ASM Sc. J., 12, Special Issue 1, 2019 for IQRAC2018, 41-49

New Highly Accurate Iterative Method of Third Order Convergence for Finding The Multiple Roots of Nonlinear Equations

N. A. A. Jamaludin [1], N. M. A. Nik Long *[1,2], and F. Ismail [1,2]

[1]*Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia*
[2]*Institute for Mathematical Research, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia*

Corresponding author:nmasri@upm.edu.my

We present a new third order convergence iterative method for m multiple roots of nonlinear equation. The proposed method requires one evaluation of function and two evaluations of the first derivative of function. In numerical tests exhibit that the present method provides provides high accuracy numerical result as compared to other methods. The stability of the dynamical behaviour of iterative method is investigated by displaying the basin of attraction. Basin of attraction displays less black points which give us wider choices of initial guess in computation. **Keywords:** Basin of attraction, Multi-point iterative methods, Multiple roots, Nonlinear equations, Order of convergence.

I. Introduction

Solving nonlinear equation accurately is one of the most important task in numerical analysis. The well-known Newton's method for finding multiple roots, x^* is written as

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

where m is multiplicity of roots. This method was developed by Schröder and Stewart (1998), which converges quadratically. In recent years, some modification of Newton's method for multiple roots with third order convergence have been developed and analyzed. Examples are Dong (1982), Dong (1987), Ferrara et al. (2015), Heydari et al. (2010), Homeier (2009), Sharifi et al. (2015) and Victory Jr and Neta (1983). All of those methods require the multiplicity, m, to be known. Parida and Gupta (2008), Soleymani and Babajee (2013) and Yun (2009) developed the iterative methods with the unknown m. Chun et al. (2009), Hansen and Patrick (1976), Neta (2008) and Osada (1994) developed the iterative methods which require the evaluation of second derivative of function.

In this paper we deal with the iterative methods of third order convergence to find the multiple roots x^*, of non-linear equation, with known multiplicity, m. Our proposed method is free from the evaluation of second derivative, f'' of function.

II. Construction of method

Osada (1994) proposed the third order one-point iterative method for finding the multiple roots of nonlinear equation, $f(x) = 0$ as

$$x_{k+1} = x_k - \frac{1}{2}m(m+1)\frac{f(x_k)}{f'(x_k)} + \frac{1}{2}(m-1)^2\frac{f'(x_k)}{f''(x_k)}. \quad (2)$$

Consider a Newton-type iterative method for multiple roots as

$$y_k = x_k - \frac{2m}{m+2}\frac{f(x_k)}{f'(x_k)}, \quad (3)$$

where m is the multiplicity of multiple roots. With the help of Taylor's series, we obtain

$$f''(x_k) \cong \frac{(m+2)f'(x_k)\left[f'(x_k)-f'(y_k)\right]}{2mf(x_k)}. \quad (4)$$

Substitute (4) into (2) and incorporate with (3), we have

$$\begin{cases} y_k &= x_k - \dfrac{2m}{m+2}\dfrac{f(x_k)}{f'(x_k)}, \\ x_{k+1} &= x_k - \dfrac{m(m+1)f(x_k)}{2f'(x_k)} \\ & +\dfrac{(m-1)^2 m f(x_k)}{(m+2)\left(f'(x_k)-f'(y_k)\right)}. \end{cases} \quad (5)$$

The scheme in (5) is not in stable state as third order convergence. In order to archive the stable third order convergence, we write

$$\begin{cases} y_k = & x_k - \dfrac{2m}{m+2}\dfrac{f(x_k)}{f'(x_k)}, \\ x_{k+1} = & x_k - \dfrac{m(m+1)f(x_k)}{\alpha f'(x_k)} \\ & +\dfrac{\beta(m-1)^2 m f(x_k)}{(m+2)\left(f'(x_k)-f'(y_k)\right)}, \end{cases} \quad (6)$$

where α and β are free disposable parameters. For simple root case, where $m = 1$, method (6) becomes one point iterative Newton's method, with the first step is void.

III. Convergence Analysis

In this section, we describe a choice of parameters α and β in order to get the third-order convergence for our scheme (6).

Theorem 1. *Let $x^* \in D$ be a multiple roots of a sufficiently smooth function $f : D \subseteq R \to R$ defined on an open interval D with the multiplicity $m > 1$, which includes x_0 as an initial approximation of x^*. Then, the iterative methods defined by (6) has third order convergence when*

$$\begin{cases} \alpha &= -\frac{2m^m(m+1)}{-m^2(m+2)^m+m^m(-4+m(m+2))}, \\ \beta &= \frac{m^{-m}(m+2)^{1-m}(m^m(m+2)-m(m+2)^m)^2}{4(m-1)^2}, \end{cases}$$

with the error term $e_{n+1} = \frac{2(m^m-(q)^m)c_1^2 e_n^3}{m^q q - m^3 q^m} + O(e_n^4)$.

Proof. Let $e_n := x_n - x^*, e_{n,y} := y_n - x^*, c_i := \frac{m!}{(m+i)!}\frac{f^{m+i}(x^*)}{f^m(x^*)}, c_0 = 1, p = m+1, q = m+2, r = m-1$. Using the fact that $f(x^*) = 0$, Taylor expansion of f at x^* yields

$$f(x_n) = e_n^m\left(1 + c_1 e_n + c_2 e_n^2 + c_3 e_n^3\right) + O(e_n^4) \quad (7)$$

and

$$f'(x_n) = e^r(m + e_n p c_1 + e_n^2(q)c_2 + e_n^3\left((m+3)c_3 + O(e_n^4)\right)). \quad (8)$$

Thus

$$e_{n,y} = y_n - x^* = \frac{me_n}{q} + \frac{2c_1 e_n^2}{m(q)} + \frac{\left(-2(p)c_1^2 + 4mc_2\right)e_n^3}{m^2(q)} + O(e_n^4). \quad (9)$$

For $f(y_n)$ we have

$$f(y_n) = e_{n,y}^m\left(1 + c_1 e_{n,y} + c_2 e_{n,y}^2 + c_3 e_{n,y}^3\right) + O(e_{n,y}^4). \quad (10)$$

Substituting (7)-(10) into (6) gives the error term as

$$e_{n+1} = D_1 e_n + D_2 e_n^2 + D_3 e_n^3 + O(e_n^4),$$

where

$$D_1 = 1 - \frac{1}{2}m\left(\frac{1+\frac{1}{m}}{\alpha} + \frac{2r^2\beta}{mq - m^m q^{2-m}}\right), \quad (11)$$

$$D_2 = \frac{1}{2}m\Big(-\frac{p}{m\alpha} + \frac{p^2}{m^2\alpha} - \frac{2r^2\beta}{mq - m^m q^{2-m}} + \frac{2r^2 q^r\left((m^2 p)q^m + m^m(4 - mpq)\right)\beta}{m^2\left(m^m q - mq^m\right)^2}\Big)c_1, \quad (12)$$

$$D_3 = \frac{c_1^2 V + 2c_2 m\left(m^m q - mq^m\right)S}{2m^2 q\left(m^m q - mq^m\right)^3\alpha}, \quad (13)$$

$$V = -m^{3m}p^2q^4 + m^3pq^{3m}(2+3m+m^2 + 2r^2\alpha\beta) - m^mq^{2m}(3m^2p^2q^2 + 4r^2(-4 - 2m + 3m^3 + m^4)\alpha\beta) + m^{2m}q^m(3mp^2q^3 + 2r^2(-16 + m(-8 + mp(4+m)))\alpha\beta)$$

and

$$S = m^{2m}pq^3 + m^2q^{2m}\left(2+3m+m^2+2r^2\alpha\beta\right) - 2m^mq^m\left(mpq^2 + r^2(-4+mq)\alpha\beta\right).$$

To obtain the third convergence order, it is necessary to choose $D_i = 0$ $(i = 1, 2)$, which yield

$$\alpha = -\frac{2m^mp}{-m^2q^m + m^m(-4+mq)}$$

and

$$\beta = -\frac{m^{-m}q^r\left(m^mq - mq^m\right)^2}{4r^2},$$

and the error terms becomes

$$e_{n+1} = \frac{2\left(m^m - (q)^m\right)c_1^2e_n^3}{m^qq - m^3q^m} + O(e_n^4)$$

which completes the proof. □

IV. Numerical Analysis

Consider the following iterative methods of third order convergence

1. Chun et al.'s method (CBN) in (Chun et al., 2009) is given by

$$x_{k+1} = x_k - \frac{m\big((2\gamma-1)m + 3 - 2\gamma\big)}{2}\frac{f(x_k)}{f'(x_k)} + \frac{\gamma(m-1)^2}{2}\frac{f'(x_k)}{f''(x_k)} - \frac{m^2(1-\gamma)}{2}\frac{f(x_k)^2f''(x_k)}{(f'(x_k))^3}, \quad (14)$$

where $\gamma = -1$.

2. Dong's method (DM) in (Dong, 1987) is given by

$$\begin{cases} y_k = & x_k - \dfrac{f(x_k)}{f'(x_k)}, \\ x_{k+1} = & y_k - \dfrac{f(x_k)}{\gamma f'(y_k) + \lambda f'(x_k)}, \end{cases} \quad (15)$$

where $\gamma = \left(\frac{m}{m-1}\right)^{m+1}$ and $\lambda = \frac{m-m^2-1}{(m-1)^2}$.

3. Ferrara et al.'s method (FSS) in (Ferrara et al., 2015) is given by

$$\begin{cases} y_k & = \dfrac{f(x_k)}{f'(x_k)}, \\ x_{k+1} & = x_k - \dfrac{\theta f(x_k)}{\theta f(x_k) - f(y_k)}\dfrac{f(x_k)}{f'(x_k)}, \end{cases} \quad (16)$$

where $\theta = \left(\dfrac{-1+m}{m}\right)^{-1+m}$.

4. Osada's method (OS) in (Osada, 1994) is given by

$$x_{k+1} = x_k - \frac{1}{2}m(m+1)\frac{f(x_k)}{f'(x_k)} + \frac{1}{2}(m-1)^2\frac{f'(x_k)}{f''(x_k)}. \quad (17)$$

5. Victory and Neta's method (VN) in (Victory Jr and Neta, 1983) is given by

$$\begin{cases} y_k = & x_k - \dfrac{f(x_k)}{f'(x_k)}, \\ x_{k+1} = & y_k - \dfrac{f(y_k)}{f'(x_k)}\dfrac{f(x_k) + Af(y_k)}{f(x_k) + Bf(y_k)}, \end{cases} \quad (18)$$

where $A = \mu^{2m} - \mu^{m+1}$, $B = -\dfrac{\mu^m(m-2)(m-1)+1}{(m-1)^2}$ and $\mu = \dfrac{m}{m-1}$.

Test functions listed in Table 1 are used for the numerical experiments. Table 2 exhibits that our proposed method provides a high accuracy numerical results. It is noticed that error per iteration is much smaller as compared to other existing methods (equations (14)-(18)) for the same number of iterations. Note that the computational order of convergence (COC) and approximated computational order of convergence (ACOC) are defined, respectively, as (Grau-Sánchez et al., 2010)

$$COC \approx \frac{\ln |(x_{n+1} - x^*)/(x_n - x^*)|}{\ln |(x_n - x^*)/(x_{n-1} - x^*)|} \tag{19}$$

and

$$\text{ACOC} \approx \frac{\ln |(x_{n+1} - x_n)/(x_n - x_{n-1})|}{\ln |(x_n - x_{n-1})/(x_{n-1} - x_{n-2})|}. \tag{20}$$

V. Basin of attraction

In this section we investigate the stability of the dynamical behaviour of iterative methods by utilizing the basin of attraction in complex plane. Let a square $D \subset \mathbb{C}$ and we choose the initial guess, $z_0 \in D$. We assign the grid 300×300 point and set the square D with $[-3.0, 3.0] \times [-3.0, 3.0]$. In basin of attraction we assign the different colour to different roots. The intensity of colour corresponding to the number of iteration needed to converge; region with the brighter colour require less number of iterations to converge to the roots, x^* as compared to the darker colour. If the initial guess x^* is chosen in the black region, the numerical convergence will not be archived even after 100 iterations with minimum tolerance $> 10^{-3}$. Table 3 shows the list of test functions with the complex roots used to generate basin of attraction.

Figures 1-5 show the comparison of the dynamical behaviour of the iterative methods listed in Table IV.. Its clearly seen that our newly developed iterative method provides less black points and larger brighter region as compared to others, which means the choice of initial guess is vast.

Conclusion

We developed a new iterative method to obtain multiple roots of the nonlinear equation. The proposed method requires one evaluations of function and two evaluation of the first derivative of function, and it is free from second derivative. Numerical performances exhibit that our method provides faster convergence and higher accuracy as compared to other methods with the same order of convergence. Basin of attraction displays that the proposed scheme contains less black points which gives us more choices of initial guess.

Acknowledgements

This project is supported by the Universiti Putra Malaysia under Putra Grant vot no: 9567900

References

[1] Changbum Chun, Hwa Ju Bae, and Beny Neta. New families of nonlinear third-order solvers for finding multiple roots. *Computers and Mathematics with Applications*, 57(9):1574–1582, 2009.

[2] C Dong. A basic theorem of constructing an iterative formula of the higher order for computing multiple roots of an equation. *Mathematics Numerical Sinica*, 11: 445–450, 1982.

[3] Chen Dong. A family of multipoint iterative functions for finding multiple roots of equations. *International Journal of Computer Mathematics*, 21(3-4):363–367, 1987.

[4] Massimiliano Ferrara, Somayeh Sharifi, and Mehdi Salimi. Computing multiple zeros by using a parameter in Newton-Secant method. *SeMA Journal*, pages 1–9, 2015.

[5] Miquel Grau-Sánchez, Miquel Noguera, and José Manuel Gutiérrez. On some computational orders of convergence. *Applied Mathematics Letters*, 23(4):472–478, 2010.

[6] Eldon Hansen and Merrell Patrick. A family of root finding methods. *Numerische Mathematik*, 27(3):257–269, 1976.

[7] M Heydari, S M Hosseini, and G B Loghmani. Convergence of a family of third-order methods free from second derivatives for finding multiple roots of nonlinear equations. *World Applied Sciences Journal*, 11(5):507–512, 2010.

[8] Herbert H H Homeier. On Newton-type methods for multiple roots with cubic convergence. *Journal of Computational and Applied Mathematics*, 231(1): 249–254, 2009.

[9] B Neta. New third order nonlinear solvers for multiple roots. *Applied Mathematics and Computations*, 202(1):162–170, 2008.

[10] Naoki Osada. An optimal multiple root-finding method of order three. *Journal of Computational and Applied Mathematics*, 51(1):131–133, 1994.

[11] P.K. Parida and D.K. Gupta. An improved method for finding multiple roots and it's multiplicity of nonlinear equations in r. *Applied Mathematics and Computation*, 202(2): 498 – 503, 2008. ISSN 0096-3003. doi: https://doi.org/10.1016/j.amc.2008.02.030. URL `http://www.sciencedirect.com/science/article/pii/S0096300308001240`.

[12] E Schröder and G W Stewart. On infinitely many algorithms for solving equations. *Translation of paper by E. Schröder*, UMIACS-TR-92-121, 1998.

[13] S Sharifi, M Ferrara, N M A Nik Long, and M Salimi. Modified Potra-Pták method to determine the multiple zeros of nonlinear equations. *arXiv preprint arXiv:1510.00319*, 2015.

[14] F Soleymani and D K R Babajee. Computing multiple zeros using a class of quartically convergent methods. *Alexandria Engineering Journal*, 52(3):531–541, 2013.

[15] H D Victory Jr and B Neta. A higher order method for multiple zeros of nonlinear functions. *International Journal of Computer Mathematics*, 12(3-4):329–335, 1983.

[16] Beong In Yun. A derivative free iterative method for finding multiple roots of nonlinear equations. *Applied Mathematics Letters*, 22(12): 1859 – 1863, 2009. ISSN 0893-9659. doi: https://doi.org/10.1016/j.aml.2009.07.013. URL `http://www.sciencedirect.com/science/artic`

Table 1: List of Test Functions

Test functions f_n	Root x^*	Multiplicity m
$f_1(x) = (\sin^2 x + x)^5$	0	5
$f_2(x) = (\ln(1 + x^2) + e^{x^2-3x} \sin x)^6$	0	6
$f_3(x) = (x^3 + \ln(1 + x))^7$	0	7
$f_4(x) = (x^6 - 8)^2 \ln(x^6 - 7)$	$\sqrt{2}$	3
$f_5(x) = (\ln(x^3 - x + 1) + 4 \sin x - 1)^{10}$	1	10

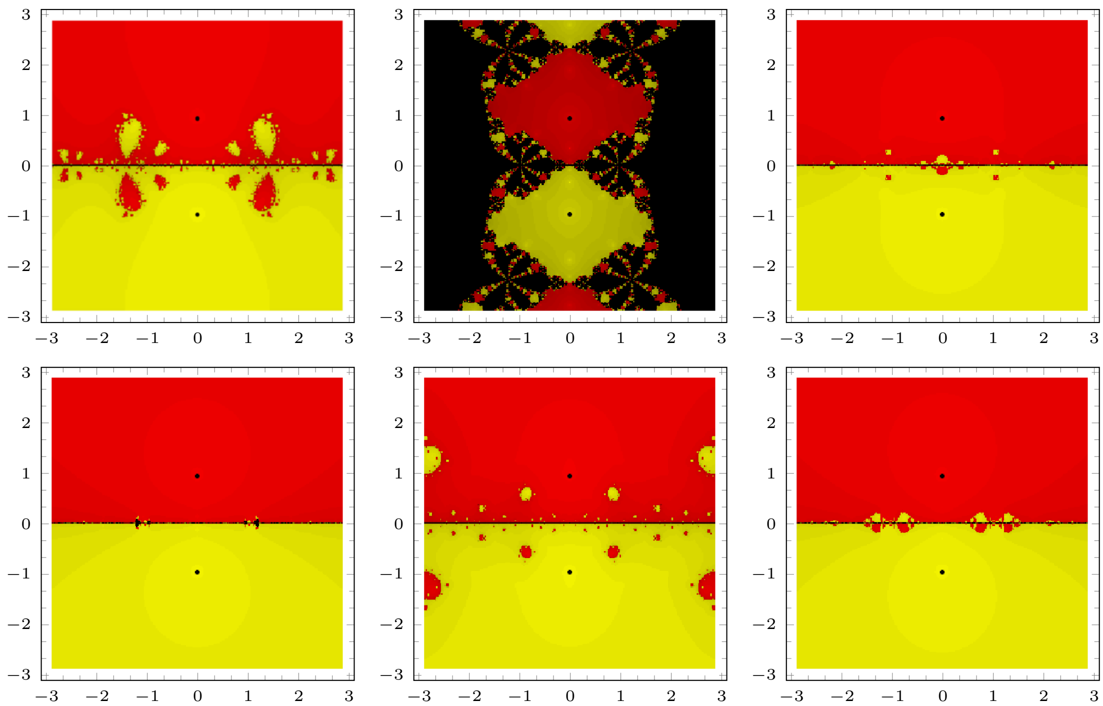

Figure 1: Basin of attraction of Present Method, CBN, DM, FSS, OS, VN for test function $p_1(z)$.

Table 2: Error, COC and ACOC of iterative methods

Methods	Method(6)	CBN(14)	DM(15)	FSS(16)	OS(17)	VN(18)
$f_1, x_0 = 0.1$						
$\|x_1 - x^*\|$	$0.482e^{-3}$	$0.113e^{-2}$	$0.420e^{-3}$	$0.740e^{-3}$	$0.163e^{-2}$	$0.820e^{-3}$
$\|x_2 - x^*\|$	$0.494e^{-10}$	$0.215e^{-8}$	$0.314e^{-10}$	$0.364e^{-9}$	$0.107e^{-7}$	$0.565e^{-9}$
$\|x_3 - x^*\|$	$0.531e^{-31}$	$0.150e^{-25}$	$0.132e^{-31}$	$0.434e^{-28}$	$0.304e^{-23}$	$0.186e^{-27}$
COC	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000
ACOC	3.0001	2.9998	3.0000	2.9999	2.9994	2.9999
$f_2, x_0 = 0.3$						
$\|x_1 - x^*\|$	$0.114e^{-1}$	$0.623e^{-1}$	$0.479e^{-1}$	$0.564e^{-1}$	$0.600e^{-1}$	$0.562e^{-1}$
$\|x_2 - x^*\|$	$0.219e^{-5}$	$0.665e^{-3}$	$0.116e^{-3}$	$0.178e^{-4}$	$0.122e^{-2}$	$0.111e^{-3}$
$\|x_3 - x^*\|$	$0.151e^{-16}$	$0.320e^{-9}$	$0.223e^{-11}$	$0.437e^{-14}$	$0.784e^{-8}$	$0.471e^{-12}$
COC	3.0000	3.0011	3.0001	3.0000	3.0002	2.9997
ACOC	3.0040	3.2118	2.9490	2.7447	3.0858	3.0946
$f_3, x_0 = 0.2$						
$\|x_1 - x^*\|$	$0.644e^{-3}$	$0.104e^{-1}$	$0.781e^{-2}$	$0.925e^{-2}$	$0.100e^{-1}$	$0.921e^{-2}$
$\|x_2 - x^*\|$	$0.201e^{-10}$	$0.967e^{-6}$	$0.376e^{-6}$	$0.702e^{-6}$	$0.682e^{-6}$	$0.673e^{-6}$
$\|x_3 - x^*\|$	$0.621e^{-33}$	$0.830e^{-18}$	$0.425e^{-19}$	$0.316e^{-18}$	$0.238e^{-18}$	$0.270e^{-18}$
COC	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000
ACOC	2.9995	2.9934	2.9986	2.9972	2.9894	2.9968
$f_4, x_0 = 1.5$						
$\|x_1 - x^*\|$	$0.951e^{-3}$	$0.422e^{-2}$	$0.221e^{-2}$	$0.329e^{-2}$	$0.418e^{-2}$	$0.329e^{-2}$
$\|x_2 - x^*\|$	$0.113e^{-7}$	$0.592e^{-5}$	$0.329e^{-6}$	$0.163e^{-5}$	$0.248e^{-5}$	$0.147e^{-5}$
$\|x_3 - x^*\|$	$0.180e^{-22}$	$0.129e^{-13}$	$0.970e^{-18}$	$0.171e^{-15}$	$0.492e^{-15}$	$0.115e^{-15}$
COC	3.0000	3.0000	3.0000	3.0000	3.000	3.0000
ACOC	3.0058	3.0339	3.0134	3.0207	3.0067	3.0186
$f_5, x_0 = 1.2$						
$\|x_1 - x^*\|$	$0.552e^{-4}$	$0.182e^{-2}$	$0.146e^{-2}$	$0.164e^{-2}$	$0.183e^{-2}$	$0.164e^{-2}$
$\|x_2 - x^*\|$	$0.350e^{-15}$	$0.171e^{-8}$	$0.688e^{-9}$	$0.109e^{-8}$	$0.175e^{-8}$	$0.109e^{-8}$
$\|x_3 - x^*\|$	$0.892e^{-49}$	$0.141e^{-26}$	$0.719e^{-28}$	$0.322e^{-27}$	$0.155e^{-26}$	$0.327e^{-27}$
COC	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000
ACOC	3.0001	2.9998	2.9999	2.9999	2.9998	2.9999

Table 3: List of test functions and their roots

Test problem $p_n(z)$	Root x^*
$p_1(z) = (z + \frac{1}{z})^5$	$\pm i$
$p_2(z) = (z^3 - 1)^{10}$	$1, \ -0.5 \pm 0.866025i$
$p_3(z) = (z^3 + 1)^3$	$-1, \ 0.5 \pm 0.866025i$
$p_4(z) = (2z^4 - z)^8$	$0, \ -0.39685 \pm 0.687365i,$ 0.793701
$p_5(z) = (z^5 - z^2 + 1)^{15}$	$-0.808731,$ $-0.464912 \pm 1.07147i,$ $0.869278 \pm 0.388269i$

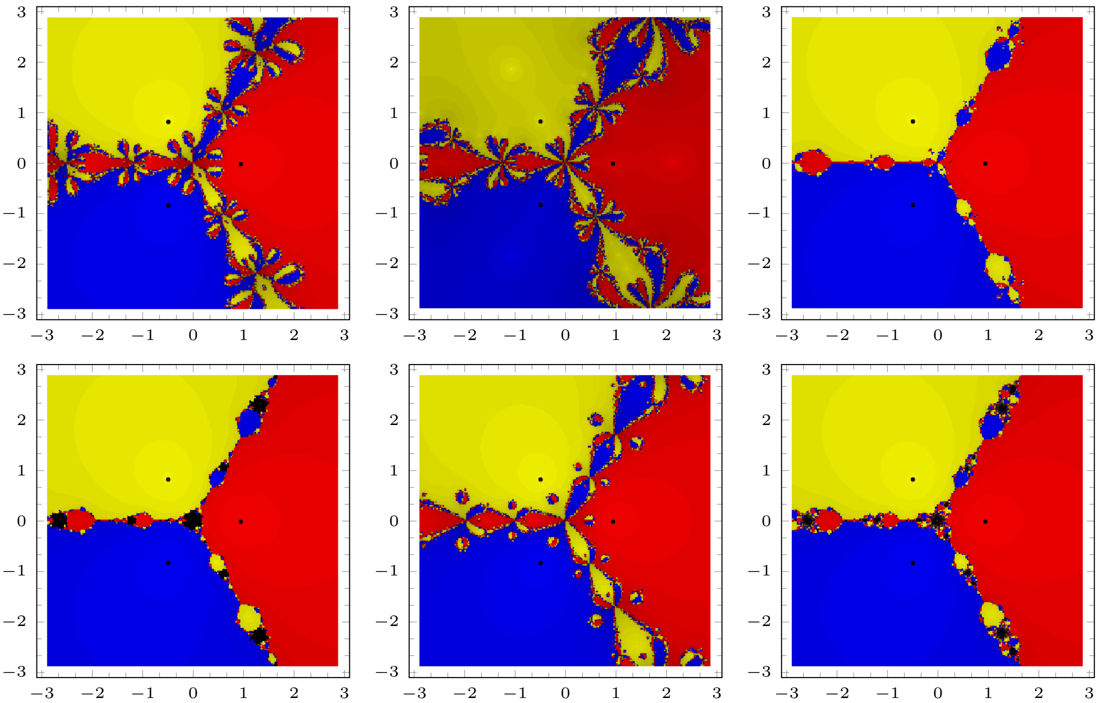

Figure 2: Basin of attraction of Present Method, CBN, DM, FSS, OS, VN for test function $p_2(z)$.

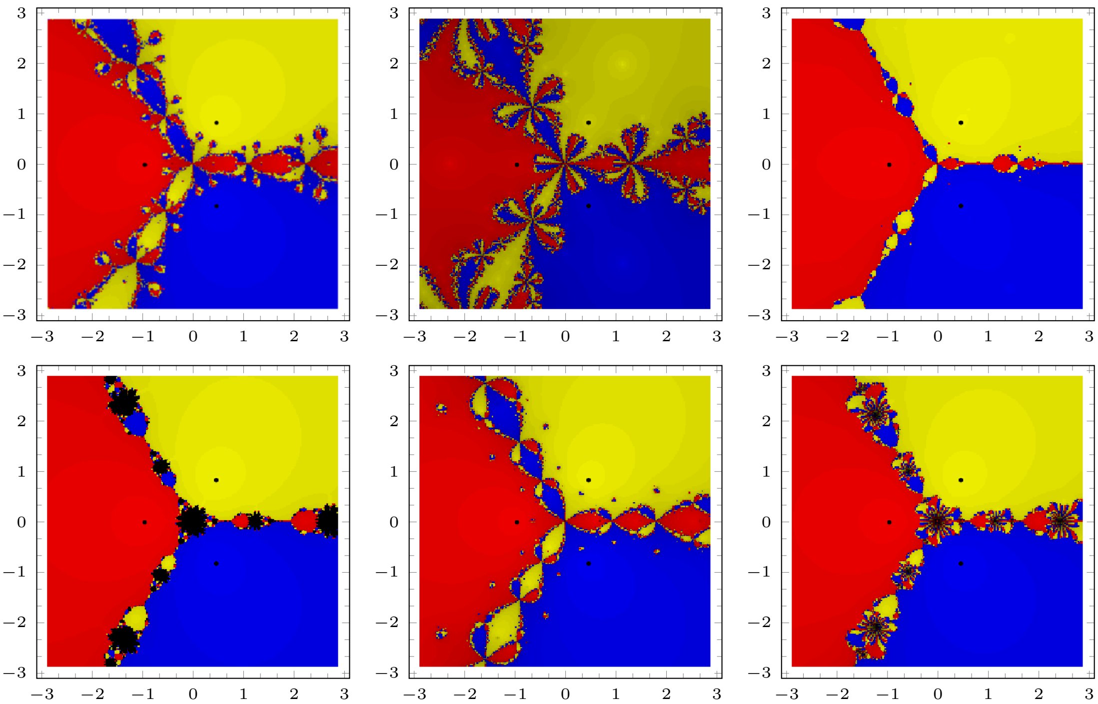

Figure 3: Basin of attraction of Present Method, CBN, DM, FSS, OS, VN for test function $p_3(z)$.

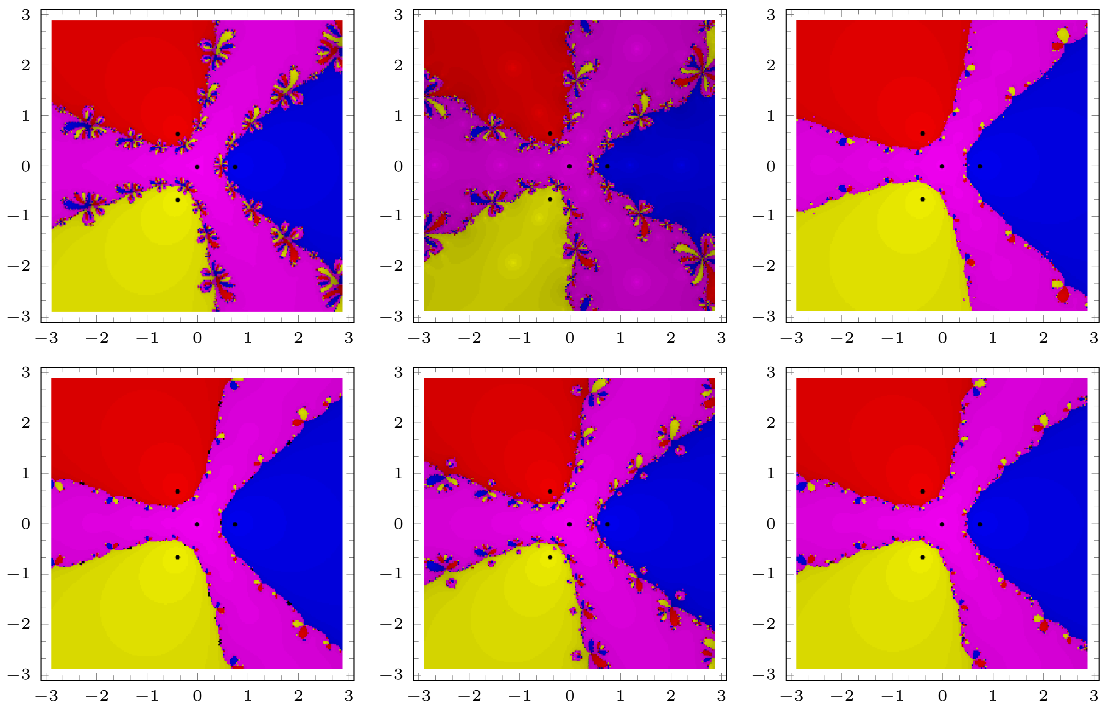

Figure 4: Basin of attraction of Present Method, CBN, DM, FSS, OS, VN for test function $p_4(z)$.

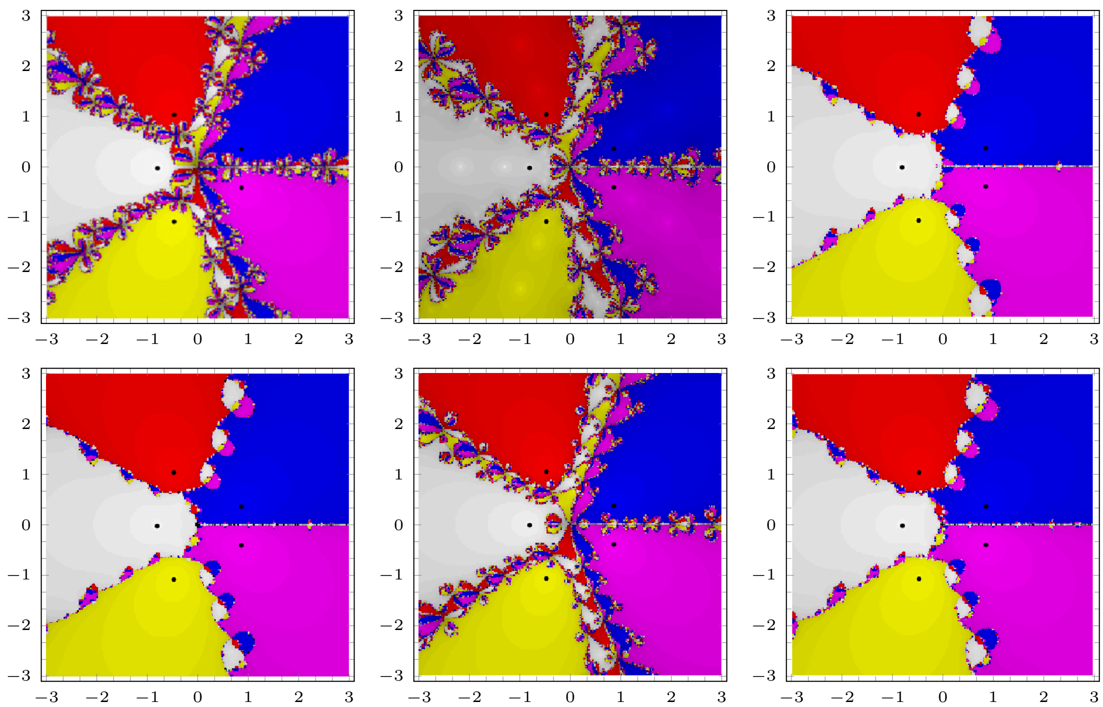

Figure 5: Basin of attraction of Present Method, CBN, DM, FSS, OS, VN for test function $p_5(z)$.