \exp\left(-\frac{\alpha}{x_n} \frac{h(x_n)}{1 - ph(x_n)}\right), \\ x_{n+1} &= z_n \exp\left(-\alpha \frac{f(z_n)}{z_n f'(z_n)}\right). \end{aligned} \quad (2.5)$$

We propose the following three two-step algorithms for finding multiple zeros of non-linear equations:

Algorithm 1 (Chen-Li Mamta Method for Multiple Zeros(CLMM))

Step 1: For initial guess x_o , a tolerance $\varepsilon > 0$, and for iterations n , set $k = 0$.

Step 2: Calculate

$$z_k = x_k \exp\left(-\alpha \frac{f(x_k)}{x_k f'(x_k)}\right),$$

$$x_{k+1} = z_k \exp \left(-\frac{\alpha}{z_k} \frac{h(z_k)}{1 - ph(z_k)} \right), \quad (2.6)$$

where

$$h(z_k) = \frac{f(z_k)}{f'(z_k) + pf(z_k)}.$$

Step 3: If $|x_{k+1} - x_k| < \varepsilon$ or $k > n$ then stop.

Step 4: Set $k \rightarrow k + 1$ and go to **Step 2**.

Algorithm 2(Mamta Chen-Li Method for Multiple Zeros(MCLM))

Step 1: For initial guess x_o , a tolerance $\varepsilon > 0$, and for iterations n , set $k = 0$.

Step 2: Calculate

$$z_k = x_k \exp \left(-\frac{\alpha}{x_k} \frac{h(x_k)}{1 - ph(x_k)} \right),$$

where

$$h(x_k) = \frac{f(x_k)}{f'(x_k) + pf(x_k)},$$

$$x_{k+1} = z_k \exp \left(-\alpha \frac{f(z_k)}{z_k f'(z_k)} \right). \quad (2.7)$$

Step 3: If $|x_{k+1} - x_k| < \varepsilon$ or $k > n$ then stop.

Step 4: Set $k \rightarrow k + 1$ and go to **Step 2**.

Algorithm 3(Modified Mamta Newton Method for Multiple Zeros(MMNM))

Step 1: For initial guess x_o , a tolerance $\varepsilon > 0$, and for iterations n , set $k = 0$.

Step 2: Calculate

$$z_k = x_k \exp \left(-\frac{\alpha}{x_k} \frac{h(x_k)}{1 - ph(x_k)} \right),$$

where

$$h(x_k) = \frac{f(x_k)}{f'(x_k) + pf(x_k)}$$

$$x_{k+1} = z_k - \alpha \frac{f(z_k)}{f'(z_k)}. \quad (2.8)$$

Step 3: If $|x_{k+1} - x_k| < \varepsilon$ or $k > n$ then stop.

Step 4: Set $k \rightarrow k + 1$ and go to **Step 2**.

Taking $\alpha = 1$, gives us the following three two-step algorithms for finding distinct zeros of non-linear equations:

Algorithm 4 (Chen-Li Mamta Method for Distinct Zeros(CLMD))

Step 1: For initial guess x_o , a tolerance $\varepsilon > 0$, and for iterations n , set $k = 0$.

Step 2: Calculate

$$z_k = x_k \exp \left(-\frac{f(x_k)}{x_k f'(x_k)} \right), \quad (2.9)$$

$$x_{k+1} = z_k \exp \left(-\frac{1}{z_k} \frac{h(z_k)}{1 - ph(z_k)} \right)$$

Step 3: If $|x_{k+1} - x_k| < \varepsilon$ or $k > n$ then stop.

Step 4: Set $k \rightarrow k + 1$ and go to **Step 2**.

Algorithm 5 (Mamta Chen-Li Method for Distinct Zeros(MCLD))

Step 1: For initial guess x_o , a tolerance $\varepsilon > 0$, and for iterations n , set $k = 0$.

Step 2: Calculate

$$z_k = x_k \exp \left(-\frac{1}{x_k} \frac{h(x_k)}{1 - ph(x_k)} \right),$$

$$x_{k+1} = z_k \exp \left(-\frac{f(z_k)}{z_k f'(z_k)} \right). \quad (2.10)$$

Step 3: If $|x_{k+1} - x_k| < \varepsilon$ or $k > n$ then stop.

Step 4: Set $k \rightarrow k + 1$ and go to **Step 2**.

Algorithm 6 (Modified Mamta Newton Method for Distinct Zeros (MMND))

Step 1: For initial guess x_o , a tolerance $\varepsilon > 0$, and for iterations n , set $k = 0$.

Step 2: Calculate

$$z_k = x_k \exp \left(-\frac{1}{x_k} \frac{h(x_k)}{1 - ph(x_k)} \right),$$

$$x_{k+1} = z_k - \frac{f(z_k)}{f'(z_k)}. \quad (2.11)$$

Step 3: If $|x_{k+1} - x_k| < \varepsilon$ or $k > n$ then stop.

Step 4: Set $k \rightarrow k + 1$ and go to **Step 2**.

3. Convergence Analysis

In this section, we prove that algorithms 1, 2 and 3 as well as algorithms 4, 5 and 6 have convergence order four.

Theorem 1. *Let ω be the root of sufficiently differentiable function $f : (a, b) \subseteq \mathbb{R} \rightarrow \mathbb{R}$ with multiplicity α . If x_0 is sufficiently close to ω , then the two-step method defined by Algorithm 1 has fourth order convergence. The error equation in this case is given by:*

$$e_{n+1} = \frac{1}{8\alpha^4\omega^3}(\alpha^4 + 6\alpha^3\omega c_2 + 12\alpha^2\omega^2 c_2^2 + 8\alpha\omega^3 c_2^3 - 8p\alpha\omega^3 c_2^2 - 8p\alpha^2\omega^2 c_2 + 8p\omega^3 c_2^2 + 8p\alpha\omega^2 c_2 - 2p\alpha^3\omega)e_n^4 + O(e_n^5),$$

where $e_n = x_n - \omega$ and $c_j = \frac{f^{(\alpha+j-1)}(\omega)}{(\alpha+1)(\alpha+2)\dots(\alpha+j-1)f^{(\alpha)}(\omega)}$, $j = 2, 3, \dots$

Proof. The two-step method is given by

$$z_n = x_n \exp \left(-\alpha \frac{f(x_n)}{x_n f'(x_n)} \right), \quad (3.1)$$

$$x_{n+1} = z_n \exp \left(-\frac{\alpha}{z_n} \frac{h(z_n)}{1 - ph(z_n)} \right), \quad (3.2)$$

where

$$h(z_k) = \frac{f(z_n)}{f'(z_n) + pf(z_n)}.$$

Let ω be the root of multiplicity α , i.e. $f(\omega) = f'(\omega) = f''(\omega) = \dots = f^{(\alpha-1)}(\omega) = 0$ and $f^{(\alpha)}(\omega) \neq 0$.

By Taylor expansion of $f(x_n)$ about ω we get

$$\begin{aligned} f(x_n) &= \frac{f^{(\alpha)}(\omega)}{\alpha!} e_n^\alpha + \frac{f^{(\alpha+1)}(\omega)}{(\alpha+1)!} e_n^{\alpha+1} + \frac{f^{(\alpha+2)}(\omega)}{(\alpha+2)!} e_n^{\alpha+2} + O(e_n^{\alpha+3}) \quad (3.3) \\ &= \frac{f^{(\alpha)}(\omega)}{\alpha!} e_n^\alpha [1 + c_2 e_n + c_3 e_n^2 + c_4 e_n^3 + O(e_n^4)], \end{aligned}$$

where $e_n = x_n - \omega$ and $c_j = \frac{f^{(\alpha+j-1)}(\omega)}{(\alpha+1)(\alpha+2)\dots(\alpha+j-1)f^{(\alpha)}(\omega)}$, $j = 2, 3, \dots$.

Similarly, we obtain

$$f'(x_n) = \frac{f^{(\alpha)}(\omega)}{(\alpha-1)!} e_n^{\alpha-1} \left[1 + c_2 \frac{\alpha+1}{\alpha} e_n + c_3 \frac{\alpha+2}{\alpha} e_n^2 + O(e_n^3) \right]. \quad (3.4)$$

From (3.3) and (3.4), we get

$$\begin{aligned} \frac{f(x_n)}{f'(x_n)} &= \frac{1}{\alpha} [e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + O(e_n^5)] \\ &\quad \left[1 + c_2 \frac{\alpha+1}{\alpha} e_n + c_3 \frac{\alpha+2}{\alpha} e_n^2 + O(e_n^3) \right]^{-1} \\ &= \frac{e_n}{\alpha} - \frac{c_2}{\alpha^2} e_n^2 + \left(-2 \frac{c_3}{\alpha^2} + \frac{c_2^2}{\alpha^2} + \frac{c_2^2}{\alpha^3} \right) e_n^3 + O(e_n^4), \quad (3.5) \end{aligned}$$

and thus we have

$$\begin{aligned} -\alpha \frac{f(x_n)}{x_n f'(x_n)} &= \frac{e_n}{\alpha} + \frac{c_2}{\alpha^2} e_n^2 - \frac{1}{\alpha^3} (\alpha c_2^2 + c_2^2 - 2\alpha c_3) e_n^3 + \frac{1}{\alpha^4} (3\alpha^2 c_2 c_3 \\ &\quad + 4\alpha c_2 c_3 - 3\alpha^2 c_4 - \alpha^2 c_2^3 - 2\alpha c_2^3 - c_2^3) e_n^4 + O(e_n^5), \quad (3.6) \end{aligned}$$

implies

$$\begin{aligned} z_n &= x_n \exp \left(-\alpha \frac{f(x_n)}{x_n f'(x_n)} \right) \\ &= \omega + \frac{1}{2\alpha\omega} (2c_2\omega + \alpha) e_n^2 - \frac{1}{3\omega^2\alpha^2} (-3\alpha\omega^2 c_2^2 - 3\omega^2 c_2^2 + 6\alpha\omega^2 c_3 \\ &\quad - 3\alpha\omega c_2 - 6\alpha^2) e_n^3 + \frac{1}{8\omega^3\alpha^3} (8\alpha^2\omega^3 c_2^3 + 16\alpha\omega^3 c_2^3 - 24\alpha^2 c_2 c_3 \omega^3 \\ &\quad + 8\omega^3 c_2^3 + 8\alpha^2\omega^2 c_2^2 - 32\alpha\omega^3 c_2 c_3 + 24\alpha^2\omega^3 c_4 + 12\alpha\omega^2 c_2^2 \\ &\quad - 16\alpha^2\omega^2 c_3 + 12\alpha^2\omega c_2 + 7\alpha^3) e_n^4 + O(e_n^5). \quad (3.7) \end{aligned}$$

Now by Taylor expansion of $f(z_n)$ about ω we get

$$\begin{aligned} f(z_n) &= \frac{f^{(\alpha)}(\omega)}{\alpha!}(z_n - \omega)^\alpha + \frac{f^{(\alpha+1)}(\omega)}{(\alpha+1)!}(z_n - \omega)^{\alpha+1} \\ &\quad + \frac{f^{(\alpha+2)}(\omega)}{(\alpha+2)!}(z_n - \omega)^{\alpha+2} + O((z_n - \omega)^{\alpha+3}) \\ &= \frac{f^{(\alpha)}(\omega)}{\alpha!}(z_n - \omega)^\alpha [1 + c_2(z_n - \omega) + c_3(z_n - \omega)^2 + \\ &\quad c_4(z_n - \omega)^3 + O((z_n - \omega)^4)]. \end{aligned} \quad (3.8)$$

Similarly,

$$\begin{aligned} f'(z_n) &= \frac{f^{(\alpha)}(\omega)}{(\alpha-1)!}(z_n - \omega)^{\alpha-1} [1 + c_2 \frac{\alpha+1}{\alpha}(z_n - \omega) \\ &\quad + c_3 \frac{\alpha+2}{\alpha}(z_n - \omega)^2 + O((z_n - \omega)^3)]. \end{aligned} \quad (3.9)$$

From (3.8) and (3.9), we have

$$\begin{aligned} h(z_n) &= \frac{f(z_n)}{f'(z_n) + pf(z_n)} \\ &= \frac{1}{2\alpha^2\omega}(\alpha + 2\omega c_2)e_n^2 + \frac{1}{3\alpha^3\omega^2}(-3\alpha\omega c_2 + 6\alpha\omega^2 c_3 - 2\alpha^2 - 3\alpha\omega^2 c_2^2 \\ &\quad - 3\omega^2 c_2^2)e_n^3 + \frac{1}{8\alpha^5\omega^3}(24c_4\alpha^3\omega^3 - 32\alpha^2 c_2 c_3 + 8\alpha^3\omega^3 c_2^3 - 8p\omega^3 c_2^2 \\ &\quad + 10\alpha^3\omega c_2 + 7\alpha^4 - 24\alpha^3\omega^3 c_2 c_3 + 4\alpha^2\omega^2 c_2^2 - 16\alpha^3\omega^2 c_3 \\ &\quad + 16\alpha^2\omega^3 c_2^3 - 2p\alpha^2\omega - 8pc_2\alpha\omega^2)e_n^4 + O(e_n^5). \end{aligned} \quad (3.10)$$

Now, we have

$$\begin{aligned} \frac{h(z_n)}{1 - ph(z_n)} &= \frac{1}{2\alpha^2\omega}(2\omega c_2 + \alpha)e_n^2 + \frac{1}{3\alpha^3\omega^2}(-3\alpha\omega^2 c_2^2 + 6\alpha\omega^2 c_3 - 3\omega^2 c_2^2 \\ &\quad - 3\alpha\omega^2 c_2 - 2\alpha^2)e_n^3 + \frac{1}{8\alpha^5\omega^3}(8\alpha^3\omega^3 c_2^3 - 24\alpha^3\omega^3 c_3 c_2 \\ &\quad + 16\alpha^2\omega^3 c_2^3 + 8\alpha^3\omega^2 c_2^2 + 24\alpha^3\omega^3 c_4 - 16\alpha^3\omega^2 c_3 + 4\alpha^2\omega^2 c_2^2 \\ &\quad + 10\alpha^3\omega c_2 + 7\alpha^4 - 8c_2^2 p\omega^3 + 2p\alpha^3\omega - 2\alpha^2\omega^3 c_3 c_2 + 8pc_2\alpha^2\omega^4 \\ &\quad + 8p\alpha\omega^3 c_2^2 - 8p\alpha\omega^2 c_2 - 2p\alpha^2\omega)e_n^4 + O(e_n^5). \end{aligned} \quad (3.11)$$

Finally, by using (3.7) and (3.11) we have,

$$\begin{aligned} x_{n+1} &= z_n \exp\left(-\frac{\alpha}{z_n} \frac{h(z_n)}{1 - ph(z_n)}\right) \\ &= \omega + \frac{1}{8\alpha^4\omega^3}(\alpha^4 + 6\alpha^3\omega c_2 + 12\alpha^2\omega^2 c_2^2 + 8\alpha\omega^3 c_2^3 - 8p\alpha\omega^3 c_2^2 \\ &\quad - 8p\alpha^2\omega^2 c_2 + 8p\omega^3 c_2^2 + 8p\alpha\omega^2 c_2 - 2p\alpha^3\omega)e_n^4 + O(e_n^5). \end{aligned}$$

that is,

$$\begin{aligned} e_{n+1} &= \frac{1}{8\alpha^4\omega^3}(\alpha^4 + 6\alpha^3\omega c_2 + 12\alpha^2\omega^2 c_2^2 + 8\alpha\omega^3 c_2^3 - 8p\alpha\omega^3 c_2^2 - 8p\alpha^2\omega^2 c_2 \\ &\quad + 8p\omega^3 c_2^2 + 8p\alpha\omega^2 c_2 - 2p\alpha^3\omega)e_n^4 + O(e_n^5), \end{aligned}$$

showing that the algorithm has fourth order convergence.

Remark 2. Similarly, convergence order four of the second and third algorithms can also be determined.

Remark 3. Note that in case $\alpha = 1$, we have fourth order two-step method for finding distinct zeros of non-linear equations, given in the form of algorithm 4. Similar proofs follow for algorithms 5 and 6 to show that they also have fourth order convergence.

4. Numerical Tests

Here, we give some numerical examples to test and illustrate the performance of our new six two-step methods for finding multiple zeros as well as distinct zeros of non-linear equations. We name the classical one-step Newton method for multiple zeros as **NMM**. We consider here $\varepsilon = 1.0E - 64$ as given tolerance; n , the maximum number of iterations to be performed and $|x_{i+1} - x_i| < \varepsilon$ as stopping criterion. The computations are performed using Maple 10.0.

The following examples are used for numerical testing:

$$Ex.1 : f_1 = (x - 1)^{40} (x - 2)^{30} (x - 3)^{20} (x - 4)^{10} \quad [11]$$

$$Ex.2 : f_2 = [x - (0.3 + 0.6i)]^{100} [x - (0.1 + 0.7i)]^{200} [x - (0.7 + 0.5i)]^{300} [x - (0.3 + 0.4i)]^{10} \quad [11]$$

$$Ex.3 : f_3 = (x - 1)(e^{x-1} - 1) \quad [8]$$

$$Ex.4 : f_4 = f(x) = \ln(x) \quad [2]$$

Ex.5 : $f_5 = f(x) = \arctan(x)$ [2]

Ex 1:	Initial point	Exact root	Number of Iterations			
	x_0	x_n	MCLM	CLMM	MMNM	NMM
	0.6	1	3	3	3	7
	2.5	2	4	4	4	8
	3.5	3	4	4	4	8
	4.4	4	5	5	5	10

Table 4.1

Ex 2:	Initial point	Exact root	Number of Iterations			
	x_0	x_n	MCLM	CLMM	MMNM	NMM
	$0.301 + 0.601i$	$0.3 + 0.6i$	3	3	3	6
	$0.100 + 0.702i$	$0.1 + 0.7i$	3	3	3	6
	$0.702 + 0.498i$	$0.7 + 0.5i$	3	3	3	6
	$0.289 + 0.401i$	$0.3 + 0.4i$	3	3	3	6

Table 4.2

Ex 3:	Initial point	Exact root	Number of Iterations			
	x_0	x_n	MCLM	CLMM	MMNM	NMM
	1.5	1.0	4	4	3*	7

*Maximam Accuracy 34 digits

Table 4.3

We have tested our three methods , namely MCLM, CLMM, and MMNM for determining the multiple zeros of algebraic as well as non-algebraic equations on a sufficient number of examples to check their robustness and efficiency. We mention here, however, three examples. which are taken from [1,8,11]. Numerical results are provided in Tables 4.1 to 4.3. We compare our methods among themselves and also with one-step Newton method for multiple zeros. From Table 4.1, we, observe that, in general, the ill conditioned roots 1,2,3 and 4 of polynomial of degree 100 are calculated in 3 to 5 iterations with 64 digits accuracy by our proposed methods, whereas, Newton’s method takes double the number of iterations to reach the same accuracy. Table 4.2 describes the result for polynomial of degree 1000 with complex coefficients and complex roots with huge multiplicities. We obtained the roots

on third iteration getting 64 digits accuracy by our methods, whereas Newton's method takes double the number of iterations to reach the same accuracy. Table 4.3 gives the numerical results for multiple zeros of non-algebraic equations. In three to 4 iterations, we obtained the roots correct up to 64 digits accuracy, whereas, Newton's method takes double the number of iterations to reach the same accuracy.

Ex 4:

Initial point	Exact root	Number of Iterations					Formula 9
x_0	x_n	MCLD	CLMD	MMND	NM		
6	1	1	1	1	D		7
4	1	1	1	1	D		4
2	1	1	1	1	8		5
6.3	1	1*	1*	1**	D		D

* Maximam accuracy 58 digits and ** Maximam accuracy 61 digits

Table 4.4

Ex5:

Initial point	Exact root	Number of Iterations					Formula 9
x_0	x_n	MCLD	CLMD	MMND	NM		
2	1	4	4	4	6		7
1.8	1	4	4	4	6		6
1.4	1	4	3	4	5		5
2.4	1	4	4	4	D		D

Table 4.5

Similarly, we compare our methods, namely MCLD, CLMD and MMND with the one-step methods, namely the best method (Formula(9)) of Chen, and Newton method [2] for determining distinct zeros of non-linear equations. The numerical results are provided in Table 4.4. to 4.5 We observe that our proposed methods provide highly accurate results in less number of iterations, where as the method of Chen and Newton method diverge at some points as well.

5. Conclusion

We have established six new two-step iterative methods of order four for finding multiple and distinct zeros of algebraic as well as non-algebraic equations. In view of the numerical tests, it follows that our methods for multiple zeros are very effective in determining the highly multiple ill-conditioned zeros of real as well as complex polynomial equations with high accuracy. These are also very effective and efficient in determining multiple zeros of non-algebraic equations, which shows the robustness of the methods. Similarly our methods for distinct zeros provide highly accurate results in less number of iterations as compared to some well known existing methods.

References

- [1] R. L. Burden and J. D. Faires, *Numerical analysis*, PWS publishing company, Boston USA, 2001.
- [2] J. Chen and W. Li, On new exponential quadratically convergent iterative formulae, *Appl. Math. Comput.*, **180** (2006), 242-246.
- [3] C. Chun, Iterative methods Improving Newton's method by the decomposition method, *Int. J. Comp. Math. Appl.*, **50**(2005), 1559-1568.
- [4] M. Frontini and E. Sormani, Some variants of Newton's method with third order convergence and multiple roots, *J. Comput. Appl. Math.*, **156**(2003), 345-354.
- [5] M. Frontini and E. Sormani, Third order methods for quadrature formulae for solving system of non-linear equations, *Appl. Math. Comput.*, **149**(2004), 771-782.
- [6] M. V. Kanwar, V. K. Kukreja, and S. Singh, On a class of quadratically convergent iteration formulae, *Appl. Math. Comput.*, **166(3)**(2004), 633-637.

- [7] Mamta, V. Kanwar, V. K. Kukreja, and S. Singh, On some third order iterative methods for solving non-linear equations, *Appl. Math. Comput.*, **171**(2005), 272-280.
- [8] G. Wheatley, Applied Numerical Analysis, Pearson Education Inc.
- [9] N. A. Mir and Naila Rafiq, Third-order and fourth-order iterative methods for determining multiple and distinct zeros of non-linear equations, *Appl. Math. Comput.*, **190**(2007), 432-440.
- [10] N. Ujevic, A method for solving non-linear equations, *App. Math. Comput.*, **174**(2006), 416-426.
- [11] Z. Zeng, Computing multiple roots of inexact polynomials, *Mathematics of Computation*, **74 (250)**(2004), 869-903.