JOURNAL OF NUMERICAL ANALYSIS AND APPROXIMATION THEORY
J. Numer. Anal. Approx. Theory, vol. 51 (2022) no. 1, pp. 88–102, http://doi.org/10.33993/jnaat511-1248
ictp.acad.ro/jnaat

BASINS OF ATTRACTION FOR FAMILY OF POPOVSKI'S METHODS AND THEIR EXTENSION TO MULTIPLE ROOTS

BENY NETA*

Abstract. In this paper we revisit Popovski's family of methods for simple roots. We compare several members using basins of attraction visually and qualitatively by comparing the run-time on several examples, the average number of iterations and the number of divergent points. We chose five different members of the family. We also develop an equivalent family of methods for multiple roots and compare several members on six different numerical examples.

MSC. 65H05, 37D99.
Keywords. Nonlinear equations, simple roots, multiple roots, Popovski's method.

1. INTRODUCTION

The literature for solving a single nonlinear equation is vast, see Traub [1] and the more recent book by Petković *et al.* [2]. This is a fundamental problem in many areas of engineering and science. Here we are interested in analyzing the one-step family of third order methods for simple roots due to Popovski [3]. The method is given by

$$z_{k+1} = z_k - \tfrac{(1-r)f'(z_k)}{f''(z_k)} \left\{ \left[1 - \tfrac{r}{r-1} \tfrac{f(z_k)f''(z_k)}{(f'(z_k))^2} \right]^{1/r} - 1 \right\} \tag{1}$$

where the parameter $r \neq 1$. This family includes Chebyshev's method [1] (when $r = 1/2$), Halley's method [4] (when $r = -1$), Euler-Cauchy's method see [5] or [6] (when $r = 2$). At the limit as $r \to 1$ the method becomes the well known Newton's method of order only two. See also Neta [7].

Halley's method was compared to various schemes available in the literature in Scott *et al.* [8], Neta *et al.* [9], [10] and [11]. Neta and Chun [10] has shown that Halley's method has no extraneous fixed points, *i.e.* fixed points of the iteration that are **not** zeros of the function $f(z)$ (see, Vrcsay and Gilbert [12]). It is easy to show that all members of Popovski's family of methods have no extraneous fixed points. It was shown that Halley's method performed better than Chebyshev's and Euler-Cauchy's method. Therefore in the sequel we only

*Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA 93943, USA, e-mail: bneta@nps.edu.

compare Halley's method to other members not considered in the literature. Recently, Herceg and Petković [13] analyzed Popovski's family for $r = -1$ (Halley) and $r = -2$. They concluded that overall HM and Pm2 are better than Chebyshev and Euler-Cauchy. They also developed an equivalent family for multiple roots when the multiplicity is known in advance.

In this paper we consider also the cases $r = -3$, $r = -4$ and $r = -8$. We also develop a family for multiple roots that does **not** require the knowledge of the multiplicity. Instead we use the idea that the quotient $\frac{f}{f'}$ must have simple roots at the points that f has multiple roots. The use of such function was mentioned in [1], [2] and [14]. several third order methods for multiple roots were compared in [15]. Those methods all assume the knowledge of the multiplicity. See also [16] on Laguerre family of methods which tend to Halley's method for multiple roots when the parameter tends to zero.

2. CORRESPONDING CONJUGACY MAPS FOR QUADRATIC POLYNOMIALS

Given two maps ϕ and ψ from the Riemann sphere into itself, an analytic conjugacy between the two maps is a diffemorphism H from the Riemann sphere onto itself such that $H \circ \phi = \psi \circ H$. Here we consider only quadratic polynomials. For more details, see Amat *et al.* [17] and Beardon [18].

Theorem 1. *For a rational map $R_m(z)$ arising from Popovski's family of methods applied to $q(z) = (z-a)(z-b)$, $a \neq b$, $R_m(z)$ is conjugate via the Möbius transformation given by $M(z) = \frac{z-a}{z-b}$ to*

$$S(z) = \frac{((z+1)m + 3z + 1)A(z) - (z+1)(m+1)}{((z+1)m + z + 3)A(z) - (z+1)(m+1)}.$$

where

$$A(z) = \left(\frac{(m+1)z^2 + 2z + m + 1}{(m+1)(z+1)^2} \right)^{1/m}$$

Proof. Let $q(z) = ((z-a)(z-b))$, $a \neq b$ and let M be the Möbius transformation given by $M(z) = \frac{z-a}{z-b}$ with its inverse $M^{-1}(u) = \frac{ub-a}{u-1}$, which may be considered as a map from $\mathbb{C} \cup \{\infty\}$. We then have

$$S(u) = M \circ R_p \circ M^{-1}(u) = M \circ R\left(\frac{ub-a}{u-1}\right) = \frac{((u+1)m+3u+1)A(u)-(u+1)(m+1)}{((u+1)m+u+3)A(u)-(u+1)(m+1)}$$

and A is given as above. The computation was done using Maple symbolic software. □

3. EXTRANEOUS FIXED POINTS

The iterative procedure given by (1) can be written as

$$z_{k+1} = z_k - R_m(z_k), \qquad m = -r.$$

A fixed point of the iteration is a point ζ such that $R(\zeta) = 0$. For our method, there are points ζ which are **not** the solution of the nonlinear equation. These points are called extraneous fixed points. The points are classified as attracting, repelling or indifferent, based on the derivative of R_m.

In our case the map R_m is given by

$$R_m(z) = \frac{(1+m)f'(z)}{f''(z)} \left\{ \frac{1}{\left[1 - \frac{m}{m+1}\frac{f(z)f''(z)}{(f'(z))^2}\right]^{1/m}} - 1 \right\}.$$

For the quadratic polynomial $(z-a)(z-b)$, we have

$$R_m(z) = (m+1)(z - \tfrac{a+b}{2}) \left\{ \left(\frac{(m+1)(-2z+a+b)^2}{2(m+2)z^2 - (2a+2b)(m+2)z + m(a^2+b^2) + (a+b)^2} \right)^{\frac{1}{m}} - 1 \right\}$$

This map for any value of m vanishes at $z = a$, $z = b$ and $z = \frac{a+b}{2}$. The last one is the only extraneous fixed point.

We have found that $|R'_m(\frac{a+b}{2})| > 1$ for all values of interest of m and thus the point is repelling. These results are demonstrated in Fig. 4.1 where we can see that the boundary of the basins is the imaginary axis and the repelling point is on the boundary.

4. NUMERICAL EXPERIMENTS

In this section, we compare 5 members of Popovski's method. The methods are:

a. HM, Halley's method $r = -1$
b. Pm2, Popovski's method with $r = -2$
c. Pm3, Popovski's method with $r = -3$
d. Pm4, Popovski's method with $r = -4$
e. Pm8, Popovski's method with $r = -8$

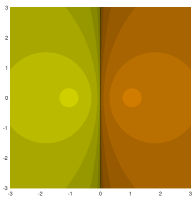
The computation of the cubic root (for $r = -3$) is more expensive than square root and thus in the case of $r = -4$ and $r = -8$, we have used square root successively.

We ran these five methods on seven examples on a 6 by 6 square centered at the origin. The functions are:

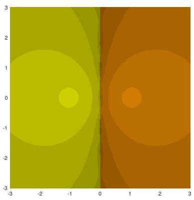
$$f_1(z) = z^2 - 1$$
$$f_2(z) = z^3 - 1$$
$$f_3(z) = z^5 - 1$$
$$f_4(z) = z^7 - 1$$
$$f_5(z) = (z^2 - 1/4)(z^2 - 1)(z^2 - 9/4) \quad \text{(Wilkinson-type polynomial)}$$
$$f_6(z) = z^{15} - z$$
$$f_7(z) = \left(e^{z+1} - 1\right)\left(e^{z-1} - 1\right) \quad \text{(Non-polynomial).}$$

The square is divided horizontally and vertically by equally spaced lines. We took the intersection of all these lines as initial points in the complex plane for the iterative schemes. The code collected the number of iterations required to converge within a tolerance of 10^{-7} and the root to which the sequence converged. If the sequence did not converge within 40 iteration, we denote it as a divergent point. Each point is colored by the color corresponding to the root. Note that we have used 6 different colors, therefore some roots will have the same color but they are far apart. Moreover, the shade of the color is darker for slower converging initial point. A divergent point is colored black. We also collected the CPU run-time to execute the code on all initial points using MacBook Pro computer.

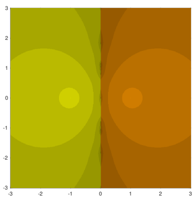
In Fig. 4.1 we have depicted the basins of attraction for the 5 methods of the first function. It is clear that all methods have performed the same and very well. The basins are separated by a vertical line through the origin, separating the square exactly in the middle. There are no lobes.



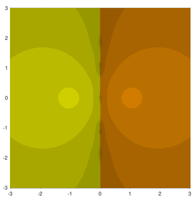
(a) HM



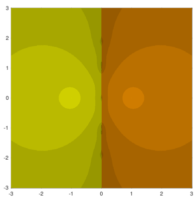
(b) Pm2



(c) Pm3



(d) Pm4



(e) Pm8

Fig. 4.1. Basins of attraction of analyzed methods for the roots of the function $f_1(z)$.

We have also collected in Tables 1 to 3, the average number of iterations per point for each scheme, the CPU run-time in seconds and the percentage of divergent points. The method Pm8 has the lowest number of iterations but the other schemes use only slightly more. On average Pm8 uses 4.89 and HM (the highest) uses 5.82. The fastest method is HM and the slowest is Pm3. The extra cost of Pm2-Pm4 and Pm8 is compensated by having fewer divergent points. But maybe Pm3 is much too costly to justify its use.

The basins of attraction for the methods in the second example are given in Fig. 4.2. Again Pm8 has the lowest number and HM the highest number of iterations. In fact this is true consistently for all seven examples. The method

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	Ex7	average
HM	3.88	4.43	5.35	6.19	6.26	9.44	5.23	5.82
Pm2	3.71	4.21	4.94	5.66	5.92	9.59	4.24	5.47
Pm3	3.66	4.11	4.77	5.42	5.78	9.11	3.88	5.25
Pm4	3.64	4.06	4.67	5.28	5.69	8.77	3.76	5.12
Pm8	3.60	3.97	4.47	4.97	5.51	8.12	3.61	4.89

Table 1. Average number of iterations per point for each example and each of the methods.

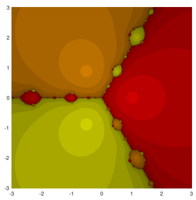
Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	Ex7	average
HM	128.266	222.771	383.329	535.911	1138.745	1215.177	299.842	560.577
Pm2	181.467	265.410	406.871	535.513	978.024	1211.534	404.578	569.057
Pm3	523.924	612.607	767.627	942.978	1228.775	942.978	677.603	813.784
Pm4	299.854	363.574	476.486	607.038	885.186	1263.360	452.870	621.072
Pm8	395.004	443.572	537.761	674.721	1010.587	1343.976	491.072	699.528

Table 2. CPU time (sec) for each example and each of the methods.

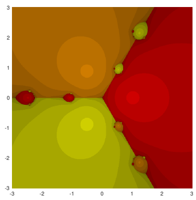
Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	Ex7	average
HM	$1.66(-3)$	$5.54(-6)$	$5.81(-5)$	$2.46(-4)$	$1.66(-3)$	$3.15(-4)$	$2.35(-2)$	$3.92(-3)$
Pm2	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$1.77(-3)$	$5.90(-3)$	$1.10(-3)$
Pm3	$2.77(-6)$	$5.54(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$1.53(-3)$	$8.75(-4)$	$3.46(-4)$
Pm4	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$1.30(-3)$	$5.20(-4)$	$2.62(-4)$
Pm8	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$2.77(-6)$	$1.05(-3)$	$1.32(-4)$	$1.71(-4)$

Table 3. Percent of black points for each example and each of the methods.

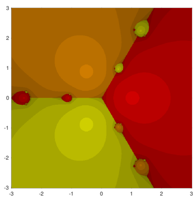
HM is again the fastest, but it has more divergent points than Pm2, Pm4 and Pm8.



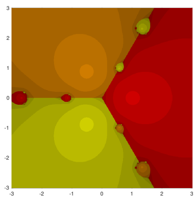
(a) HM



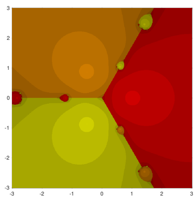
(b) Pm2



(c) Pm3

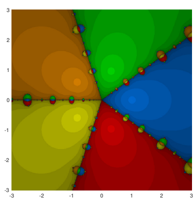


(d) Pm4

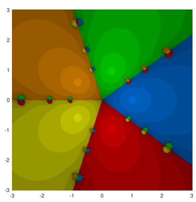


(e) Pm8

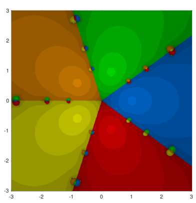
Fig. 4.2. Basins of attraction of analyzed methods for the roots of the function $f_2(z)$.



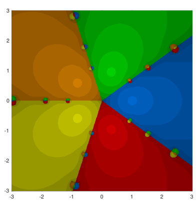
(a) HM



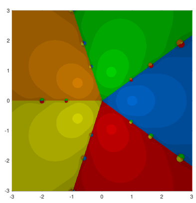
(b) Pm2



(c) Pm3

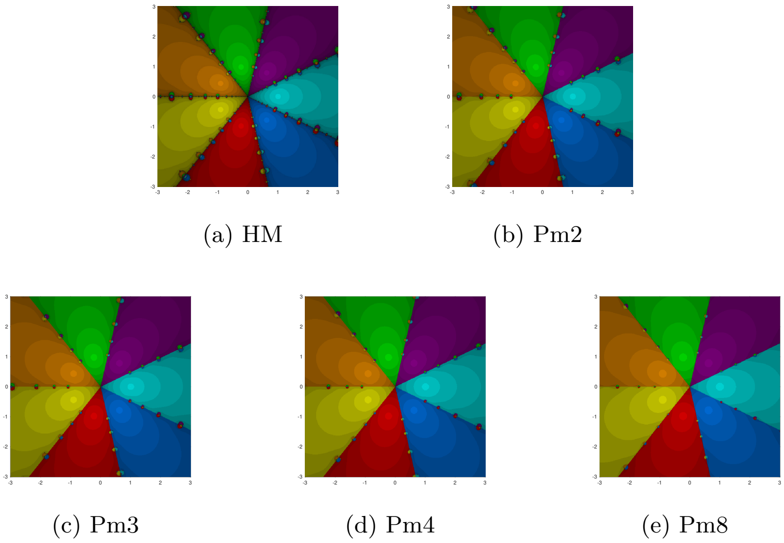


(d) Pm4



(e) Pm8

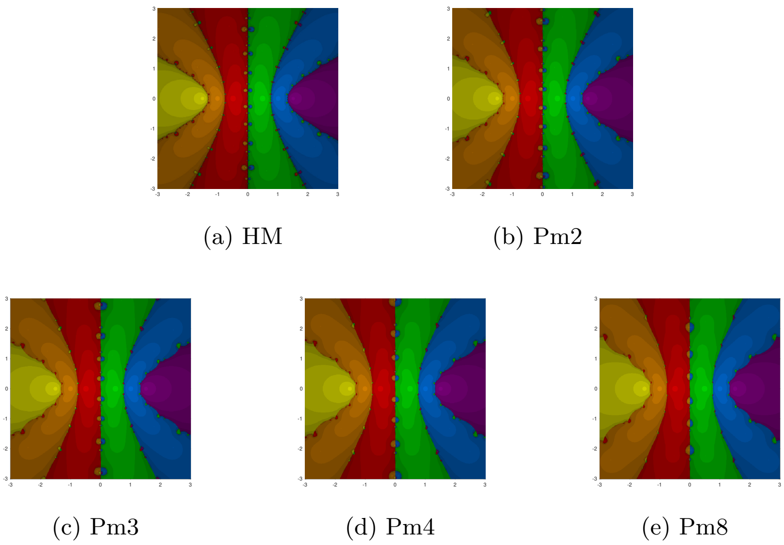
Fig. 4.3. Basins of attraction of analyzed methods for the roots of the function $f_3(z)$.



(a) HM (b) Pm2

(c) Pm3 (d) Pm4 (e) Pm8

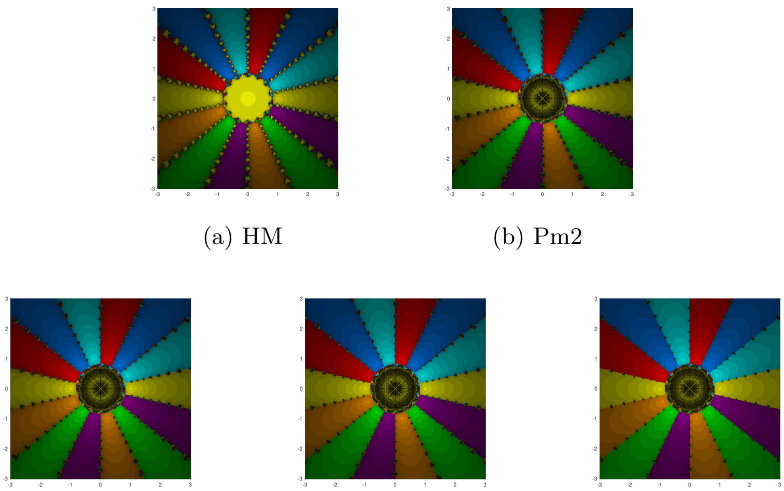
Fig. 4.4. Basins of attraction of analyzed methods for the roots of the function $f_4(z)$.



(a) HM (b) Pm2

(c) Pm3 (d) Pm4 (e) Pm8

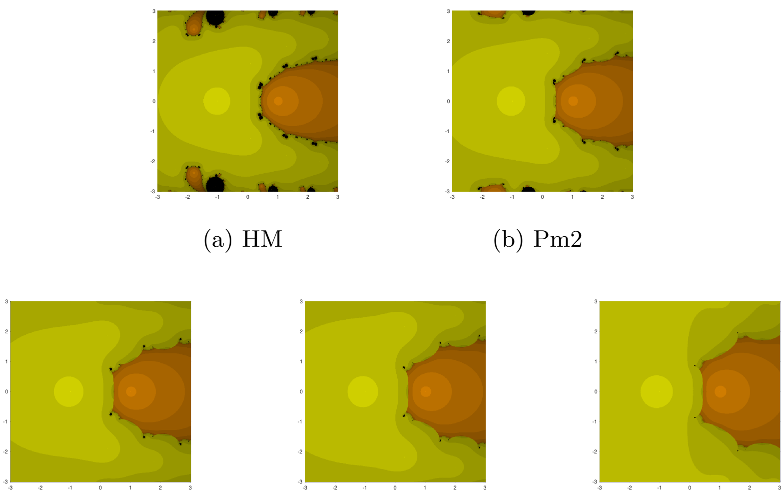
Fig. 4.5. Basins of attraction of analyzed methods for the roots of the function $f_5(z)$.



(a) HM (b) Pm2

(c) Pm3 (d) Pm4 (e) Pm8

Fig. 4.6. Basins of attraction of analyzed methods for the roots of the function $f_6(z)$.



(a) HM (b) Pm2

(c) Pm3 (d) Pm4 (e) Pm8

Fig. 4.7. Basins of attraction of analyzed methods for the roots of the function $f_7(z)$.

The numerical results over the seven examples show that Pm8 uses the lowest number of iterations but require more CPU run-time than the others. The fastest methods on average are HM and Pm2 and the slowest method is Pm3. In terms of divergent points Pm3, Pm4 and Pm8 have fewer than other methods on average. In fact, HM has fewer divergent points only for f_6. In all other examples, Pm8 has the lowest number. We can conclude that Pm2 and Pm4 are competitive except for the CPU run-time. Pm3 is the slowest of all because of the cost of computing the cubic root at each step. We should note that for the non-polynomial (last) example the basins are not equally divided as in the first example, even though both have the same roots $z = \pm 1$.

We have included a problem related to Planck's radiation law [19]. The energy density $H(y)$ is related to the wavelength of radiation (y), absolute temperature of the black-body (T), Boltzman constant (k), speed of light (c) and Planck's constant (h) via

$$H(y) = \frac{8\pi chy^{-5}}{e^{ch/(ykT)} - 1}$$

In order to evaluate the wavelength corresponding to the maximum energy density, we solve $H'(y) = 0$. Let us introduce the variable $x = \frac{ch}{ykT}$, then we get

$$e^{-x} + \frac{x}{5} - 1 = 0.$$

The two solutions of this equation are $x_1 = 0$ and $x_2 = 5 + LambertW(-5e^{-5}) \approx 4.9651142317$. For the LambertW function see Corless *et al.* [20]. Clearly the only physical solution is the second. We ran all 5 methods on this example on a rectangle $[-3, 8] \times [-3, 3]$ and the results did not change our conclusions.

5. MULTIPLE ROOTS

In contrast to Herceg and Petković, we developed a family of methods for multiple roots by applying Popovski's family to the quotient $\frac{f}{f'}$. Of course, this will require the knowledge of the third order derivative of f and thus the number of function-evaluation per iteration is increased by one.

The family of methods becomes

$$z_{k+1} = z_k - \frac{(1-r)g'(z_k)}{g''(z_k)} \left\{ \left[1 - \frac{r}{r-1} \frac{g(z_k)g''(z_k)}{(g'(z_k))^2} \right]^{1/r} - 1 \right\} \tag{2}$$

where $g(z) = \frac{f(z)}{f'(z)}$.

We ran five members of the family, denoted gHM $(r = -1)$, gPm2 $(r = -2)$, gPm3 $(r = -3)$, gPm4 $(r = -4)$ and gPm8 $(r = -8)$ on six examples with various multiplicities and various sizes of squares centered in the origin. Because the domains are of different sizes, then the number of initial points is different. Therefore we should not average the results across the examples as was done in the simple root case, but take the CPU run-time per, say, 1000 initial points.

Example 1. *On* $[-2, 2] \times [-2, 2]$

$$\phi_1(z) = (z^3 - 1)^3.$$

Example 2. *On* $[-3, 3] \times [-3, 3]$

$$\phi_2(z) = (z^5 - 1)^4.$$

Example 3. *On* $[-2, 2] \times [-2, 2]$

$$\phi_3(z) = (z^{15} - z)^4.$$

Example 4. *On* $[-3, 3] \times [-3, 3]$

$$\phi_4(z) = \Big((z^2 - 1/4)(z^2 - 1)(z^2 - 9/4)(z^2 - 4)\Big)^2.$$

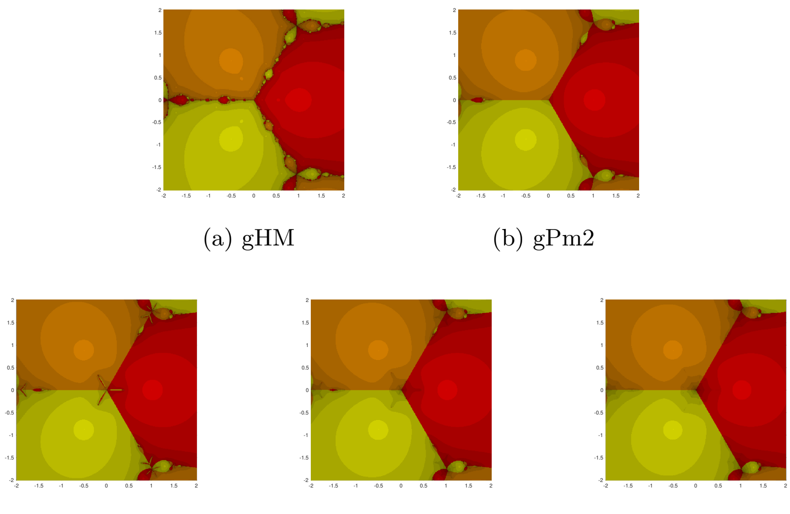
Example 5. *On* $[-4, 4] \times [-4, 4]$

$$\phi_5(z) = \Big((z + 1)(z^8 - 6561)(z^4 - 1)\Big)^2.$$

Example 6. *On* $[-4, 4] \times [-4, 4]$

$$\phi_6(z) = \Big((e^{z-1} - 1)(z^2 + 1/4)\cos(z)\Big)^2.$$

The basins of attraction for the first example are given in Fig. 5.8. It seems that Halley's method has more lobes at the boundary than the other methods.



(a) gHM (b) gPm2

(c) gPm3 (d) gPm4 (e) gPm8

Fig. 5.8. Basins of attraction of analyzed methods for the roots of the function $\phi_1(z)$.

We have collected the numerical results in Tables 4 to 6. Note that Table 5 lists the CPU run-time in sec per 1000 initial points to account for the disparity in domain size. The method gPm2 has the lowest number of iterations in the first example, gPm4 in the second example. The member gHm has the lowest number of iterations in Examples 3 and 5 and also on average across the examples. The member gPm8 has the lowest number in Examples 4 and 6. The method gPm3 never reached that low. Based on Table 5, we see that the run-time is the highest for gPm3, even higher than gPm8. The fastest method in four examples and average overall is gHM followed by gPm4 and gPm8. The number of divergent points is zero for gHM for Examples 3 and 5. For Example 1 all methods have the same number of divergent points. For Example 4, all methods except gHM have the same small number of divergent points. On average gPm2 has the lowest number of divergent points.

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	average
gHM	3.87	5.79	2.82	8.44	4.17	6.42	5.25
gPm2	3.79	5.24	13.04	8.22	4.33	5.58	6.70
gPm3	3.84	4.85	13.82	8.18	4.40	5.46	6.75
gPm4	3.86	4.42	14.63	8.16	4.43	5.38	6.82
gPm8	3.88	5.48	17.70	8.13	4.49	5.27	7.49

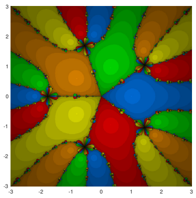
Table 4. Average number of iterations per point for each example and each of the methods.

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	average
gHM	1.026	2.269	2.149	16.197	6.269	6.839	6.699
gPm2	1.355	2.562	5.910	19.789	10.890	6.120	8.661
gPm3	2.168	3.383	7.846	17.859	7.933	7.072	8.126
gPm4	1.706	2.463	5.435	14.827	6.524	6.808	6.852
gPm8	2.112	3.464	6.829	15.405	6.735	6.727	7.258

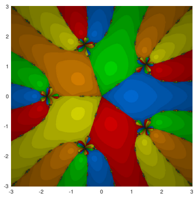
Table 5. CPU time (sec) per 1000 initial points for each example and each of the methods.

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	average
gHM	6.22(-6)	8.30(-6)	0.	1.67(-3)	0.	3.30(-2)	5.78(-3)
gPm2	6.22(-6)	2.77(-6)	2.24(-1)	2.77(-6)	1.56(-6)	1.54(-2)	4.00(-2)
gPm3	6.22(-6)	2.77(-6)	2.43(-1)	2.77(-6)	1.56(-6)	1.13(-2)	4.26(-2)
gPm4	6.22(-6)	1.11(-5)	2.60(-1)	2.77(-6)	1.56(-6)	1.18(-2)	4.58(-2)
gPm8	6.22(-6)	2.77(-6)	3.34(-1)	2.77(-6)	1.56(-6)	9.14(-3)	5.66(-2)

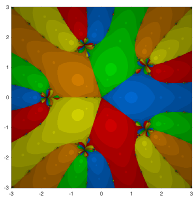
Table 6. Percent of black points for each example and each of the methods.



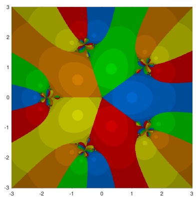
(a) gHM



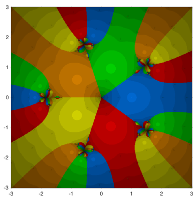
(b) gPm2



(c) gPm3

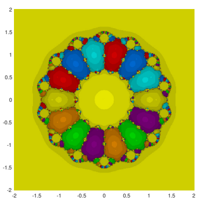


(d) gPm4

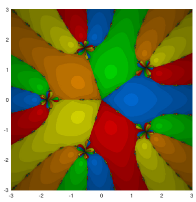


(e) gPm8

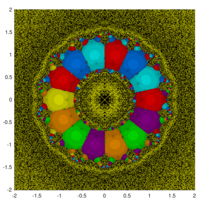
Fig. 5.9. Basins of attraction of analyzed methods for the roots of the function $\phi_2(z)$.



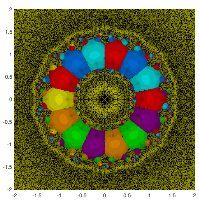
(a) gHM



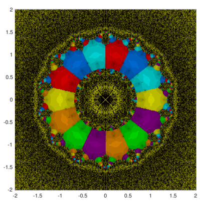
(b) gPm2



(c) gPm3



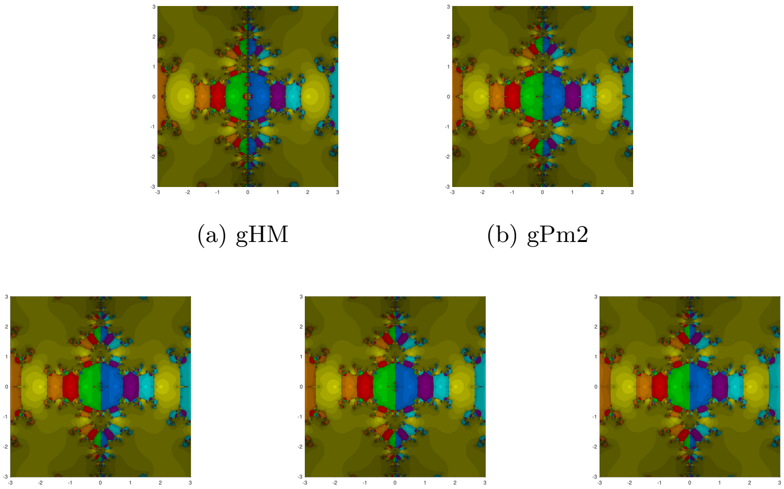
(d) gPm4



(e) gPm8

Fig. 5.10. Basins of attraction of analyzed methods for the roots of the function $\phi_3(z)$.

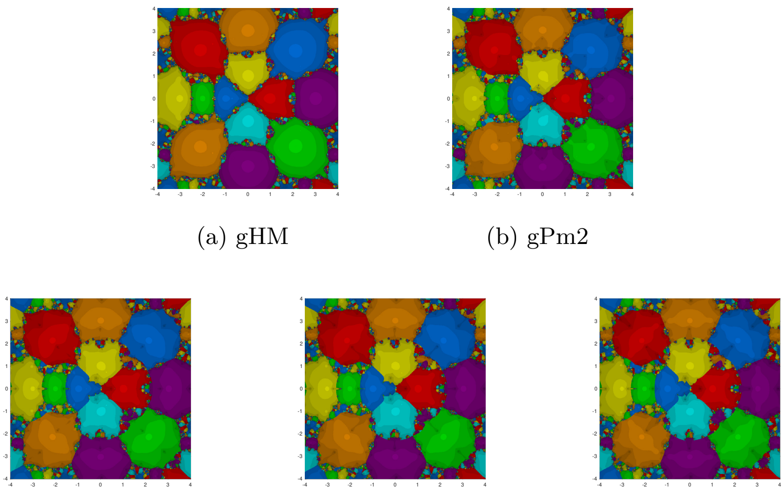
Based on the 6 examples we find that gHM and gPm2 are better than the other members tested.



(a) gHM (b) gPm2

(c) gPm3 (d) gPm4 (e) gPm8

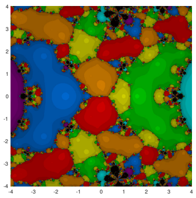
Fig. 5.11. Basins of attraction of analyzed methods for the roots of the function $\phi_4(z)$.



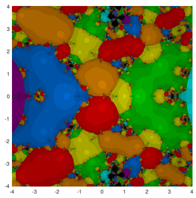
(a) gHM (b) gPm2

(c) gPm3 (d) gPm4 (e) gPm8

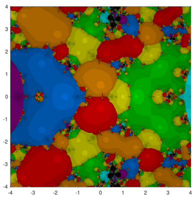
Fig. 5.12. Basins of attraction of analyzed methods for the roots of the function $\phi_5(z)$.



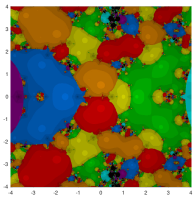
(a) gHM



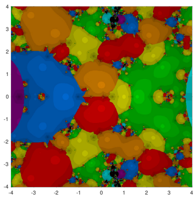
(b) gPm2



(c) gPm3



(d) gPm4



(e) gPm8

Fig. 5.13. Basins of attraction of analyzed methods for the roots of the function $\phi_6(z)$.

6. CONCLUSIONS

In this manuscript, we have analyzed five members of the Popovski's family. We have noted that methods with $r = -2^k$ for k integer can use the square root successively and cost less than using $1/r$ power directly. In the case of simple roots, we have shown that there is some benefit in taking $r = -2$ and $r = -4$. In the multiple root case, we find that gHm and gPm2 are better than the other members. This is the same conclusion reached by Herceg and Petković for their idea for multiple roots requiring the knowledge of the multiplicity instead of another function evaluation.

REFERENCES

[1] J.F. Traub, *Iterative methods for the solution of equations*, Prentice-Hall, Englewood Cliffs, 1964.
[2] M. Petković, B. Neta, L. Petković, J. Džunić, *Multipoint Methods for Solving Nonlinear Equations*. Academic Press, Boston, 2013. `https://doi.org/10.1016/j.amc.2013.10.072`
[3] D.B. Popovski, *A family of one point iteration formulae for finding roots*, Int. J. Comput. Math., **8** (1980), 85–88. `https://doi.org/10.1080/00207168008803193`
[4] E. Halley, *A new, exact and easy method of finding the roots of equations generally and that without any previous reduction*, Phil. Trans. Roy. Soc. London, **18** (1694), 136–148. `http://dx.doi.org/10.1098/rstl.1694.0029`
[5] A. Cauchy, *Sur la Determination Approximative des Racines d'une Équation Algebrique ou Transcendante*, Oeuvres Completès Série 2, Tome 4, Gauthier Villars, Paris, 1899, pp. 573–609.

[6] S. Hitotumatu, *A method of successive approximation based on the expansion of second order*, Math. Japan, **7** (1962), 31–50.

[7] B. Neta, *On Popovski's method for nonlinear equations*, Appl. Math. Comput., **201** (2008), 710–715. https://doi.org/10.1016/j.amc.2008.01.012

[8] M. Scott, B. Neta, C. Chun, *Basin attractors for various methods*, Appl. Math. Comput., **218** (2011), 2584–2599. https://doi.org/10.1016/j.amc.2011.07.076

[9] B. Neta, M. Scott, C. Chun, *Basins of attraction for several methods to find simple roots of nonlinear equations*, Appl. Math. Comput., **218** (2012), 10548–10556. https://doi.org/10.1016/j.amc.2012.04.017

[10] B. Neta, C. Chun, *On a family of Halley-like methods to find simple roots of nonlinear equations*, Appl. Math. Comput., **219** (2013), 7940–7944. https://doi.org/10.1016/j.amc.2013.02.035

[11] C. Chun, B. Neta, *Comparative study of methods of various orders for finding simple roots of nonlinear equations*, J. Appl. Anal. Comput., **9** (2019), 400–427. https://doi.org/10.11948/2156-907X.20160229

[12] E.R. Vrscay, W.J. Gilbert, *Extraneous fixed points, basin boundaries and chaotic dynamics for Schröder and König rational iteration functions*, Numer. Math., **52** (1988), 1–16. https://doi.org/10.1007/BF01401018

[13] D. Herceg, I. Petković, *Computer visualization and dynamic study of new families of root-solvers*, J. Comput. Appl. Math., **401** (2022) 113775, https://doi.org/10.1016/j.cam.2021.113775

[14] I. Petković, B. Neta, *On an application of symbolic computation and computer graphics to root-finders: The case of multiple roots of unknown multiplicity*, J. Comput. Appl. Math., **308** (2016), 215–230. https://doi.org/10.1016/j.cam.2016.06.008

[15] C. Chun, B. Neta, *Basins of attraction for several third order methods to find multiple roots of nonlinear equations*, Appl. Math. Comput., **268** (2015), 129–137. https://doi.org/10.1016/j.amc.2015.06.068

[16] B. Neta, C. Chun, *On a family of Laguerre methods to find multiple roots of nonlinear equations*, Appl. Math. Comput., **219** (2013), 10987–11004. https://doi.org/10.1016/j.amc.2013.05.002

[17] S. Amat, S. Busquier, S. Plaza, *Dynamics of a family of third-order iterative methods that do not require using second derivatives*, Appl. Math. Comput., **154** (2004), 735–746. https://doi.org/10.1016/S0096-3003(03)00747-1

[18] A.F. Beardon, *Iteration of Rational Functions Complex Analytic Dynamical Systems*, Springer-Verlag, New York, 1991.

[19] O. Passon, J. Grebe-Ellis, *Planck's radiation law, the light quantum, and the prehistory of indistinguishability in the teaching of quantum mechanics*, Eur. J. Phys., **3** (2017), 035404. https://doi.org/10.1088/1361-6404/aa6134

[20] R.M. Corless, G.H. Gonnet, D.E.G. Hare, D.J. Jeffrey, D.E. Knuth, *On the LambertW Function*, Adv. Comput. Math., **5** (1996), 329–359, https://doi.org/10.1007/BF02124750

Received by the editors: received: November 23, 2021; accepted: June 20, 2022; published online: August 25, 2022.